FreedomBox Manual

Contents

1 FreedomBox: take your online privacy back **1**
1.1 Typical usage: Private Cloud . . . 1
1.2 Typical usage: Network-Attached Storage (NAS) . . . 1
1.3 Advanced usage: Smart Home Router . . . 1
1.4 Advanced usage: For Communities . . . 2
1.5 FreedomBox Interface . . . 4
1.5.1 Screenshot . . . 4
1.5.2 Video resources . . . 5

2 Quick Start **5**
2.1 What you need to get started . . . 5
2.2 How to get started . . . 5
2.3 Finding your way around . . . 9
2.3.1 Front page . . . 9
2.3.2 Apps menu . . . 9
2.3.3 Help menu . . . 10
2.3.4 System menu . . . 10
2.3.5 User menu . . . 11
2.3.6 Burger menu . . . 11

3 Getting Help **12**
3.1 Discussion Forum . . . 12
3.2 Matrix . . . 12
3.3 IRC #freedombox . . . 13
3.4 Email . . . 13
3.5 Help Back . . . 13

4 Download and Install **13**
4.1 Downloading on Debian . . . 13
4.2 Downloading for SBC or Virtual Machine . . . 13
4.2.1 Prepare your device . . . 13
4.2.2 Downloading Images . . . 13
4.2.3 Verifying the Downloaded Images . . . 14
4.2.4 Installation . . . 14
4.2.5 Troubleshooting . . . 15
4.3 Obtaining Source Code . . . 15
4.3.1 From within FreedomBox . . . 16
4.3.2 Other Ways to Obtain Source Code . . . 16
4.3.3 Building Disk Images . . . 16
4.3.3.1 U-boot on Pioneer Edition Images . . . 17
4.3.4 Old Signing Keys . . . 17

5 Apps **17**

5.1 Bepasty (File & Snippet Sharing) . . . 17
5.1.1 What is bepasty? . . . 18
5.1.2 Screenshot . . . 18
5.1.3 Passwords and Permissions . . . 19
5.1.4 Distributing passwords . . . 19
5.1.5 Using bepasty . . . 19
5.1.6 Managing passwords . . . 19
5.1.7 External links . . . 20
5.2 Calibre (e-Library) . . . 20
5.2.1 Managing Libraries . . . 20
5.2.2 Access . . . 20
5.2.3 External links . . . 20
5.3 Coturn (VoIP Helper) . . . 21
5.3.1 How it works . . . 21
5.3.2 Configuring Matrix Synapse . . . 22
5.3.3 Port Forwarding . . . 22
5.3.4 External links . . . 22
5.4 Deluge (Distributed File Sharing via BitTorrent) . . . 22
5.4.1 What is Deluge? . . . 23
5.4.2 Screenshot . . . 23
5.4.3 Initial Setup . . . 23
5.4.4 External links . . . 25
5.5 Ejabberd (Chat Server) . . . 25
5.5.1 What is ejabberd? . . . 25
5.5.2 What is XMPP? . . . 25
5.5.3 Privacy . . . 25
5.5.4 Setting the Domain Name . . . 26
5.5.5 Registering FreedomBox users to use XMPP . . . 26
5.5.6 Port Forwarding . . . 26
5.5.7 Compatible clients . . . 26
5.5.7.1 FreedomBox webclient . . . 27
5.5.7.2 Mobile clients . . . 27
5.5.7.2.1 Conversations (Android) . . . 27
5.5.7.2.2 Monal (iOS) . . . 28
5.5.7.2.3 Siskin (iOS) . . . 28
5.5.7.3 Desktop clients . . . 28
5.5.7.3.1 Gajim (Windows, MacOS, Linux) . . . 28
5.5.7.3.2 Dino (Linux) . . . 29

5.5.7.3.3 Movim (Linux) . . . 29
5.5.7.3.4 Monal (MacOS) . . . 29
5.5.8 External links . . . 30
5.5.8.1 Client Apps . . . 30
5.6 Postfix/Dovecot/Rspamd (Email Server) . . . 30
5.6.1 About the Email Server . . . 30
5.6.2 Prerequisites . . . 31
5.6.3 Installing . . . 31
5.6.4 Configuring the Email Server . . . 33
5.6.5 Using the Email Server . . . 34
5.6.5.1 With FreedomBox Webmail Client (RoundCube) . . . 34
5.6.5.2 With Thunderbird . . . 34
5.6.5.3 Manual Configuration . . . 34
5.6.5.4 Email Aliases . . . 35
5.6.5.5 Automatic Email Aliases . . . 35
5.6.6 Advanced: Troubleshooting . . . 36
5.6.6.1 How to debug an action script failure? How to access the system log? . . . 36
5.6.6.2 Why does the server say "relay access denied"? . . . 36
5.6.6.3 Cannot send anything from Roundcube. It says "SMTP Error (250): Authentication failed". . . . 36
5.6.7 Providing user feedback . . . 36
5.6.8 Technical info and discussion . . . 36
5.6.9 External links . . . 37
5.6.9.1 Client Apps . . . 37
5.7 Feather Wiki (Personal Notebooks) . . . 37
5.7.1 A Nest of Feather Wikis . . . 38
5.7.2 Setting up wikis . . . 38
5.7.3 Publishing a wiki . . . 38
5.7.4 As many wikis as you need . . . 39
5.7.5 Use cases . . . 39
5.7.6 External links . . . 39
5.8 GitWeb (Simple Git Hosting) . . . 39
5.8.1 What is Git . . . 40
5.8.2 Managing the repositories . . . 40
5.8.3 Access . . . 40
5.8.4 HTTP basic auth . . . 40
5.8.5 Mirroring . . . 41
5.8.6 Enabling/Disabling Features . . . 41
5.8.7 External links . . . 41
5.9 Home Assistant . . . 41

5.9.1 Hardware . . . 42
5.9.2 Need for Subdomain . . . 43
5.9.3 Setup . . . 43
5.9.4 Usage . . . 46
5.9.5 Advanced Manual Configuration . . . 48
5.9.6 External links . . . 48
5.10 Ikiwiki (Wiki and Blog) . . . 48
5.10.1 What is Ikiwiki? . . . 48
5.10.2 Creating a wiki or blog . . . 48
5.10.3 Accessing your wiki or blog . . . 49
5.10.4 User login through SSO . . . 49
5.10.5 Adding FreedomBox users as wiki admins . . . 50
5.10.6 Avoiding Spam . . . 50
5.10.7 Theming . . . 50
5.10.8 External links . . . 50
5.11 Infinoted (Collaborative text edition with Gobby) . . . 51
5.11.1 Port Forwarding . . . 51
5.11.2 Extenal links . . . 51
5.11.2.1 Client apps . . . 51
5.12 Janus (WebRTC server) . . . 51
5.12.1 Screenshot . . . 52
5.12.2 Using Janus . . . 53
5.12.3 External links . . . 53
5.13 JSXC (Web Chat Client) . . . 53
5.13.1 Technical Specifications . . . 53
5.13.2 Installation . . . 53
5.13.3 Usage . . . 54
5.13.4 Port Forwarding . . . 54
5.13.5 External links . . . 54
5.13.5.1 Client apps . . . 54
5.14 Kiwix (Offline Wikipedia) . . . 54
5.14.1 Use Cases . . . 55
5.14.2 Kiwix on FreedomBox . . . 56
5.14.3 Finding content archives . . . 56
5.14.4 Creating content archives . . . 56
5.14.5 External links . . . 56
5.15 Matrix Synapse (Chat Server) . . . 56
5.15.1 What is Matrix? . . . 57
5.15.2 How to access your Matrix Synapse server? . . . 57

5.15.3 Port Forwarding . . . 57
5.15.4 Setting up Matrix Synapse on your FreedomBox . . . 57
5.15.5 Setting up Audio/Video calls . . . 58
5.15.6 Federating with other Matrix instances . . . 58
5.15.7 Memory usage . . . 58
5.15.8 Advanced usage . . . 58
5.15.9 External links . . . 59
5.15.9.1 Client apps . . . 59
5.16 MediaWiki (Wiki) . . . 59
5.16.1 About MediaWiki . . . 59
5.16.2 MediaWiki on FreedomBox . . . 60
5.16.2.1 User management . . . 60
5.16.2.2 Use cases . . . 60
5.16.2.2.1 Personal Knowledge Repository . . . 60
5.16.2.2.2 Community Wiki . . . 60
5.16.2.2.3 Personal Wiki-based Website . . . 60
5.16.2.3 Editing Wiki Content . . . 60
5.16.2.3.1 WikiText Editor . . . 60
5.16.2.3.2 Visual Editor . . . 61
5.16.2.3.3 Other Formats . . . 61
5.16.2.3.4 Image Uploads . . . 61
5.16.2.4 Customization . . . 61
5.16.2.4.1 Skins . . . 61
5.16.3 External links . . . 61
5.17 Minetest (Block Sandbox) . . . 61
5.17.1 Port Forwarding . . . 62
5.17.2 Install and enable mods . . . 62
5.17.3 External links . . . 62
5.17.3.1 Client Apps . . . 63
5.18 MiniDLNA / ReadyMedia (Simple Media Server) . . . 63
5.18.1 What is UPnP/DLNA? . . . 63
5.18.2 Setting up MiniDLNA on your FreedomBox . . . 63
5.18.3 Using MiniDLNA to play media on your devices . . . 64
5.18.4 Supported media formats . . . 64
5.18.5 File systems for external drives . . . 64
5.18.6 External links . . . 65
5.19 Miniflux (News Feed Reader) . . . 65
5.19.1 Clients . . . 65
5.19.2 Reader . . . 65

5.19.3 Other Features . . . 66
5.19.4 Miniflux on FreedomBox . . . 66
5.19.4.1 User Management . . . 66
5.19.4.2 Migrating from Tiny Tiny RSS . . . 66
5.19.4.3 Integrations . . . 66
5.19.5 External links . . . 66
5.20 Mumble (Voice Chat) Server . . . 66
5.20.1 What is Mumble? . . . 67
5.20.2 Using Mumble . . . 67
5.20.3 Port Forwarding . . . 67
5.20.4 Managing Permissions . . . 67
5.20.5 External links . . . 67
5.21 Nextcloud (File Storage & Collaboration) . . . 68
5.21.1 What is Nextcloud? . . . 68
5.22 OpenVPN (Virtual Private Network) . . . 68
5.22.1 What is OpenVPN? . . . 69
5.22.2 Port Forwarding . . . 69
5.22.3 Setting up . . . 69
5.22.4 Troubleshooting . . . 70
5.22.5 Browsing Internet after connecting to VPN . . . 70
5.22.6 Usage . . . 71
5.22.6.1 On Android/LineageOS . . . 71
5.22.6.2 On Debian . . . 75
5.22.7 Checking if you are connected . . . 75
5.22.7.1 On Debian . . . 75
5.22.8 Accessing internal services . . . 76
5.22.9 External Links . . . 76
5.23 Privoxy (Web Proxy) . . . 76
5.23.1 Screencast . . . 77
5.23.2 Setting up . . . 77
5.23.3 Advanced Users . . . 78
5.23.4 External links . . . 78
5.24 Quassel (Text Chat Client via IRC) . . . 79
5.24.1 Why run Quassel? . . . 79
5.24.2 How to setup Quassel? . . . 79
5.24.3 Port Forwarding . . . 80
5.24.4 Clients . . . 81
5.24.4.1 Desktop . . . 81
5.24.4.2 Android . . . 86

5.24.5 External links . . . 87
5.25 Radicale (Calendar and Addressbook) . . . 87
5.25.1 Why should I run Radicale? . . . 88
5.25.2 How to setup Radicale? . . . 88
5.25.3 Synchronizing over Tor . . . 91
5.25.4 Synchronizing with your Android phone . . . 92
5.25.5 Advanced Users . . . 93
5.25.5.1 Sharing resources . . . 93
5.25.5.2 Importing files . . . 94
5.25.6 External links . . . 94
5.26 Roundcube (Email Client) . . . 94
5.26.1 What is Roundcube? . . . 94
5.26.2 Email privacy . . . 94
5.26.3 Using Roundcube . . . 95
5.26.4 Using Gmail with Roundcube . . . 95
5.26.5 External links . . . 96
5.27 RSS Bridge (RSS Feed Generator) . . . 96
5.27.1 What is RSS Bridge? . . . 97
5.27.2 Usage Example . . . 97
5.27.2.1 Subscribing to a YouTube account . . . 97
5.27.2.2 Subscribing to feed with Tiny Tiny RSS . . . 99
5.27.3 External links . . . 100
5.28 Samba (Network File Storage) . . . 100
5.28.1 Using Samba . . . 101
5.28.1.1 Connecting from an Android device . . . 101
5.28.1.2 Connecting from a macOS device . . . 101
5.28.2 Integration with other apps . . . 101
5.28.3 Comparison with other apps . . . 101
5.28.3.1 Syncthing . . . 101
5.28.4 External links . . . 101
5.29 Searx (Web Search) . . . 102
5.29.1 About Searx . . . 102
5.29.2 Screenshot . . . 102
5.29.3 Screencast . . . 102
5.29.4 Why use Searx? . . . 103
5.29.4.1 Personalization and Filter Bubbles . . . 103
5.29.4.2 Advertisement filtering . . . 103
5.29.4.3 Privacy . . . 103
5.29.5 Searx on FreedomBox . . . 103

5.29.6 External links . . . 103
5.30 Shaarli (Bookmarks) . . . 103
5.30.1 What is Shaarli? . . . 104
5.30.2 External links . . . 105
5.31 Shadowsocks (Bypass Censorship) . . . 105
5.31.1 What is Shadowsocks? . . . 105
5.31.2 Using Shadowsocks? . . . 105
5.31.3 Configuring your FreedomBox for Shadowsocks Client . . . 106
5.31.4 Configuring your FreedomBox for Shadowsocks Server . . . 106
5.31.4.1 Port Forwarding . . . 106
5.31.5 External links . . . 106
5.32 Sharing (File Publishing) . . . 106
5.32.1 What Is Sharing App? . . . 107
5.32.2 Setting Up Shares . . . 107
5.32.3 Providing/Updating Content . . . 107
5.32.4 Technicalities . . . 108
5.32.5 External Links . . . 108
5.33 Syncthing (File Synchronization) . . . 108
5.33.1 Synchronizing over Tor . . . 109
5.33.2 Avoiding Syncthing Relays . . . 109
5.33.3 Using Syncthing with other applications . . . 109
5.33.3.1 Password Manager . . . 109
5.33.3.2 Note-taking and Journaling . . . 110
5.33.4 External links . . . 110
5.34 TiddlyWiki (Non-linear Notebooks) . . . 110
5.34.1 TiddlyWiki on FreedomBox . . . 111
5.34.1.1 Bandwidth considerations with the Autosave setting . . . 111
5.34.2 Use cases . . . 111
5.34.3 Upgrades . . . 112
5.34.4 Tips . . . 112
5.34.4.1 Setting a favicon . . . 112
5.34.5 External links . . . 112
5.35 Tiny Tiny RSS (News Feed Reader) . . . 113
5.35.1 Using the Web Interface . . . 113
5.35.1.1 Adding a new feed . . . 113
5.35.1.2 Importing your feeds from another feed reader . . . 116
5.35.2 Using the Mobile App . . . 117
5.35.3 RSS Bridge . . . 118
5.35.4 External links . . . 118

5.36 Tor (Anonymity Network) . . . 118
5.36.1 What is Tor? . . . 119
5.36.2 Using Tor to browse anonymously . . . 119
5.36.3 Using Tor Onion Service to access your FreedomBox . . . 119
5.36.4 Apps accessible via Tor . . . 120
5.36.5 Running a Tor relay . . . 121
5.36.6 External links . . . 121
5.37 Tor Proxy (Anonymity Network) . . . 121
5.37.1 What is Tor Proxy? . . . 121
5.37.2 Using Tor Browser to browse anonymously . . . 121
5.37.3 (Advanced) Using Tor Proxy as a SOCKS proxy . . . 122
5.37.3.1 Example with Firefox . . . 122
5.37.4 Circumventing Tor censorship . . . 122
5.37.5 External links . . . 123
5.37.5.1 Client Apps . . . 123
5.38 Transmission (Distributed File Sharing via BitTorrent) . . . 124
5.38.1 What is Transmission ? . . . 124
5.38.2 Screenshot . . . 125
5.38.3 Using Transmission . . . 125
5.38.4 Tips . . . 125
5.38.4.1 Transferring Downloads from the FreedomBox . . . 125
5.38.5 Port Forwarding . . . 125
5.38.6 Using Remote Apps . . . 126
5.38.7 External Links . . . 126
5.39 User Websites . . . 126
5.39.1 What is User Websites? . . . 126
5.39.2 Screenshot . . . 126
5.39.3 Using User Websites . . . 127
5.39.4 Creating public_html folder and uploading documents . . . 127
5.39.4.1 Visually from Linux . . . 127
5.39.4.2 Visually from Other Plattforms . . . 128
5.39.4.3 With a Command Line Interface (CLI) . . . 128
5.39.5 External Links . . . 129
5.40 WireGuard (Virtual Private Network) . . . 129
5.40.1 About WireGuard . . . 130
5.40.2 Installation . . . 130
5.40.3 Usage . . . 130
5.40.4 Configuration - Debian Peers . . . 130
5.40.5 Configuration - Mobile Clients . . . 130

5.40.5.1 Alternative A - Create configuration manually . . . 130
5.40.5.2 Alternative B - Create configuration from archive . . . 131
5.40.5.3 Alternative C - Import by reading a QR code (most secure method) . . . 131
5.40.6 External Links . . . 131
5.41 WordPress (Website and Blog) . . . 131
5.41.1 What is WordPress? . . . 132
5.41.2 Setting up WordPress on FreedomBox . . . 134
5.41.3 Public Access . . . 135
5.41.4 Users . . . 135
5.41.5 WordPress as Home Page . . . 135
5.41.6 Domain Name . . . 135
5.41.7 Permalinks . . . 135
5.41.8 Automatic Upgrades . . . 136
5.41.9 Plugins and Themes . . . 136
5.41.9.1 Google Fonts Privacy Issue . . . 136
5.41.9.2 Failing Updates . . . 137
5.41.9.3 Note for FreedomBox between version 21.7 and 22.12 . . . 137
5.41.10 External links . . . 137
5.42 Zoph (Photo manager) . . . 137
5.42.1 What is Zoph? . . . 138
5.42.2 Using Zoph . . . 138
5.42.2.1 Choosing a storage location for your photos . . . 139
5.42.3 External links . . . 140

6 System 140

6.1 Backups . . . 140
6.1.1 Notes for Specific App Backups . . . 140
6.1.2 How to install and use Backups . . . 140
6.1.3 External links . . . 143
6.2 BIND (Domain Name Server) . . . 143
6.2.1 Using BIND . . . 144
6.2.2 External links . . . 144
6.3 Cockpit (Server Administration) . . . 144
6.3.1 Using Cockpit . . . 144
6.3.2 Features . . . 147
6.3.2.1 System Dashboard . . . 147
6.3.2.2 Viewing System Logs . . . 148
6.3.2.3 Managing Storage . . . 149
6.3.2.4 Networking . . . 150

6.3.2.5 Services . . . 151
6.3.2.6 Web Terminal . . . 152
6.3.3 Troubleshooting . . . 153
6.3.4 External links . . . 153
6.4 Configure . . . 153
6.4.1 Hostname . . . 153
6.4.2 Domain Name . . . 153
6.4.3 Webserver Home Page . . . 153
6.5 Date & Time . . . 154
6.6 Diagnostics . . . 154
6.7 Dynamic DNS Client . . . 154
6.7.1 What is Dynamic DNS? . . . 154
6.7.2 A Free Domain for your FreedomBox . . . 155
6.7.3 GnuDIP vs. Update URL . . . 155
6.7.4 Using the GnuDIP protocol . . . 155
6.7.5 Using an Update URL . . . 156
6.7.6 Checking If It Works . . . 157
6.7.7 Recap: How to create a DNS name with GnuDIP . . . 157
6.8 Firewall . . . 157
6.8.1 Interfaces . . . 158
6.8.2 Opening Custom Ports . . . 159
6.8.3 FreedomBox Ports/Services . . . 159
6.8.4 Manual operation . . . 160
6.8.4.1 Enable/disable firewall . . . 160
6.8.4.2 Modifying services/ports . . . 161
6.8.4.3 Modifying the zone of interfaces . . . 162
6.9 Let's Encrypt (Certificates) . . . 162
6.9.1 Why using Certificates . . . 162
6.9.2 How to setup . . . 163
6.9.3 Using . . . 164
6.9.4 External links . . . 164
6.10 Name Services . . . 165
6.10.1 Adding Domain Name to FreedomBox . . . 165
6.10.2 Subdomains . . . 165
6.10.3 systemd-resolved . . . 166
6.10.4 Support for DNS-over-TLS and DNSSEC . . . 166
6.10.5 Setting a custom DNS server . . . 166
6.10.6 Fallback DNS servers . . . 166
6.11 Networks . . . 166

6.11.1 Default setup . . . 166
6.11.1.1 Single ethernet interface . . . 167
6.11.1.2 Multiple ethernet interface . . . 167
6.11.1.3 Wi-Fi configuration . . . 168
6.11.2 Internet Connection Sharing . . . 168
6.11.3 Customization . . . 168
6.11.3.1 PPPoE connections . . . 168
6.11.3.2 Connect to Internet via Wi-Fi . . . 168
6.11.3.2.1 Problems with Privacy Feature . . . 168
6.11.3.3 Adding a new network device . . . 169
6.11.3.4 Configuring a mesh network . . . 169
6.11.3.4.1 Joining a mesh network . . . 169
6.11.3.4.2 Creating a mesh network . . . 170
6.11.4 Advanced Network Operations . . . 171
6.11.5 Manual Network Operation . . . 171
6.12 PageKite (Public Visibility) . . . 172
6.12.1 What is PageKite? . . . 172
6.12.2 Using PageKite . . . 172
6.13 Performance (System Monitoring) . . . 173
6.14 Power . . . 173
6.15 Secure Shell (SSH) Server . . . 173
6.15.1 What is Secure Shell? . . . 173
6.15.2 Setting Up A User Account . . . 173
6.15.2.1 FreedomBox First Log In: Admin Account . . . 173
6.15.2.2 Default User Account . . . 173
6.15.3 Logging In . . . 174
6.15.3.1 Who can log in to FreedomBox by SSH? . . . 174
6.15.3.2 SSH Client Software . . . 174
6.15.3.2.1 Cockpit as an SSH Alternative . . . 174
6.15.3.3 SSH over Local Network . . . 175
6.15.3.4 SSH over the Internet . . . 175
6.15.3.5 SSH over Tor . . . 175
6.15.3.6 SSH Over Pagekite . . . 176
6.15.4 Becoming Superuser . . . 176
6.15.5 Changing Password . . . 176
6.15.6 SSH Keys . . . 176
6.15.6.1 SSH Public and Private Keys . . . 177
6.15.6.2 Create your personal SSH keys on your client computer using ssh-keygen . . . 177
6.15.6.3 Verify your FreedomBox Server Fingerprint . . . 177

6.15.6.4 Share your personal SSH key with FreedomBox using ssh-copy-id . . . 178
6.15.6.5 Block SSH password guessing attempts by disabling password authentication . . . 179
6.15.6.6 Remote Host Identification has Changed : What it means and what to do . . . 180
6.15.7 External links . . . 180
6.16 Security . . . 181
6.16.1 Configuration . . . 181
6.17 Service Discovery . . . 181
6.17.1 Troubleshooting . . . 181
6.17.1.1 Unable to reach <hostname>.local . . . 181
6.18 Storage . . . 182
6.18.1 Advanced Storage Operations . . . 182
6.19 Storage Snapshots . . . 183
6.20 Software Updates . . . 184
6.20.1 When Will I Get the Latest Features? . . . 186
6.20.2 Manual Updates from Web Interface . . . 186
6.20.3 Manual Updates from Terminal . . . 186
6.20.4 Auto-Update to Next Stable Release . . . 187
6.20.5 Manual Update to Next Stable Release . . . 187
6.21 Users and Groups . . . 188

7 Guides **189**
7.1 Expose a Local Service to Internet using FreedomBox . . . 189
7.1.1 Goal . . . 189
7.1.2 Exposing Local LAN servers . . . 189
7.1.3 Running the Service on FreedomBox Itself . . . 190
7.1.4 Disabling the service . . . 190
7.1.5 Adding Authentication . . . 190
7.1.6 Hosting the Service on a Separate Domain . . . 191
7.1.7 Troubleshooting . . . 191
7.1.8 Extras . . . 192

8 Hardware **192**
8.1 Recommended Hardware . . . 192
8.1.1 Libre Crafts FreedomBox . . . 192
8.1.2 Olimex's FreedomBox Pioneer Edition . . . 193
8.2 Supported Hardware . . . 193
8.2.1 Hardware Comparison . . . 194
8.3 Additional Hardware . . . 195
8.3.1 Hardware Supported with Generic Images . . . 195

8.3.2 Adding Hardware Support . . . 195
8.3.3 Deprecated Hardware . . . 195
8.4 Common Hardware Information . . . 196
8.4.1 Wi-Fi . . . 196
8.4.2 Power Supply . . . 196
8.4.3 Firmware . . . 196
8.4.4 Storage . . . 197
8.4.4.1 Secure Digital (SD) Card . . . 197
8.4.4.2 Embedded MultiMediaCard (eMMC) . . . 197
8.4.4.3 USB Disk Drive . . . 197
8.4.4.4 SATA disk drive . . . 198
8.4.4.5 NVMe disk drive . . . 198
8.5 Building Your Own Images . . . 198
8.5.1 Status of Software Used . . . 198
8.6 Pioneer Edition FreedomBox . . . 199
8.6.1 Product Features . . . 200
8.6.2 Recommended Hardware . . . 200
8.6.3 Availability . . . 200
8.6.4 Hardware Specifications . . . 200
8.6.5 Storage Add-on . . . 201
8.6.6 Download . . . 201
8.6.7 Build Image . . . 202
8.6.8 Known Issues . . . 202
8.6.9 Obtaining Source Code . . . 202
8.6.9.1 From within FreedomBox . . . 202
8.6.9.2 Other Ways to Obtain Source Code . . . 203
8.6.9.3 Building Disk Images . . . 203
8.6.9.4 U-boot on Pioneer Edition Images . . . 203
8.6.10 Power and reset buttons . . . 203
8.7 A20 OLinuXino Lime2 . . . 204
8.7.1 Similar Hardware . . . 204
8.7.2 Download . . . 204
8.7.3 Availability . . . 205
8.7.4 Hardware . . . 205
8.7.5 Non-Free Status . . . 205
8.7.6 Known Issues . . . 205
8.8 A20 OLinuXino MICRO . . . 206
8.8.1 Similar Hardware . . . 206
8.8.2 Download . . . 206

8.8.3 Availability . . . 206
8.8.4 Hardware . . . 207
8.8.5 Non-Free Status . . . 207
8.8.6 Known Issues . . . 207
8.9 APU . . . 207
8.9.1 Similar Hardware . . . 208
8.9.2 Installation . . . 208
8.9.3 Networking . . . 208
8.9.4 Availability . . . 208
8.9.5 Hardware . . . 208
8.9.6 Non-Free Status . . . 209
8.10 Cubietruck . . . 209
8.10.1 FreedomBox Danube Edition . . . 209
8.10.2 Cubietruck / Cubieboard3 . . . 209
8.10.3 Download . . . 209
8.10.4 Availability . . . 209
8.10.5 Hardware . . . 210
8.10.6 Non-Free Status . . . 210
8.10.7 Known Issues . . . 210
8.11 Cubieboard 2 . . . 210
8.11.1 Download . . . 210
8.11.2 Availability . . . 211
8.11.3 Hardware . . . 211
8.11.4 Non-Free Status . . . 211
8.12 Beagle Bone Black . . . 211
8.12.1 Download . . . 212
8.12.2 Availability . . . 212
8.12.3 Hardware . . . 212
8.12.4 Non-Free Status . . . 212
8.13 pcDuino3 . . . 212
8.13.1 Similar Hardware . . . 213
8.13.2 Download . . . 213
8.13.3 Availability . . . 213
8.13.4 Hardware . . . 213
8.13.5 Non-Free Status . . . 213
8.14 Debian . . . 214
8.14.1 Installing on Debian 12 (bookworm) or newer . . . 214
8.14.2 Tips and Troubleshooting . . . 214
8.15 VirtualBox . . . 215

8.15.1 Download . . . 215
8.15.2 Creating a Virtual Machine . . . 216
8.15.3 First Boot . . . 219
8.15.4 Using . . . 221
8.15.5 Build Image . . . 221
8.15.6 Tips & Troubleshooting . . . 221
8.15.6.1 Network Configuration . . . 221
8.15.6.2 Finding out the IP address of the virtual machine . . . 222
8.15.6.3 Networking Problems with macchanger . . . 222
8.15.6.4 Mounting Images Locally . . . 222
8.15.6.5 Fixing the time after suspend and resume . . . 223
8.15.6.6 UUID collision in VB . . . 223
8.16 Pine A64+ . . . 223
8.16.1 Similar Hardware . . . 223
8.16.2 Download . . . 224
8.16.3 Availability . . . 224
8.16.4 Hardware . . . 224
8.16.5 Non-Free Status . . . 224
8.17 Banana Pro . . . 224
8.17.1 Download . . . 225
8.17.2 Hardware . . . 225
8.17.3 Non-Free Status . . . 225
8.18 Orange Pi Zero . . . 226
8.18.1 Download . . . 226
8.18.2 Availability . . . 226
8.18.3 Hardware . . . 226
8.18.4 Non-Free Status . . . 227
8.19 RockPro64 . . . 227
8.19.1 Download . . . 227
8.19.2 Availability . . . 227
8.19.3 Hardware . . . 227
8.19.4 Non-Free Status . . . 228
8.19.5 Known Issues . . . 228
8.20 Rock64 . . . 228
8.20.1 Download . . . 229
8.20.2 Availability . . . 229
8.20.3 Hardware . . . 229
8.20.4 Non-Free Status . . . 229
8.20.5 Known Issues . . . 229

8.21 Raspberry Pi 2 Model B . . . 230
8.21.1 Download . . . 230
8.21.2 Availability . . . 230
8.21.3 Hardware . . . 230
8.21.4 Non-Free Status . . . 231
8.22 Raspberry Pi 3 Model B . . . 231
8.22.1 Download . . . 231
8.22.2 Availability . . . 231
8.22.3 Hardware . . . 231
8.22.4 Non-Free Status . . . 232
8.23 Raspberry Pi 3 Model B+ . . . 232
8.23.1 Download . . . 232
8.23.2 Availability . . . 232
8.23.3 Hardware . . . 233
8.23.4 Non-Free Status . . . 233
8.24 Raspberry Pi 4 Model B . . . 233
8.24.1 Download . . . 233
8.24.2 Booting from USB . . . 234
8.24.3 Build Image . . . 234
8.24.4 Availability . . . 234
8.24.5 Hardware . . . 234
8.24.6 Non-Free Status . . . 235
8.25 USB Wi-Fi . . . 235
8.25.1 Firmware Installation . . . 235
8.25.2 Resources . . . 235

9 Debian Upgrade Notes **235**
9.1 Debian 13 "trixie" . . . 235
9.1.1 TT-RSS no longer supported . . . 236
9.1.2 Searx app no longer supported . . . 236

10 Release Notes **236**
10.1 FreedomBox 26.5 (2026-03-23) . . . 236
10.1.1 Highlights . . . 236
10.1.2 Other Changes . . . 236
10.2 FreedomBox 26.4.2 (2026-03-08) . . . 237
10.2.1 Highlights . . . 237
10.2.2 Other Changes . . . 237
10.3 FreedomBox 26.4 (2026-03-02) . . . 237

10.3.1 Highlights . . . 237
10.3.2 Other Changes . . . 238
10.4 FreedomBox 26.3 (2026-02-02) . . . 240
10.4.1 Highlights . . . 240
10.4.2 Other Changes . . . 240
10.5 FreedomBox 26.2 (2026-01-20) . . . 241
10.6 FreedomBox 26.1 (2026-01-05) . . . 241
10.7 FreedomBox 25.17.1 (2025-12-15) . . . 241
10.7.1 Highlights . . . 241
10.7.2 Other Changes . . . 241
10.8 FreedomBox 25.17 (2025-12-08) . . . 241
10.8.1 Highlights . . . 241
10.8.2 Other Changes . . . 241
10.9 FreedomBox 25.16 (2025-11-24) . . . 242
10.9.1 Highlights . . . 242
10.9.2 Other Changes . . . 242
10.10 FreedomBox 25.15 (2025-11-10) . . . 242
10.10.1 Highlights . . . 242
10.10.2 Other Changes . . . 242
10.11 FreedomBox 25.14 (2025-10-27) . . . 243
10.12 FreedomBox 25.13.1 (2025-10-20) . . . 243
10.13 FreedomBox 25.13 (2025-10-06) . . . 243
10.13.1 Highlights . . . 243
10.13.2 Other Changes . . . 243
10.14 FreedomBox 25.12 (2025-09-22) . . . 244
10.14.1 Highlights . . . 244
10.14.2 Other Changes . . . 244
10.15 FreedomBox 25.11 (2025-09-08) . . . 244
10.15.1 Highlights . . . 244
10.15.2 Other Changes . . . 245
10.16 FreedomBox 25.10 (2025-08-18) . . . 246
10.16.1 Highlights . . . 246
10.16.2 Other Changes . . . 246
10.17 FreedomBox 25.9.4 (2025-08-02) . . . 247
10.18 FreedomBox 25.9.3 (2025-07-21) . . . 248
10.19 FreedomBox 25.9.2 (2025-06-23) . . . 248
10.20 FreedomBox 25.9.1 (2025-06-02) . . . 248
10.21 FreedomBox 25.9 (2025-05-05) . . . 248
10.22 FreedomBox 25.8 (2025-04-21) . . . 248

10.22.1 Highlights . . . 248
10.22.2 Other Changes . . . 248
10.23 FreedomBox 25.7 (2025-04-07) . . . 249
10.23.1 Highlights . . . 249
10.23.2 Other Changes . . . 249
10.24 FreedomBox 25.6 (2025-03-25) . . . 250
10.24.1 Highlight . . . 250
10.24.2 Other Changes . . . 250
10.25 FreedomBox 25.5 (2025-03-10) . . . 250
10.25.1 Highlights . . . 250
10.25.2 Other Changes . . . 250
10.26 FreedomBox 25.4.1 (2025-03-02) . . . 251
10.27 FreedomBox 25.4 (2025-02-24) . . . 251
10.27.1 Highlights . . . 251
10.27.2 Other Changes . . . 251
10.28 FreedomBox 25.3.1 (2025-02-15) . . . 252
10.29 FreedomBox 25.3 (2025-02-10) . . . 252
10.30 FreedomBox 25.2 (2025-01-27) . . . 252
10.30.1 Highlights . . . 252
10.30.2 Other Changes . . . 252
10.31 FreedomBox 25.1 (2025-01-13) . . . 253
10.31.1 Highlights . . . 253
10.31.2 Other Changes . . . 253
10.32 FreedomBox 24.26.1 (2025-01-05) . . . 254
10.32.1 Highlights . . . 254
10.32.2 Other Changes . . . 254
10.33 FreedomBox 24.26 (2024-12-30) . . . 254
10.33.1 Highlights . . . 254
10.33.2 Other Changes . . . 254
10.34 FreedomBox 24.25 (2024-12-16) . . . 256
10.34.1 Highlights . . . 256
10.34.2 Other Changes . . . 256
10.35 FreedomBox 24.24 (2024-11-18) . . . 258
10.35.1 Highlights . . . 258
10.35.2 Other Changes . . . 258
10.36 FreedomBox 24.23 (2024-11-04) . . . 259
10.37 FreedomBox 24.22 (2024-10-21) . . . 259
10.37.1 Highlights . . . 259
10.37.2 Other Changes . . . 259

10.38 FreedomBox 24.21 (2024-10-07) . . . 261
10.38.1 Highlights . . . 261
10.38.2 Other Changes . . . 261
10.39 FreedomBox 24.20.1 (2024-09-25) . . . 261
10.39.1 Highlights . . . 261
10.39.2 Other Changes . . . 262
10.40 FreedomBox 24.20 (2024-09-23) . . . 262
10.40.1 Highlights . . . 262
10.40.2 Other Changes . . . 262
10.41 FreedomBox 24.19 (2024-09-09) . . . 262
10.41.1 Highlights . . . 262
10.41.2 Other Changes . . . 263
10.42 FreedomBox 24.18 (2024-08-26) . . . 263
10.43 FreedomBox 24.17 (2024-08-12) . . . 264
10.43.1 Highlights . . . 264
10.43.2 Other Changes . . . 264
10.44 FreedomBox 24.16 (2024-07-29) . . . 265
10.44.1 Highlights . . . 265
10.44.2 Other Changes . . . 265
10.45 FreedomBox 24.15 (2024-07-16) . . . 265
10.46 FreedomBox 24.14 (2024-07-01) . . . 266
10.47 FreedomBox 24.13 (2024-06-17) . . . 266
10.48 FreedomBox 24.12 (2024-06-03) . . . 266
10.49 FreedomBox 24.11 (2024-05-20) . . . 266
10.50 FreedomBox 24.10 (2024-05-06) . . . 266
10.50.1 Highlights . . . 266
10.50.2 Other Changes . . . 266
10.51 FreedomBox 24.9 (2024-04-22) . . . 267
10.52 FreedomBox 24.8 (2024-04-08) . . . 268
10.52.1 Highlights . . . 268
10.52.2 Other Changes . . . 268
10.53 FreedomBox 24.7 (2024-03-25) . . . 268
10.53.1 Highlights . . . 268
10.53.2 Other Changes . . . 269
10.54 FreedomBox 24.6 (2024-03-11) . . . 269
10.54.1 Highlights . . . 269
10.54.2 Other Changes . . . 270
10.55 FreedomBox 24.5 (2024-02-26) . . . 270
10.56 FreedomBox 24.4 (2024-02-12) . . . 271

10.56.1 Highlights . . . 271
10.56.2 Other Changes . . . 271
10.57 FreedomBox 24.3 (2024-01-29) . . . 271
10.57.1 Highlights . . . 271
10.57.2 Other Changes . . . 272
10.58 FreedomBox 24.2 (2024-01-15) . . . 272
10.59 FreedomBox 24.1 (2024-01-01) . . . 272
10.59.1 Highlights . . . 272
10.59.2 Other Changes . . . 272
10.60 FreedomBox 23.21 (2023-11-20) . . . 272
10.61 FreedomBox 23.20 (2023-11-06) . . . 272
10.61.1 Highlights . . . 272
10.61.2 Other Changes . . . 273
10.62 FreedomBox 23.19 (2023-10-23) . . . 273
10.62.1 Highlights . . . 273
10.62.2 Other Changes . . . 273
10.63 FreedomBox 23.18 (2023-09-25) . . . 274
10.64 FreedomBox 23.17 (2023-09-11) . . . 274
10.65 FreedomBox 23.16 (2023-08-28) . . . 274
10.65.1 Highlights . . . 274
10.65.2 Other Changes . . . 274
10.66 FreedomBox 23.15 (2023-08-14) . . . 274
10.67 FreedomBox 23.14 (2023-07-31) . . . 275
10.67.1 Highlights . . . 275
10.67.2 Other Changes . . . 275
10.68 FreedomBox 23.13 (2023-07-17) . . . 276
10.69 FreedomBox 23.12 (2023-06-19) . . . 276
10.69.1 Highlights . . . 276
10.69.2 Other Changes . . . 276
10.70 FreedomBox 23.11 (2023-06-05) . . . 276
10.70.1 Highlights . . . 276
10.70.2 Other Changes . . . 277
10.71 FreedomBox 23.10 (2023-05-22) . . . 278
10.72 FreedomBox 23.9 (2023-05-08) . . . 278
10.72.1 Highlights . . . 278
10.72.2 Other Changes . . . 278
10.73 FreedomBox 23.6.2 (2023-05-01) . . . 279
10.74 FreedomBox 23.8 (2023-04-24) . . . 279
10.74.1 Highlights . . . 279

10.74.2 Other Changes . . . 279
10.75 FreedomBox 23.6.1 (2023-04-10) . . . 280
10.76 FreedomBox 23.7 (2023-03-27) . . . 280
10.76.1 Highlights . . . 280
10.76.2 Other Changes . . . 280
10.77 FreedomBox 23.6 (2023-03-13) . . . 281
10.78 FreedomBox 23.5 (2023-02-27) . . . 281
10.79 FreedomBox 23.4 (2023-02-13) . . . 281
10.79.1 Highlights . . . 281
10.79.2 Other Changes . . . 281
10.80 FreedomBox 23.3 (2023-01-30) . . . 282
10.81 FreedomBox 23.2 (2023-01-16) . . . 282
10.82 FreedomBox 23.1 (2023-01-03) . . . 282
10.82.1 Highlights . . . 282
10.82.2 Other Changes . . . 282
10.83 FreedomBox 22.27 (2022-12-19) . . . 283
10.83.1 Highlights . . . 283
10.83.2 Other Changes . . . 283
10.84 FreedomBox 22.26 (2022-12-05) . . . 283
10.84.1 Highlights . . . 283
10.84.2 Other Changes . . . 283
10.85 FreedomBox 22.25 (2022-11-21) . . . 284
10.86 FreedomBox 22.24 (2022-11-07) . . . 284
10.86.1 Highlights . . . 284
10.86.2 Other Changes . . . 284
10.87 FreedomBox 22.23 (2022-10-24) . . . 284
10.87.1 Highlights . . . 284
10.87.2 Other Changes . . . 284
10.88 FreedomBox 22.22.1 (2022-10-16) . . . 284
10.89 FreedomBox 22.22 (2022-10-10) . . . 284
10.89.1 Highlights . . . 284
10.89.2 Other Changes . . . 285
10.90 FreedomBox 22.21.1 (2022-10-01) . . . 285
10.91 FreedomBox 22.21 (2022-09-26) . . . 285
10.92 FreedomBox 22.20 (2022-09-12) . . . 286
10.92.1 Highlights . . . 286
10.92.2 Other Changes . . . 286
10.93 FreedomBox 22.19 (2022-08-29) . . . 287
10.93.1 Highlights . . . 287

10.93.2 Other Changes . . . 287
10.94 FreedomBox 22.18 (2022-08-15) . . . 288
10.94.1 Highlights . . . 288
10.94.2 Other Changes . . . 288
10.95 FreedomBox 22.17 (2022-08-01) . . . 289
10.95.1 Highlights . . . 289
10.95.2 Other Changes . . . 289
10.96 FreedomBox 22.16 (2022-07-18) . . . 289
10.96.1 Highlights . . . 289
10.96.2 Other Changes . . . 289
10.97 FreedomBox 22.15 (2022-07-04) . . . 290
10.97.1 Highlights . . . 290
10.97.2 Other Changes . . . 290
10.98 FreedomBox 22.14.1 (2022-06-27) . . . 290
10.98.1 Highlights . . . 290
10.98.2 Other Changes . . . 290
10.99 FreedomBox 22.14 (2022-06-20) . . . 291
10.99.1 Highlights . . . 291
10.99.2 Other Changes . . . 291
10.100 FreedomBox 22.13 (2022-06-06) . . . 291
10.100.1 Highlights . . . 291
10.100.2 Other Changes . . . 291
10.101 FreedomBox 22.12 (2022-05-23) . . . 292
10.101.1 Highlights . . . 292
10.101.2 Other Changes . . . 292
10.102 FreedomBox 22.11 (2022-05-09) . . . 292
10.102.1 Highlights . . . 292
10.102.2 Other Changes . . . 292
10.103 FreedomBox 22.10 (2022-04-25) . . . 292
10.104 FreedomBox 22.9 (2022-04-11) . . . 293
10.104.1 Highlights . . . 293
10.104.2 Other Changes . . . 293
10.105 FreedomBox 22.8 (2022-03-28) . . . 293
10.105.1 Highlights . . . 293
10.105.2 Other Changes . . . 293
10.106 FreedomBox 22.7 (2022-03-14) . . . 293
10.107 FreedomBox 22.6.1 (2022-03-06) . . . 293
10.108 FreedomBox 22.6 (2022-03-02) . . . 293
10.108.1 Highlights . . . 293

10.108.2 Other Changes . . . 294
10.109 FreedomBox 22.5 (2022-02-14) . . . 295
10.109.1 Highlights . . . 295
10.109.2 Other Changes . . . 295
10.110 FreedomBox 22.4 (2022-01-31) . . . 295
10.110.1 Highlights . . . 295
10.110.2 Other Changes . . . 295
10.111 FreedomBox 22.3 (2022-01-17) . . . 296
10.111.1 Highlights . . . 296
10.111.2 Other Changes . . . 296
10.112 FreedomBox 22.2 (2022-01-11) . . . 296
10.112.1 Highlights . . . 296
10.112.2 Other Changes . . . 296
10.113 FreedomBox 22.1 (2022-01-03) . . . 297
10.113.1 Highlights . . . 297
10.113.2 Other Changes . . . 297
10.114 FreedomBox 21.16 (2021-12-20) . . . 297
10.114.1 Highlights . . . 297
10.114.2 Other Changes . . . 297
10.115 FreedomBox 21.15 (2021-12-06) . . . 298
10.115.1 Highlights . . . 298
10.115.2 Other Changes . . . 298
10.116 FreedomBox 21.14.1 (2021-11-24) . . . 299
10.117 FreedomBox 21.14 (2021-11-22) . . . 299
10.117.1 Highlights . . . 299
10.117.2 Other Changes . . . 299
10.118 FreedomBox 21.13 (2021-11-08) . . . 300
10.118.1 Highlights . . . 300
10.118.2 Other Changes . . . 300
10.119 FreedomBox 21.12 (2021-10-25) . . . 300
10.120 FreedomBox 21.11 (2021-10-11) . . . 300
10.120.1 Highlights . . . 300
10.120.2 Other Changes . . . 300
10.121 FreedomBox 21.10 (2021-09-27) . . . 301
10.121.1 Highlights . . . 301
10.121.2 Other Changes . . . 301
10.122 FreedomBox 21.9 (2021-09-18) . . . 302
10.122.1 Highlights . . . 302
10.122.2 Other Changes . . . 302

10.123 FreedomBox 21.8 (2021-08-30) . . . 302
10.123.1 Highlights . . . 302
10.123.2 Other Changes . . . 302
10.124 FreedomBox 21.7 (2021-08-16) . . . 303
10.124.1 Highlights . . . 303
10.124.2 Other Changes . . . 303
10.125 FreedomBox 21.6 (2021-05-31) . . . 303
10.125.1 Highlights . . . 303
10.125.2 Other Changes . . . 303
10.126 FreedomBox 21.5 (2021-04-19) . . . 303
10.126.1 Highlights . . . 303
10.126.2 Other Changes . . . 303
10.127 FreedomBox 21.4.2 (2021-03-28) . . . 304
10.127.1 Highlights . . . 304
10.127.2 Other Changes . . . 304
10.128 FreedomBox 21.4.1 (2021-03-13) . . . 305
10.128.1 Highlights . . . 305
10.128.2 Other Changes . . . 305
10.129 FreedomBox 21.4 (2021-02-28) . . . 305
10.129.1 Highlights . . . 305
10.129.2 Other Changes . . . 305
10.130 FreedomBox 21.3 (2021-02-11) . . . 305
10.130.1 Highlights . . . 305
10.130.2 Other Changes . . . 305
10.131 FreedomBox 21.2 (2021-02-05) . . . 306
10.131.1 Highlights . . . 306
10.131.2 Other Changes . . . 306
10.132 FreedomBox 21.1 (2021-01-25) . . . 306
10.132.1 Highlights . . . 306
10.132.2 Other Changes . . . 306
10.133 FreedomBox 21.0 (2021-01-11) . . . 307
10.133.1 Highlights . . . 307
10.133.2 Other Changes . . . 307
10.134 FreedomBox 20.21 (2020-12-28) . . . 308
10.134.1 Highlights . . . 308
10.134.2 Other Changes . . . 308
10.135 FreedomBox 20.20.1 (2020-12-19) . . . 308
10.135.1 Highlights . . . 308
10.135.2 Other Changes . . . 308

10.136 FreedomBox 20.20 (2020-12-14) . . . 309
10.136.1 Highlights . . . 309
10.136.2 Other Changes . . . 309
10.137 FreedomBox 20.19 (2020-11-30) . . . 309
10.137.1 Highlights . . . 309
10.137.2 Other Changes . . . 309
10.138 FreedomBox 20.18.1 (2020-11-23) . . . 309
10.139 FreedomBox 20.18 (2020-11-16) . . . 310
10.139.1 Highlights . . . 310
10.139.2 Other Changes . . . 310
10.140 FreedomBox 20.17.1 (2020-11-07) . . . 310
10.141 FreedomBox 20.17 (2020-11-02) . . . 310
10.141.1 Highlights . . . 310
10.141.2 Other Changes . . . 310
10.142 FreedomBox 20.16 (2020-10-19) . . . 311
10.142.1 Highlights . . . 311
10.142.2 Other Changes . . . 311
10.143 FreedomBox 20.15 (2020-10-05) . . . 311
10.143.1 Highlights . . . 311
10.143.2 Other Changes . . . 311
10.144 FreedomBox 20.14.1 (2020-09-23) . . . 312
10.145 FreedomBox 20.14 (2020-09-15) . . . 312
10.145.1 Highlights . . . 312
10.145.2 Other Changes . . . 312
10.146 FreedomBox 20.13 (2020-07-18) . . . 313
10.146.1 Highlights . . . 313
10.146.2 Other Changes . . . 313
10.147 FreedomBox 20.12.1 (2020-07-05) . . . 314
10.148 FreedomBox 20.12 (2020-06-29) . . . 314
10.148.1 Highlights . . . 314
10.148.2 Other Changes . . . 314
10.149 FreedomBox 20.11 (2020-06-15) . . . 314
10.149.1 Top Highlight . . . 314
10.149.2 Other Changes . . . 314
10.150 FreedomBox 20.10 (2020-06-01) . . . 315
10.150.1 Top Highlights . . . 315
10.150.2 Other Changes . . . 315
10.151 FreedomBox 20.9 (2020-05-18) . . . 315
10.151.1 Top Highlights . . . 315

10.151.2 Other Changes . . . 315
10.152 FreedomBox 20.8 (2020-05-04) . . . 316
10.153 FreedomBox 20.7 (2020-04-20) . . . 316
10.154 FreedomBox 20.6.1 (2020-04-11) . . . 317
10.155 FreedomBox 20.6 (2020-04-06) . . . 317
10.156 FreedomBox 20.5.1 (2020-03-26) . . . 317
10.157 FreedomBox 20.5 (2020-03-23) . . . 317
10.158 FreedomBox 20.4 (2020-03-09) . . . 318
10.159 FreedomBox 20.3 (2020-02-24) . . . 319
10.160 FreedomBox 20.2 (2020-02-10) . . . 319
10.161 FreedomBox 20.1 (2020-01-27) . . . 320
10.162 FreedomBox 20.0 (2020-01-13) . . . 320
10.163 FreedomBox 19.24 (2019-12-30) . . . 321
10.164 FreedomBox 19.23 (2019-12-16) . . . 321
10.165 FreedomBox 19.22 (2019-12-02) . . . 321
10.166 FreedomBox 19.21 (2019-11-18) . . . 322
10.167 FreedomBox 19.20 (2019-11-04) . . . 323
10.168 FreedomBox 19.19 (2019-10-21) . . . 323
10.169 FreedomBox 19.18 (2019-10-07) . . . 323
10.170 FreedomBox 19.17 (2019-09-23) . . . 324
10.171 FreedomBox 19.16 (2019-09-09) . . . 324
10.172 FreedomBox 19.15 (2019-08-26) . . . 324
10.173 FreedomBox 19.14 (2019-08-12) . . . 325
10.174 FreedomBox 19.13 (2019-07-29) . . . 325
10.175 FreedomBox 19.12 (2019-07-22) . . . 325
10.176 FreedomBox 19.2.2 (2019-07-17) . . . 325
10.177 FreedomBox 19.2.1 (2019-07-09) . . . 326
10.178 FreedomBox 19.11 (2019-07-08) . . . 326
10.179 FreedomBox 19.10 (2019-06-24) . . . 326
10.180 FreedomBox 19.9 (2019-06-10) . . . 326
10.181 FreedomBox 19.8 (2019-05-27) . . . 326
10.182 FreedomBox 19.7 (2019-05-13) . . . 327
10.183 FreedomBox 19.6 (2019-04-29) . . . 327
10.184 FreedomBox 19.5 (2019-04-15) . . . 327
10.185 FreedomBox 19.4 (2019-04-01) . . . 327
10.186 FreedomBox 19.3 (2019-03-18) . . . 327
10.187 FreedomBox 19.2 (2019-03-02) . . . 328
10.188 FreedomBox 19.1 (2019-02-14) . . . 328
10.189 FreedomBox 19.0 (2019-02-09) . . . 329

10.190 Version 0.49.1 (2019-02-07) . . . 329
10.191 Version 0.49.0 (2019-02-05) . . . 329
10.192 Version 0.48.0 (2019-01-28) . . . 330
10.193 Version 0.47.0 (2019-01-14) . . . 331
10.194 Version 0.46.0 (2018-12-31) . . . 331
10.195 Version 0.45.0 (2018-12-17) . . . 331
10.196 Version 0.44.0 (2018-12-03) . . . 331
10.197 Version 0.43.0 (2018-11-19) . . . 332
10.198 Version 0.42.0 (2018-11-05) . . . 332
10.199 Version 0.41.0 (2018-10-22) . . . 332
10.200 Version 0.40.0 (2018-10-08) . . . 332
10.201 Version 0.39.0 (2018-09-24) . . . 333
10.202 Version 0.38.0 (2018-09-10) . . . 333
10.203 Version 0.37.0 (2018-08-27) . . . 333
10.204 Version 0.36.0 (2018-08-13) . . . 333
10.205 Version 0.35.0 (2018-07-30) . . . 334
10.206 Version 0.34.0 (2018-07-16) . . . 334
10.207 Version 0.33.1 (2018-07-04) . . . 334
10.208 Version 0.33.0 (2018-07-02) . . . 334
10.209 Version 0.32.0 (2018-06-18) . . . 335
10.210 Version 0.31.0 (2018-06-04) . . . 335
10.211 Version 0.30.0 (2018-05-21) . . . 335
10.212 Version 0.29.1 (2018-05-08) . . . 335
10.213 Version 0.29.0 (2018-05-07) . . . 335
10.214 Version 0.28.0 (2018-04-23) . . . 335
10.215 Version 0.27.0 (2018-04-09) . . . 336
10.216 Version 0.26.0 (2018-03-26) . . . 336
10.217 Version 0.25.0 (2018-03-12) . . . 336
10.218 Plinth v0.24.0 (2018-02-26) . . . 336
10.219 Plinth v0.23.0 (2018-02-12) . . . 337
10.220 Plinth v0.22.0 (2018-01-30) . . . 337
10.221 Plinth v0.21.0 (2018-01-15) . . . 337
10.222 Plinth v0.20.0 (2018-01-01) . . . 337
10.223 Plinth v0.19.0 (2017-12-18) . . . 338
10.224 Plinth v0.18.0 (2017-12-04) . . . 338
10.225 Plinth v0.17.0 (2017-11-20) . . . 338
10.226 Plinth v0.16.0 (2017-11-06) . . . 338
10.226.1 Added . . . 338
10.226.2 Fixed . . . 338

10.227 Plinth v0.15.3 (2017-10-20) . . . 339
10.227.1 Changed . . . 339
10.227.2 Added . . . 339
10.227.3 Fixed . . . 339
10.228 Plinth v0.15.2 (2017-09-24) . . . 339
10.228.1 Added . . . 339
10.228.2 Removed . . . 340
10.228.3 Fixed . . . 340
10.228.4 Changed . . . 340
10.229 Plinth v0.15.0 (2017-07-01) . . . 340
10.230 Plinth v0.14.0 (2017-04) . . . 341
10.231 Plinth v0.13.1 (2017-01-22) . . . 341
10.232 Plinth v0.12.0 (2016-12-08) . . . 342
10.233 Plinth v0.11.0 (2016-09-29) . . . 342
10.234 Plinth v0.10.0 (2016-08-21) . . . 343
10.235 Version 0.9.4 (2016-06-24) . . . 343
10.236 Version 0.9 (2016-04-24) . . . 344
10.237 Version 0.8 (2016-02-20) . . . 344
10.238 Version 0.7 (2015-12-13) . . . 344
10.239 Version 0.6 (2015-10-31) . . . 344
10.240 Version 0.5 (2015-08-07) . . . 345
10.241 Version 0.3 (2015-01-20) . . . 345
10.242 Version 0.2 (2014-03-16) . . . 345
10.243 Version 0.1 (2013-02-26) . . . 346

11 Customizing **346**

11.1 Change Default App . . . 346
11.2 Custom Shortcuts . . . 346
11.3 Custom Styling . . . 348

12 Contributing **349**

12.1 Get Involved . . . 349
12.1.1 Quick Links . . . 349
12.1.2 Welcome to newcomers . . . 350
12.2 Contributions needed . . . 350
12.2.1 Donate . . . 350
12.2.2 Spread the Word . . . 350
12.2.3 Feed Us Back (Comment) . . . 350
12.2.4 Request applications . . . 350

12.2.5 Translate . . . 351
12.2.6 Document: User Manual, Website and Wiki, HowTo/demo videos . . . 351
12.2.7 Assure Quality (Test and Check) . . . 351
12.2.8 Code . . . 351
12.2.8.1 Fix Bugs . . . 352
12.2.8.2 Add an Application . . . 352
12.2.8.3 Development priorities . . . 352
12.2.9 Design . . . 352
12.2.9.1 User Experience Design . . . 352
12.2.9.2 Technical Design . . . 352
12.2.10 Package Applications . . . 352

13 Developer Guide **352**

14 Hacking **353**
14.1 FreedomBox Service (Plinth) . . . 353
14.1.1 Using . . . 353
14.1.2 Screenshots . . . 353
14.1.3 Support . . . 354
14.1.4 Contributing . . . 354
14.1.4.1 Debian Package . . . 354
14.2 Freedom Maker . . . 354
14.2.1 Building FreedomBox Images . . . 355
14.2.2 Support . . . 355
14.2.3 Contributing . . . 355

15 Tell people around you **355**

1 FreedomBox: take your online privacy back

FreedomBox is a ready made personal server, designed with privacy and data ownership in mind. It is a subset of the Debian universal operating system and includes free software only. You can run it on a small, inexpensive and power-efficient computer box in your home that is dedicated for that use. It can also be installed on any computer running Debian or in a virtual machine.

In order to replace third-party communication services that are data mining your entire life, you will be able to host services yourself and use them at home or over the Internet through a browser or specialized apps. These services include chat and voice calls, webmail, file sharing and calendar, address book and news feed synchronization. For example, to start using a private chat service, activate the service from the administration interface and add your friends as authorized users of the service. They will be able to connect to the service hosted on your FreedomBox, using XMPP chat clients such as Conversations on Android, Pidgin on Windows and Linux, or Messages on Mac OS, for encrypted communications.

FreedomBox is a product you can just buy, set up and use. Once installed the interface is easy to use, similar to a smart phone.

User documentation:

- List of applications offered by FreedomBox.
- Manual
- Live Help from the community

FreedomBox can also host a Wi-Fi access point, ad blocking proxy and a virtual private network (VPN). More advanced users can replace their router with a FreedomBox.

Setting up FreedomBox on a specific hardware or on your computer running Debian may require a bit of technical expertise or help from the community.

Related technical documentation:

- Machines that support FreedomBox
- Download and Install
- FreedomBox Developer Manual

1.1 Typical usage: Private Cloud

FreedomBox provides services to the computers and mobile devices in your home, and to your friends. This includes secure instant messaging and low-bandwidth, high-quality voice conference calling. FreedomBox lets you publish your content in a blog and wiki to collaborate with the rest of the world. On the roadmap are a personal email server and federated social networking, to provide privacy-respecting alternatives to Gmail and Facebook.

1.2 Typical usage: Network-Attached Storage (NAS)

The storage space available to FreedomBox can be expanded by attaching an external disk drive. This allows FreedomBox to become a media library for your photos, music, and videos. The folders are shared to laptops and mobile phones on the local network, and the media can be streamed to local devices including smart TVs.

1.3 Advanced usage: Smart Home Router

FreedomBox runs in a physical computer and can route your traffic. It can sit between various devices at home such as mobiles, laptops and TVs and the Internet, replacing a home wireless router. By routing traffic, FreedomBox can remove tracking advertisements and malicious web bugs before they ever reach your devices. FreedomBox can cloak your location and protect your anonymity by "onion routing" your traffic over Tor. FreedomBox provides a VPN server that you can use while you are away from home to keep your traffic secret on untrusted public wireless networks and to securely access various devices at home.

It can also be carried along with your laptop and set up to offer its services on public networks at work, school or office. In the future, FreedomBox intends to deliver support for alternative ways of connecting to the Internet such as Mesh networking.

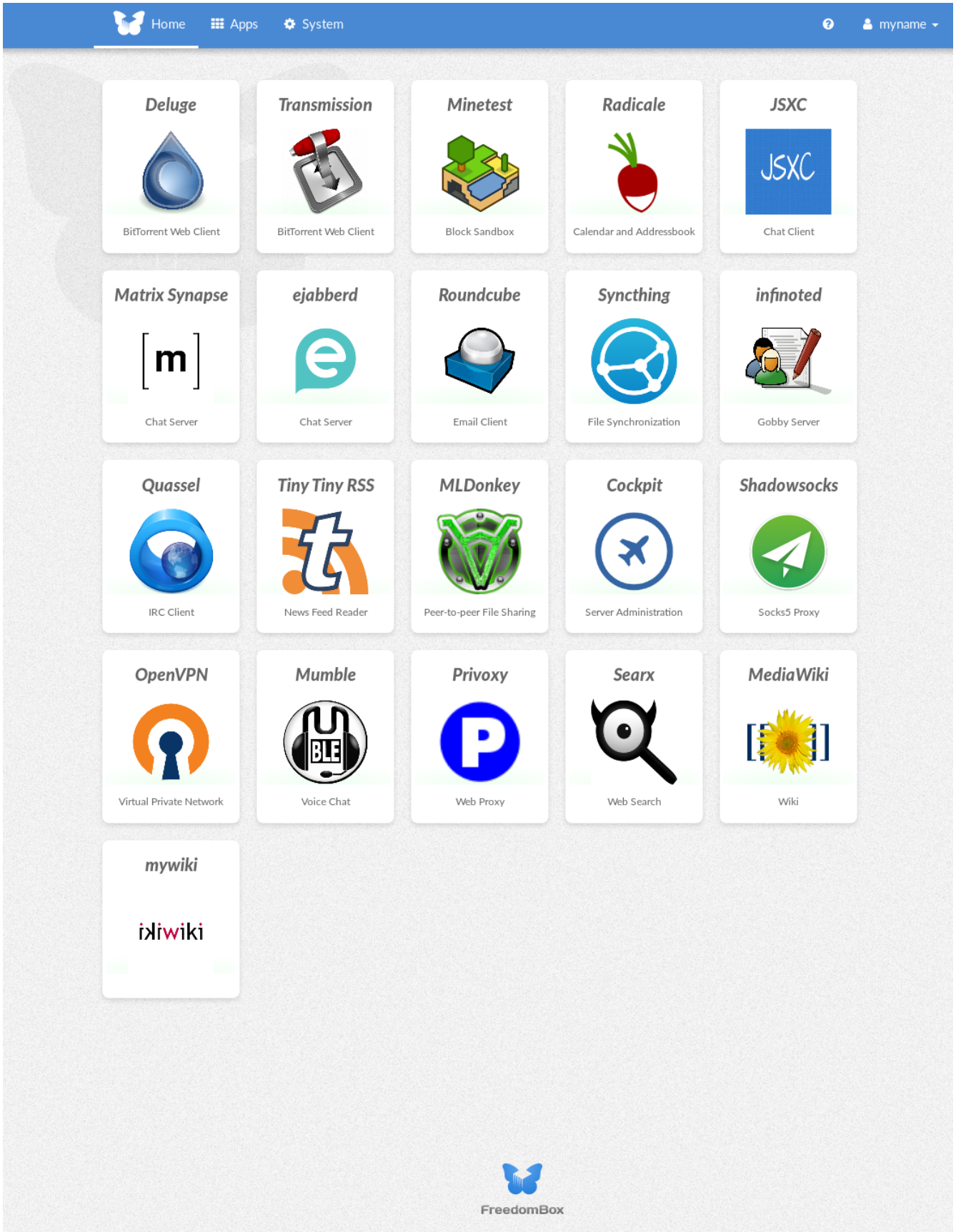
1.4 Advanced usage: For Communities

The primary design goal of FreedomBox is to be used as a personal server at home for use by a single family and their friends. However, at the core, it is a server software that can aid a non-technical user to setup services and maintain them with ease. Security is automatically managed and many of the technical choices in system administration are taken care by the software automatically thereby reducing complexity for a non-technical user. This nature of FreedomBox makes it well-suited for hosting services for small communities like villages or small firms. Communities can host their own services using FreedomBox with minimal effort. They can setup Wi-Fi networks that span the entire area of the community and draw Internet connections from long distances. Community members can enjoy previously unavailable Internet connectivity, ubiquitous Wi-Fi coverage, free VOIP services, offline education and entertainment content, etc. This will also boost privacy for individuals in the community, reduce dependence on centralized services provided by large companies and make them resistant to censorship.

The free e-book FreedomBox for Communities describes the motivation and provides detailed instructions to setup FreedomBox for this use case. Members of the FreedomBox project are involved in setting up Wi-Fi networks with free Internet connectivity in rural India. This e-book documents their knowledge and experiences.

1.5 FreedomBox Interface

1.5.1 Screenshot



1.5.2 Video resources

Eben Moglen's talk, Eben Moglen - Freedom in the cloud, delivered before the FreedomBox project was started gives insights into the philosophy behind FreedomBox.

First demonstration of FreedomBox at SFLC, University of Columbia by Sunil Mohan Adapa.

2 Quick Start

2.1 What you need to get started

The easy way is to buy a FreedomBox kit.

Alternatively you may choose to build it yourself, by gathering all the components:

- A supported device (including any device that can run Debian). We will call that the FreedomBox in the rest of this manual.
- A power cable for your device.
- An Ethernet cable.
- A microSD card (or equivalent storage media for your device), prepared according to the instructions on the Download page.

2.2 How to get started

1. Plug one end of your Ethernet cord into your FreedomBox's Ethernet port, and plug the other end into your router.
2. Power on the FreedomBox.
   - **Note:** On most single board computers, don't expect any output on a monitor connected via HDMI as the support may not exist in the kernel. See below to access and control your FreedomBox via network.
3. On first boot, FreedomBox will perform its initial setup (older versions of FreedomBox reboot after this step). This process may take several minutes on some machines. After giving it about 10 minutes, proceed to the next step.
4. After the FreedomBox has finished its initial setup, you can access its web interface through your web browser.
   - If your computer supports mDNS (GNU/Linux, Mac OSX or Windows with mDNS software installed), you can browse to: `http://freedombox.local/` (or `http://the-hostname-you-entered-during-install.local/`)
   - If your computer is connected to a monitor and FreedomBox is able to display text to it, then the login prompt shows the web address. It looks like this:

```
Debian GNU/Linux 13 freedombox tty1

FreedomBox

To start using FreedomBox and to create an account, access using a web browser
one of the following URLs.

http://freedombox.local/
http://192.168.0.2/
http://[fe80::7207:12ff:fe34:5678]/

...

freedombox login:
```

   - If your computer is connected directly to the FreedomBox through a second (LAN) Ethernet port, you can browse to: `http://freedombox/` or `http://10.42.0.1/`.

- If you know your way around the router's web interface, you can look up the IP address of the FreedomBox there, and browse to that address.
- If none of these methods are available, then you will need to figure out the IP address of your FreedomBox. You can use the "nmap" program from your computer to find its IP address:

```
nmap -p 80 --open -sV 192.168.0.0/24 (replace the ip/netmask with the one the ↩
    router uses)
```

In most cases you can look at your current IP address, and change the last digits with zero to find your home network, like so: XXX.XXX.XXX.0/24

Your FreedomBox will show up as an IP address with an open tcp port 80 using Apache httpd service on Debian, such as the example below which would make it accessible at http://192.168.0.165:

```
Nmap scan report for 192.168.0.165
Host is up (0.00088s latency).
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.17 ((Debian))
```

If nmap does not find anything with the above command, you can try replacing 192.168.0.0/24 with 10.42.0.255/24.

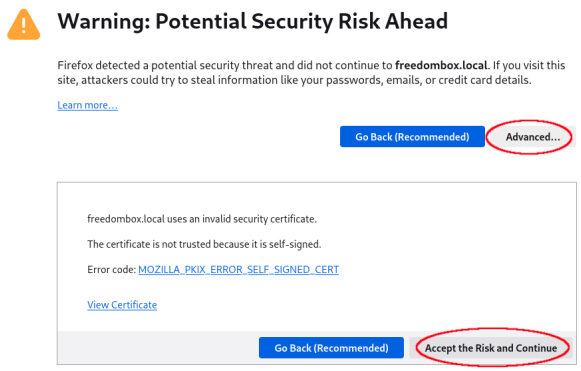
```
nmap -n -sP 10.42.0.255/24
```

The scan report will show something similar to the following:

```
Nmap scan report for 10.42.0.1
Host is up (0.00027s latency).
Nmap scan report for 10.42.0.50
Host is up (0.00044s latency).
```

In this example, the FreedomBox is accessible at http://10.42.0.50. (10.42.0.1 is my laptop.)

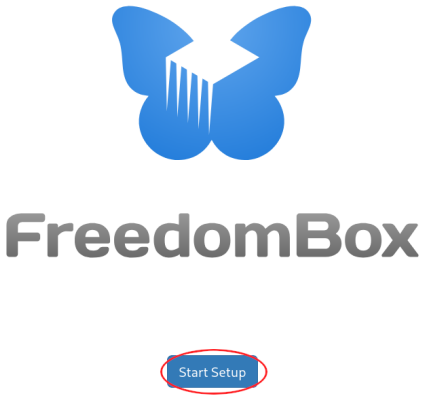
5. On accessing FreedomBox's web interface your browser will warn you that it communicates securely but that it regards the security certificate for doing so as invalid. This is a fact you need to accept because the certificate is auto generated on the box and therefore "self-signed" (the browser might also use words such as "untrusted", "not private", "privacy error" or "unknown issuer/authority"). Telling your browser that you are aware of this might involve pressing buttons such as "I understand the Risks", "proceed to ... (unsafe)" or "Add exception". After installation this certificate can be changed to a normal one using the Let's Encrypt option.



-
  - If the domain name you are using already has a valid certificate from a recognised Certificate Authority, such as from Let's Encrypt, you may not be able to access the web interface. This means that the option to continue will not be offered. This might occur, for example, if you are reinstalling your FreedomBox or are otherwise reusing a domain name that has an associated certificate.
    This is normal, because your browser has a record of having visited the site and knows about the valid official certificate. The browser will not subsequently accept a self-signed certificate and you will be blocked. There are two ways to get beyond this hurdle.

The first method is to access your FreedomBox by its IP address instead of by its domain name. So, instead of using something `https://example.com`, you would use something like `https://198.51.100.2` (substitute your own IP address).
The second method is to create a fresh profile for your browser and access your FreedomBox from the new profile. The new profile will have no memory of having visited the site nor memory of an official certificate. Once your FreedomBox has a new Let's Encrypt certificate, you can go back to using the old browser profile.

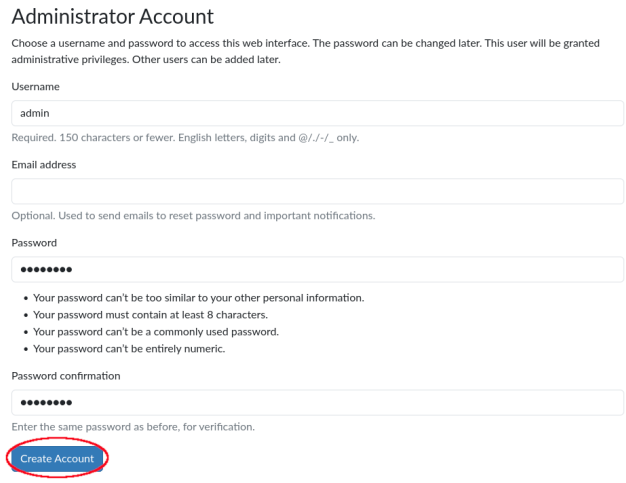
6. The first time you access the FreedomBox web interface, you will see a welcome page. Click the "Start Setup" button to continue.



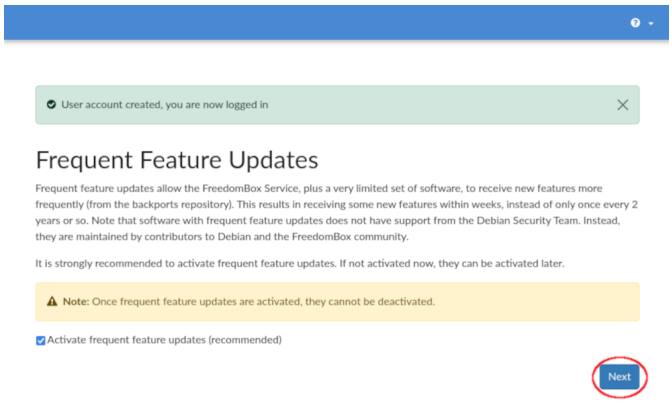
- If you have installed FreedomBox using a Debian package, you will be asked for a secret key. This secret was generated during the installation of the Debian package. It can be read from the file `/var/lib/plinth/firstboot-wizard-secr`

7. The next page asks you to provide a user name and password. Fill in the form, and then click "Create Account."

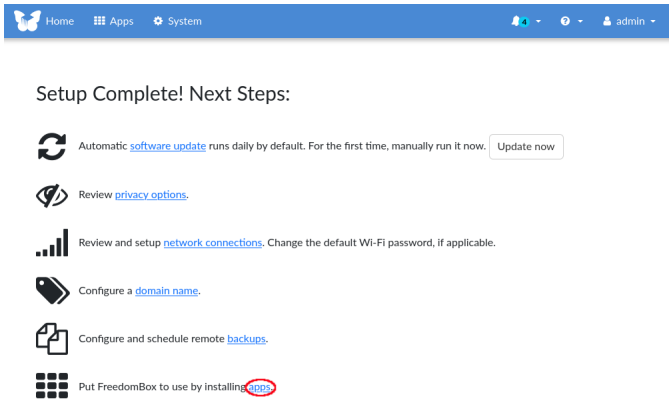
- Note: The user that you create here has Admin privileges and can also log in using ssh. For additional security, you may want to use a separate account for administrative tasks and for your normal, daily use. You can add more users later.

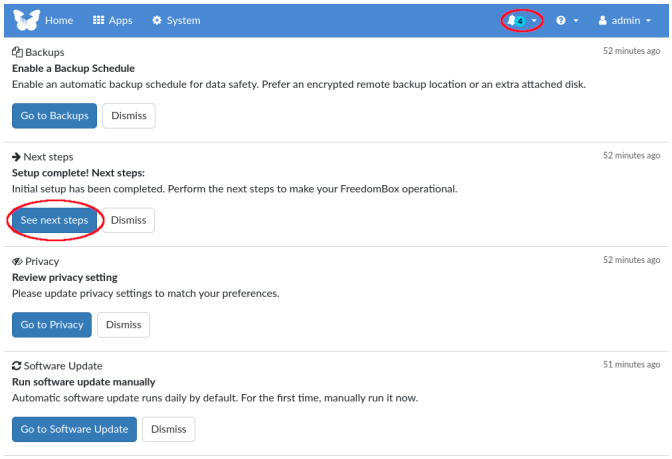


8. After completing the form, you will be logged in to FreedomBox's web interface and asked if you wish to receive more frequent updates (recommended).



9. Setup is complete. Now you can examine and possibly update several key configuration items or see which apps are available. Until you dismiss them, notifications accessible from the menu bar will remind you to review the key configuration items.





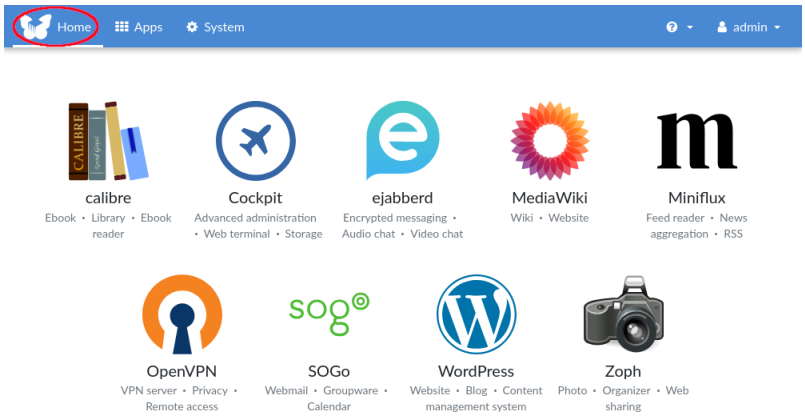
Now you can try any of the Apps that are available on FreedomBox.

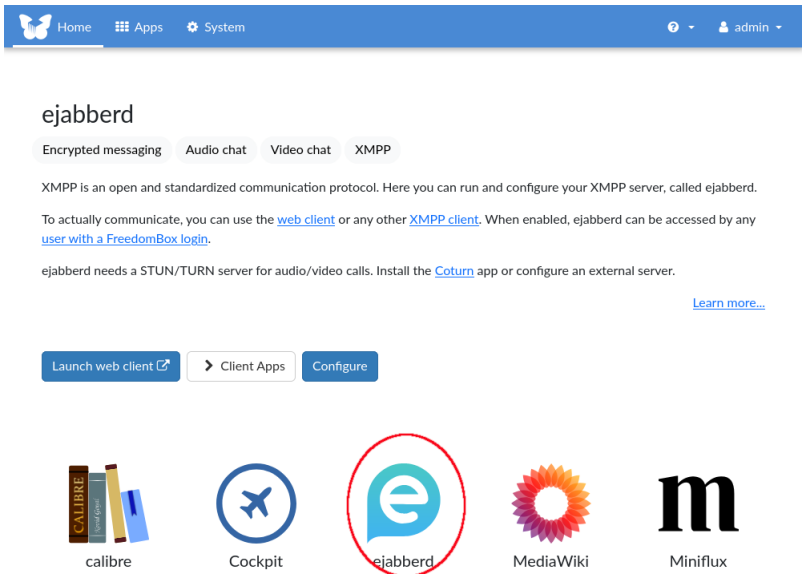
2.3 Finding your way around

2.3.1 Front page

The front page is the page that you will see when accessing the web root of your FreedomBox. You can also access it by clicking the FreedomBox logo in the top-left corner of the FreedomBox's web interface.

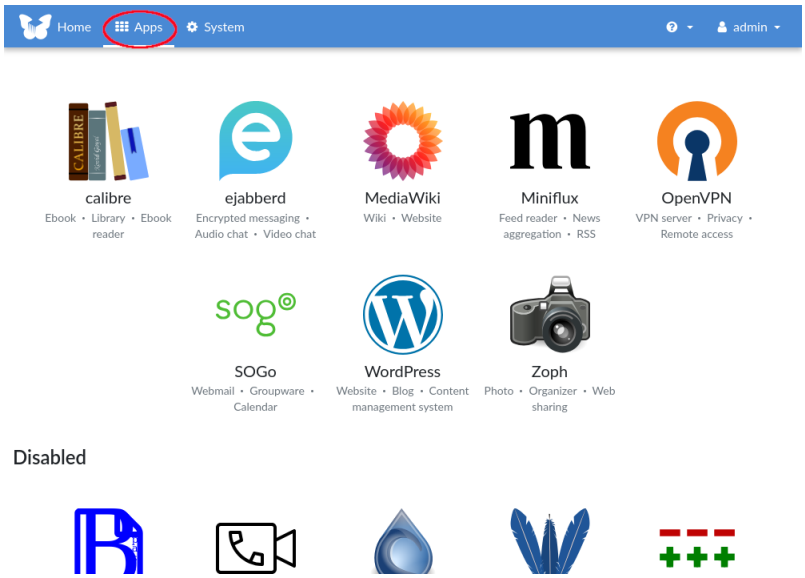
The front page includes shortcuts to apps that have been installed and are enabled. For web apps, clicking the shortcut will take you directly to the app's web page. For other services, clicking the shortcut will show more information about the service.





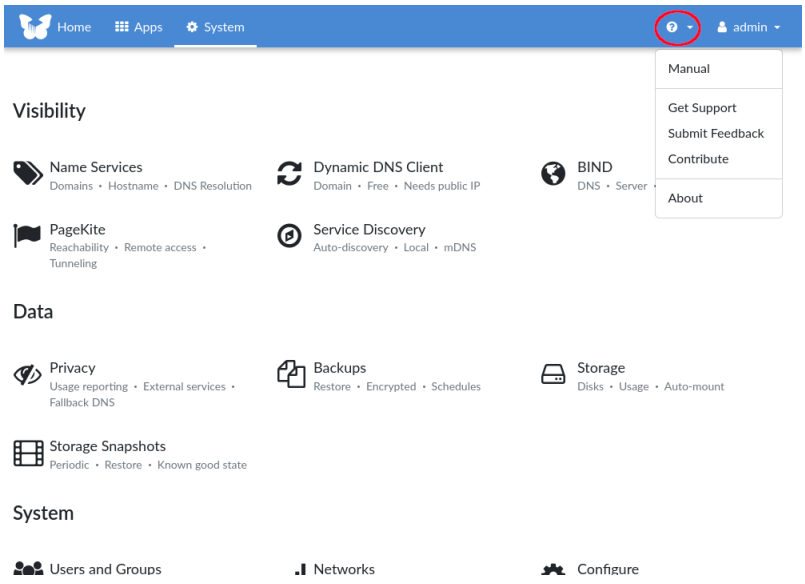
2.3.2 Apps menu

The Apps menu can be accessed by clicking the grid icon, next to the FreedomBox logo. This page lists all of the apps that are available for installing on FreedomBox. Click the name of an app to visit its page, where you can install and configure it. Currently installed and enabled apps are shown above the remaining available apps.



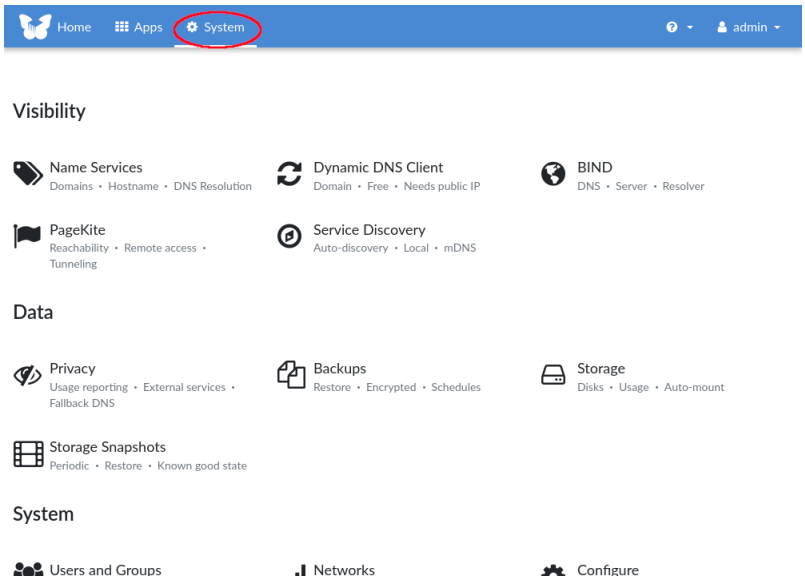
2.3.3 Help menu

A drop-down Help menu can be accessed by clicking the question mark icon near the top-right corner. The menu includes helpful links and the FreedomBox manual.



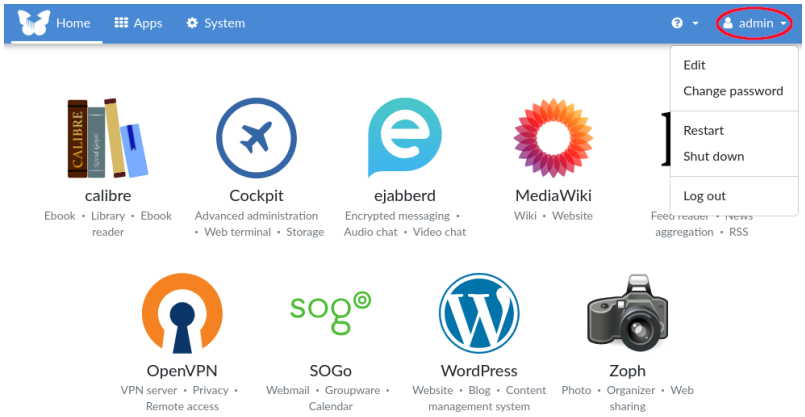
2.3.4 System menu

The System menu can be accessed by clicking the gear icon in the top-left corner. It includes a number of pages related to system configuration.



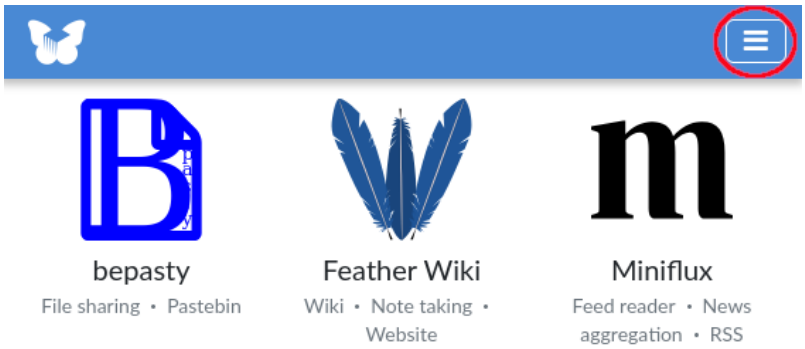
2.3.5 User menu

In the top-right corner, the name of the currently logged-in user is shown. A drop-down menu includes options for editing the current user or logging out of the user interface.



2.3.6 Burger menu

FreedomBox's web interface is responsive. When the display or browser window is very narrow, menu options may be hidden.



That is because the top menu options are collapsed into the burger icon shown at the top right corner of the window. Click on it to display a drop-down menu.



3 Getting Help

The FreedomBox community provides live help via forum, chat and email. Feel free to join and ask anything you like. If you receive help, please consider to report your solution to the Questions and Answers page, so others can benefit in the future.

3.1 Discussion Forum

The easiest way to get support is by using the discussion forum. You can browse solutions to known problems or request help from community contributors by asking a question. This is also the best way to provide community contributors with feedback about your FreedomBox experience.

To post new content, you will need to register for an account with name and email address (but you can provide pseudonym and non-primary email address). By watching topics and categories or by enabling 'mailing list mode' in your account preferences, you can interact with the forum by just sending and receiving emails similar to a mailing list.

3.2 Matrix

You can join our Matrix room **#freedombox:matrix.org**. The room is federated with the IRC channel and remembers the chat history. If you do not yet have a client installed, you can use your web browser to join. For more options, see this matrix client overview page.

3.3 IRC #freedombox

Providing you are familiar with Internet Relay Chat (IRC) and IRC client, you can get an instant online help from the community on **irc.debian.org**, channel **#freedombox**. Potentially it takes some time before some member is answering you, be patient, a reaction will come later.

3.4 Email

FreedomBox users and contributors can be reached by email via a discussion list. In order to ask a question and get an answer from the community, please register from the mailing list page providing your email adress and creating a password. You can also read discussions archives. This list gathers about 700 readers.

3.5 Help Back

Once you've got your solution, don't forget to add it to the Questions and Answers page and tell which features do you use from the box on Use Cases page. It could help others to use FreedomBox in a way they would have not imagined.

4 Download and Install

Welcome to the FreedomBox download page.

- **Note**: If you purchased a FreedomBox kit, this section is not meant for you, so you can just skip it entirely. (Unless you specifically want to build an alternative software image).

You may either install FreedomBox on one of the supported inexpensive hardware devices, on any Debian operating system, or deploy it on a virtual machine.

Installing on a machine running a Debian system is easy because FreedomBox is available as a package. We do recommend to install FreedomBox on a supported single board computer (SBC). The board will be dedicated for FreedomBox use from home, this will prevent a lot of risks, such as accidental misconfiguration by the user. In case of trouble deciding which hardware is best for you or during the installation, please use the support page or read the Questions and Answers page based on posts on the Freedombox-discuss mailing list archives.

4.1 Downloading on Debian

If you are installing on an existing Debian installation, you don't need to download these images. Instead, read the instructions on setting up FreedomBox on Debian.

4.2 Downloading for SBC or Virtual Machine

4.2.1 Prepare your device

Read the hardware specific instructions on how to prepare your device at the Hardware section. On the web, there is a lot of documentation about setting your device up and flashing USB or SD Cards to boot your hardware.

4.2.2 Downloading Images

Recent images for supported targets are available here:

- Official Images: `https://freedombox.org/download/`
- Official Images: `https://ftp.freedombox.org/pub/freedombox/`

4.2.3 Verifying the Downloaded Images

It is important to verify the images you have downloaded to ensure that the file has not been corrupted during the transmission and that it is indeed the image built by FreedomBox developers.

Note: Testing and nightly images are automatically signed by the FreedomBox CI server.

- First open a terminal and import the public keys of the FreedomBox developers who built the images:

```
# This is the new FreedomBox CI server's key
$ gpg --keyserver keyserver.ubuntu.com --recv-keys  ↩
    D4B069124FCF43AA1FCD7FBC2ACFC1E15AF82D8C
```

- Next, verify the fingerprint of the public keys:

```
$ gpg --fingerprint D4B069124FCF43AA1FCD7FBC2ACFC1E15AF82D8C
pub   rsa4096 2022-03-09 [SC]
      D4B0 6912 4FCF 43AA 1FCD  7FBC 2ACF C1E1 5AF8 2D8C
uid           [ unknown] FreedomBox CI (Continuous Integration server) <admin@freedombox. ↩
    org>
sub   rsa4096 2022-03-09 [E]
```

- Finally, verify your downloaded image with its signature file `.sig`. For example:

```
$ gpg --verify freedombox-bookworm_all-amd64.img.xz.sig
gpg: assuming signed data in 'freedombox-bookworm_all-amd64.img.xz'
gpg: Signature made Wed 14 Jun 2023 03:22:04 PM PDT
gpg:                using RSA key D4B069124FCF43AA1FCD7FBC2ACFC1E15AF82D8C
gpg: Good signature from "FreedomBox CI (Continuous Integration server) <admin@freedombox. ↩
    org>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: D4B0 6912 4FCF 43AA 1FCD  7FBC 2ACF C1E1 5AF8 2D8C
```

4.2.4 Installation

After the download you can use the image to boot your chosen hardware (including virtual machines). You'll need to copy the image to the memory card or USB stick as follows:

1. Figure out which device your card actually is.
    1. Unplug your card.
    2. Run `dmesg -w` to show and follow the kernel messages.
    3. Plug your card in. You will see messages such as following:

```
[33299.023096] usb 4-6: new high-speed USB device number 12 using ehci-pci
[33299.157160] usb 4-6: New USB device found, idVendor=058f, idProduct=6361
[33299.157162] usb 4-6: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[33299.157164] usb 4-6: Product: Mass Storage Device
[33299.157165] usb 4-6: Manufacturer: Generic
[33299.157167] usb 4-6: SerialNumber: XXXXXXXXXXXX
[33299.157452] usb-storage 4-6:1.0: USB Mass Storage device detected
[33299.157683] scsi host13: usb-storage 4-6:1.0
[33300.155626] scsi 13:0:0:0: Direct-Access     Generic- Compact Flash    1.01 PQ:  ↩
    0 ANSI: 0
[33300.156223] scsi 13:0:0:1: Direct-Access     Multiple Flash Reader     1.05 PQ:  ↩
    0 ANSI: 0
[33300.157059] sd 13:0:0:0: Attached scsi generic sg4 type 0
[33300.157462] sd 13:0:0:1: Attached scsi generic sg5 type 0
```

```
[33300.462115] sd 13:0:0:1: [sdg] 30367744 512-byte logical blocks: (15.5 GB/14.4 ↩
    GiB)
[33300.464144] sd 13:0:0:1: [sdg] Write Protect is off
[33300.464159] sd 13:0:0:1: [sdg] Mode Sense: 03 00 00 00
[33300.465896] sd 13:0:0:1: [sdg] No Caching mode page found
[33300.465912] sd 13:0:0:1: [sdg] Assuming drive cache: write through
[33300.470489] sd 13:0:0:0: [sdf] Attached SCSI removable disk
[33300.479493]  sdg: sdg1
[33300.483566] sd 13:0:0:1: [sdg] Attached SCSI removable disk
```

4. In the above case, the disk that is newly inserted is available as */dev/sdg*. Very carefully note this and use it in the copying step below.

2. Decompress the downloaded image using tar:

```
$ xz -d freedombox-stable-free_buster_cubietruck-armhf.img.xz
```

   The above command is an example for the *cubietruck* stable image. Your downloaded file name will be different.

3. Copy the image to your card. Double check to make sure you don't write to your computer's main storage (such as /dev/sda). Also make sure that you don't run this step as root to avoid potentially overriding data on your hard drive due to a mistake in identifying the device or errors while typing the command. USB disks and SD cards inserted into the system should typically be write accessible to normal users. If you don't have permission to write to your SD card as a user, you may need to run this command as root. In this case triple check everything before you run the command. Another safety precaution is to unplug all external disks except the SD card before running the command.

   For example, if your SD card is */dev/sdg* as noted in the first step above, then to copy the image, run:

```
$ dd bs=1M if=freedombox-stable-free_buster_cubietruck-armhf.img of=/dev/sdg conv= ↩
    fdatasync status=progress
```

An alternative to copy to SD card command

-
```
$ cat freedombox-stable-free_buster_cubietruck-armhf.img > /dev/sdg ; sync
```

On MS Windows you will need a tool like *etcher*. On MacOS (OSX) you can use programs like *balenaetcher* and *rosaimagewriter*.

- The above command is an example for the *cubietruck* stable image. Your image file name will be different.

  When picking a device, use the drive-letter destination, like */dev/sdg*, not a numbered destination, like */dev/sdg1*. The device without a number refers to the entire device, while the device with a number refers to a specific partition. We want to use the whole device. Downloaded images contain complete information about how many partitions there should be, their sizes and types. You don't have to format your SD card or create partitions. All the data on the SD card will be wiped off during the write process.

- Use the image by inserting the SD card or USB disk into the target device and booting from it. Your device should also be prepared (see the Hardware section).

- Read (the rest of) the Manual for instructions on how to use applications in FreedomBox.

4.2.5 Troubleshooting

- Can't boot off your MicroSD card (and/or disk utilities like GPartEd report a missing/corrupt partition table). You likely forgot or failed to extract the .img file with `xz -d` before writing it to your device (e.g. */dev/sdg*).

4.3 Obtaining Source Code

FreedomBox is fully free software and you can obtain the source code to study, modify and distribute improvements.

4.3.1 From within FreedomBox

FreedomBox is made up of several software programs and you can obtain the source code to any of them. These instructions are similar to obtaining and building source code for Debian since FreedomBox is a pure blend of Debian. Using this process you can obtain the source code to the exact version of the package you are currently using in FreedomBox.

1. To see the list of software packages installed on your FreedomBox, run the following in a terminal:

   ```
   dpkg -l
   ```

2. To obtain the source code for any of those programs, then run:

   ```
   apt source <package_name>
   ```

   This requires that the apt sources list contains information about the source code repositories. These are present by default on all FreedomBox images. If you have installed FreedomBox using a package from Debian, you need to ensure that source repositories are added in the file.

3. To build the package from source code, first install its dependencies

   ```
   apt build-dep <package_name>
   ```

   Switch to the source directory created by the *apt source* command:

   ```
   cd <source_directory>
   ```

   Then build the package

   ```
   dpkg-buildpackage -rfakeroot -uc
   ```

4. Install the package:

   ```
   dpkg -i ../<built_package>.deb
   ```

4.3.2 Other Ways to Obtain Source Code

1. Source code for any of the packages can be browsed and searched using the web interface at sources.debian.org. For example, see the plinth package.
2. Source code and pre-built binary package for any version of a package including historic versions can be obtained from snapshot.debian.org. For example, see the plinth package.
3. You can also obtain the links to upstream project homepage, upstream version control, Debian's version control, changelog, etc. from the Debian tracker page for a project at tracker.debian.org. For example, see the tracker page for plinth package.
4. You can build and install a package from its Debian's version control repository. For example,

   ```
   git clone https://salsa.debian.org/freedombox-team/freedombox.git
   cd freedombox
   apt build-dep .
   dpkg-buildpackage -rfakeroot -uc
   dpkg -i ../freedombox*.deb
   ```

4.3.3 Building Disk Images

You can also build FreedomBox disk images for various hardware platforms using the freedom-maker tool. This is also available as a Debian package and source code for it may be obtained using the above methods. Build instructions for creating disk images are available as part of the source code for freedom-maker package.

FreedomBox disk images are built and uploaded to official servers using automated Continuous Integration infrastructure. This infrastructure is available as source code too and provides accurate information on how FreedomBox images are built.

4.3.3.1 U-boot on Pioneer Edition Images

There is one minor exception to the u-boot package present on the hardware sold as FreedomBox Home Server Kits Pioneer Edition. It contains a small but important fix that is not part of Debian sources. The fork of the Debian u-boot source repository along with the minor change done by the FreedomBox is available as a separate repository. We expect this change to be available in upstream u-boot eventually and this repository will not be needed. This package can be built on a Debian armhf machine as follows (cross compiling is also possible, simply follow instructions for cross compiling Debian packages):

```
apt install git git-buildpackage
git clone https://salsa.debian.org/freedombox-team/u-boot.git
cd u-boot
pbuilder create --distribution=buster
gbp buildpackage --git-pbuilder
```

The u-boot Debian package will be available in *u-boot-sunxi*.deb*. This package will contain

```
mkdir temp
dpkg -x u-boot-suxi*.deb temp
unxz <lime2_image_built_with_freedom_maker>
dd if=temp/usr/lib/u-boot/A20-OLinuXino-Lime2/u-boot-sunxi-with-spl.bin of=<lime2.img> seek ↩
    =8 bs=1k conv=notrunc
```

The resulting image will have the modified u-boot in it.

4.3.4 Old Signing Keys

Some very old disk images of FreedomBox have been signed by different GPG keys than the one listed above. Those signatures are still valid and can be verified using the older keys.

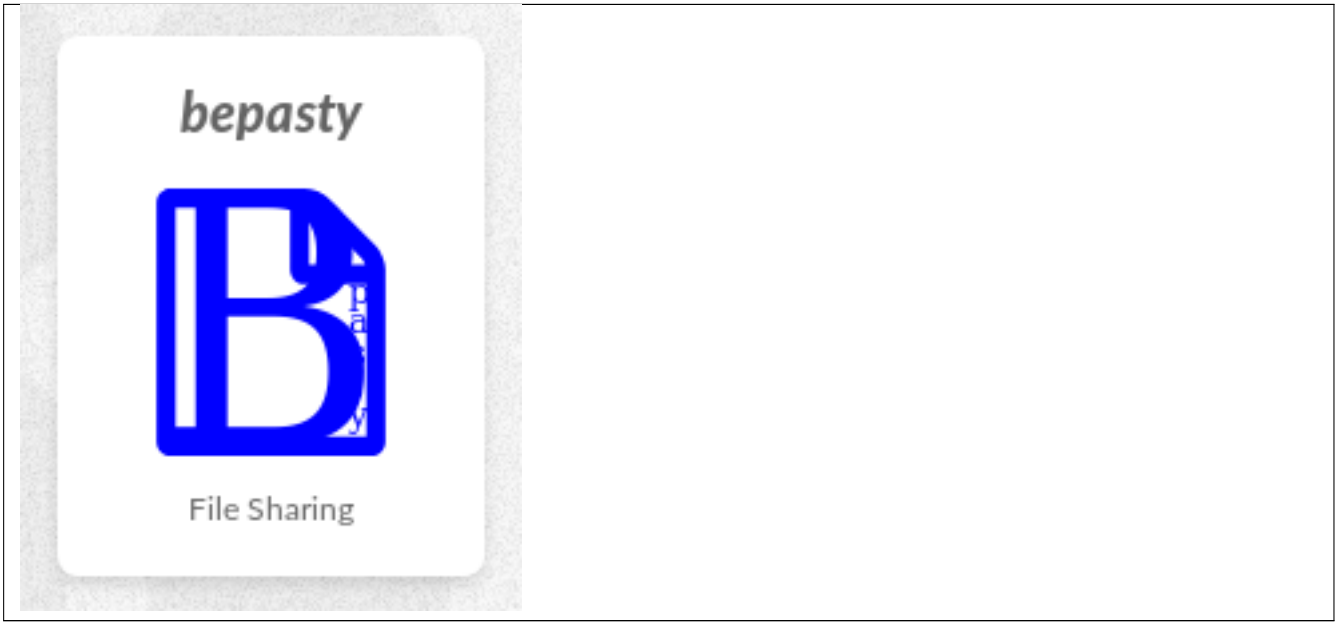
```
$ gpg --keyserver keyserver.ubuntu.com --recv-keys BCBEBD57A11F70B23782BC5736C361440C9BC971
$ gpg --keyserver keyserver.ubuntu.com --recv-keys 7D6ADB750F91085589484BE677C0C75E7B650808
# This is the FreedomBox CI server's key
$ gpg --keyserver keyserver.ubuntu.com --recv-keys 013D86D8BA32EAB4A6691BF85D4153D6FE188FC8
$ gpg --fingerprint BCBEBD57A11F70B23782BC5736C361440C9BC971
pub   4096R/0C9BC971 2011-11-12
      Key fingerprint = BCBE BD57 A11F 70B2 3782  BC57 36C3 6144 0C9B C971
uid                  Sunil Mohan Adapa <sunil@medhas.org>
sub   4096R/4C1D4B57 2011-11-12

$ gpg --fingerprint 7D6ADB750F91085589484BE677C0C75E7B650808
pub   4096R/7B650808 2015-06-07 [expires: 2020-06-05]
      Key fingerprint = 7D6A DB75 0F91 0855 8948  4BE6 77C0 C75E 7B65 0808
uid                  James Valleroy <jvalleroy@mailbox.org>
uid                  James Valleroy <jvalleroy@freedombox.org>
sub   4096R/25D22BF4 2015-06-07 [expires: 2020-06-05]
sub   4096R/DDA11207 2015-07-03 [expires: 2020-07-01]
sub   2048R/2A624357 2015-12-22

$ gpg --fingerprint 013D86D8BA32EAB4A6691BF85D4153D6FE188FC8
pub   rsa4096 2018-06-06 [SC]
      013D 86D8 BA32 EAB4 A669  1BF8 5D41 53D6 FE18 8FC8
uid           [ unknown] FreedomBox CI (Continuous Integration server) <admin@freedombox. ↩
    org>
sub   rsa4096 2018-06-06 [E]
```

5 Apps

5.1 Bepasty (File & Snippet Sharing)

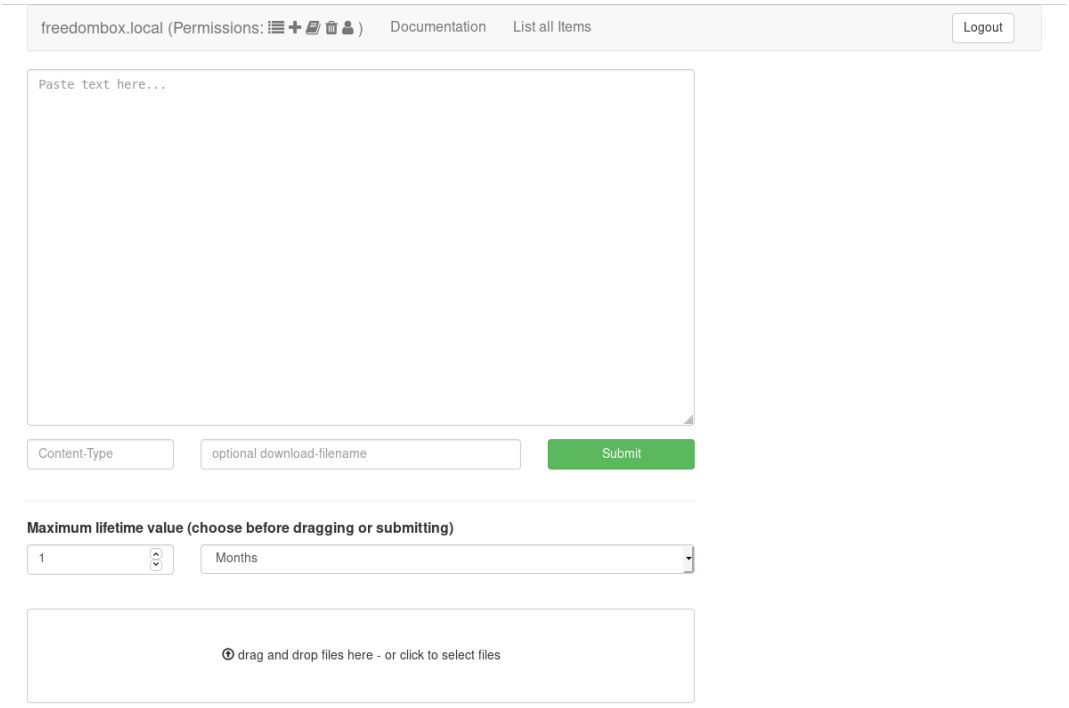


Available since: version 20.14

5.1.1 What is bepasty?

bepasty is a web application that allows large files to be uploaded and shared. Text and code snippets can also be pasted and shared. Text, image, audio, video and PDF documents can be previewed in the browser. Shared files can be set to expire after a time period.

5.1.2 Screenshot



5.1.3 Passwords and Permissions

bepasty uses only passwords (without usernames) to control access. Depending on which password is used to login to bepasty, the user will have different permissions. They can have any combination of the following permissions:

- **read**: Read a file, if they know the URL.
- **list**: List all files.
- **create**: Paste or upload a new file.
- **delete**: Delete a file.
- **admin**: Can lock and unlock files.

After bepasty is installed, it comes pre-configured for the following roles:

- Viewer: can view and list files
- Editor: can view, list, create, and delete files
- Administrator: has all permissions

These roles support a use-case of file sharing between known, authorized users. If needed, you can re-configure bepasty to support other roles and use-cases.

5.1.4 Distributing passwords

By default, the Public Access configuration is set to *None*, so a password is required for any use of bepasty. This means that you will need to distribute the passwords to the appropriate users, through any communication channels that you have.

Note that you may want to create multiple passwords with the same permissions. This allows you to distribute a unique password to each user (or to a group of users). Then if you want to revoke access to one user, you can simply delete their password. The other users with their own passwords will not be affected.

5.1.5 Using bepasty

After logging in to bepasty, if you have the Create permission, you will see a large text box where you can paste any text. Optionally, you can provide a filename or Content-Type for the data. After clicking Submit, the file is created.

You can also drag and drop files in the area at the bottom. They are uploaded immediate after dropping them in this area. You can also create a list to track a collection of uploaded files.

For either case, you can set a maximum lifetime value. After this time expires, the file will be deleted.

If you have the List permission, then you will see a link *List all Items* at the top of the page. This will show all files that have been created or uploaded.

If you have the Delete or Admin permission, you will see extra actions shown next to each file on the list page.

If you only have the Read permission, then to read files, you will need to have both a password and one or more URLs for existing files.

5.1.6 Managing passwords

The bepasty configuration page in FreedomBox interface allows you to create new passwords, or to remove a password. When you create a password, you can choose any combination of the permissions described above. Note that a typical Administrator should have all of the permissions (not just "Admin").

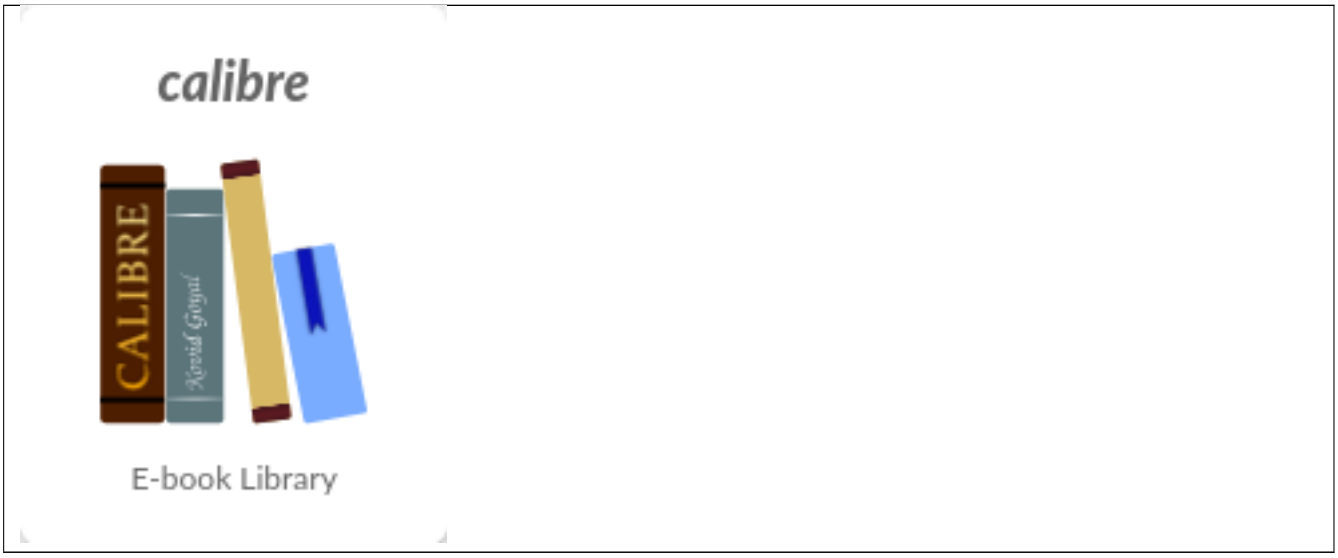
You can also set a Comment. This is recommended, and you should use the comment to help yourself remember the purpose of the password, or who will be using the password.

You can also configure Public Access, which sets the default permissions that are available even without logging in with a password. You can set this to allow reading files by their URL, or reading and listing all files.

5.1.7 External links

- Upstream project: https://github.com/bepasty
- Upstream end user documentation: https://bepasty-server.readthedocs.io/en/latest/user.html

5.2 Calibre (e-Library)



Available since: version 20.15

calibre is an e-book management solution. You can organize your e-books into collections in calibre known as "libraries". calibre can do e-book format conversion between most of the popular e-book formats. It can also manage metadata of your e-books such as book covers, descriptions, author and publisher information etc.

Moving your calibre library from your desktop to your FreedomBox has the benefit of being able to access your e-books from any device on the local network or through the Internet.

Only users who are members of the *calibre* group have access to the libraries. You can assign users to this group via the system app users and groups.

You might be familiar with the e-book reader shipped with the calibre application on your desktop. The server version of calibre that's installed on your FreedomBox has a web-based e-book reader with similar look and feel. This allows you to read your e-books from any device with a web browser.

5.2.1 Managing Libraries

After installation of calibre, a default library called "Library" will be made available. The FreedomBox administrator can add or delete any of the libraries including the default one from the app settings in FreedomBox web interface.

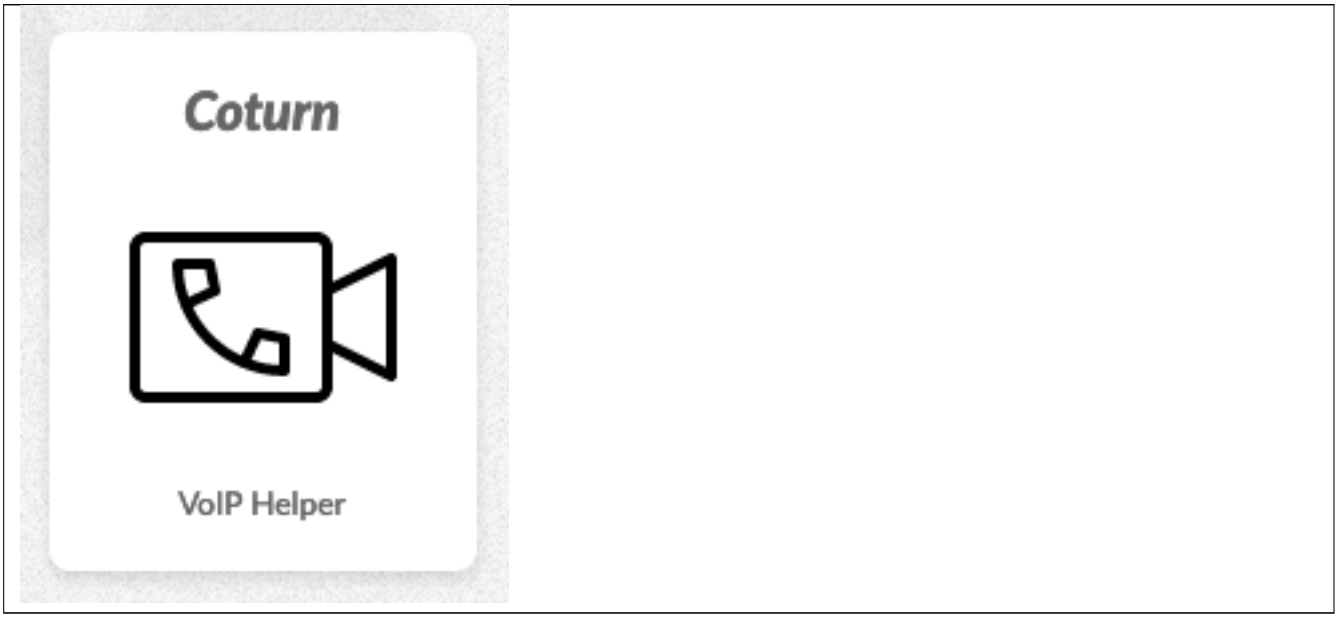
5.2.2 Access

calibre can be accessed after installation through the web client at `https://<my_freedombox_name>/calibre`.

5.2.3 External links

- Upstream project: https://calibre-ebook.com
- Upstream end user documentation: https://calibre-ebook.com/help

5.3 Coturn (VoIP Helper)



Available since: version 20.8

Coturn is a server to facilitate audio/video calls and conferences by providing an implementation of TURN and STUN protocols. WebRTC, SIP and other communication servers can use it to establish a call between parties who are otherwise unable connect to each other.

It is not meant to be used directly by users. Servers such as Matrix Synapse need to be configured with the details provided on the Coturn app page. Apart from Matrix Synapse, Jitsi, Ejabberd, Nextcloud Talk, etc. can use Coturn server for audio/video calls and conferences. There is no need for the servers to be running on the same machine as FreedomBox and external servers can use Coturn running on FreedomBox.

Coturn is configured in FreedomBox as an advanced app. This means that you need to check "Show advanced apps and features" in "General Configuration" to see Coturn icon in the "Apps" section.

5.3.1 How it works

When making an audio/video call, it is best to route the media streams between two peers directly. This will give the best possible latency (better signal quality) and avoid depending on a centralized server (privacy). It scales well because a simple chat server can host thousands of calls without involving the server in any way other than to setup the call. However, this approach does not work most of the time due to how networks are configured. Most peers on the network do not have a unique IP address allocated to them. They work hidden behind a network device that performs "Network Address Translation" (NAT) for them. This means that the two peers have no way of reaching each other.

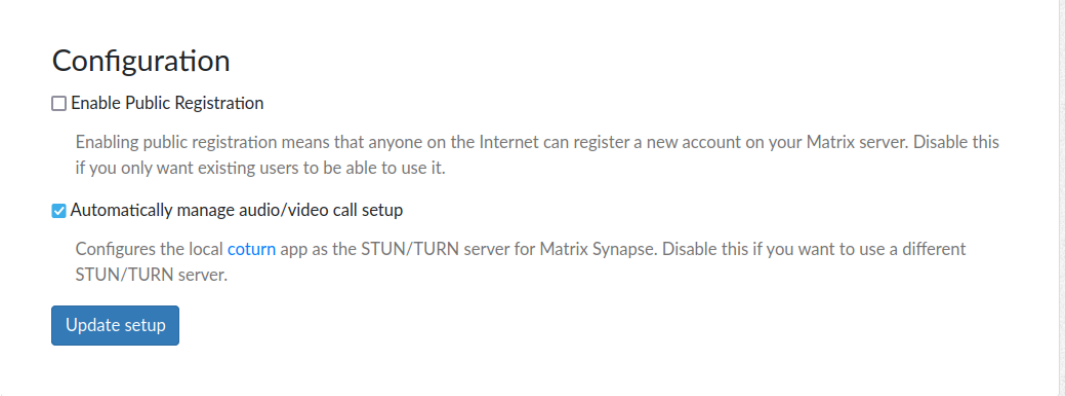
To address this problem, a simple technique known as STUN was introduced. With the help of a third party STUN server, the peers can trick the NAT devices, to carry the traffic between the two peers. Unfortunately, this trick only works about 80% of the time. So, if STUN fails, peers have no choice but to route their traffic through an intermediary server called TURN server. All the mechanism of trying out STUN first and then falling back to TURN is described in a protocol known as ICE.

On FreedomBox, Coturn provides both STUN and TURN servers. Both services are provided over TCP as well as UDP. They are provided on unencrypted as well as encrypted channels (which have a higher chance of success). Since STUN servers are very inexpensive and don't consume a lot of server resources, there is no authentication needed to use them. TURN servers on the other hand need authentication. This authentication is highly simplified and does not require maintaining a database of users. A server such as matrix-synapse which is about to setup an audio/video call between two peers will generate a username and password using a shared secret. When the peers use the TURN server, they will be validated using these credentials because the TURN server also knows the same secret.

In summary, a communication server needs to know the URLs of the STUN/TURN servers along with a shared authentication secret for TURN. After that, during audio/video call setup, they will correctly guide the peers to use STUN/TURN servers. Coturn app in FreedomBox provides exactly this information. This information can be used to configure a communication server irrespective of whether it is running on the same FreedomBox or on another server.

5.3.2 Configuring Matrix Synapse

To configure Matrix Synapse to use Coturn TURN/STUN server, you need to check "Automatically manage audio/video call setup" in Matrix Synapse's configuration section, and then click on "Update seup".



5.3.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Coturn:

- UDP 3478
- TCP 3478
- UDP 3479
- TCP 3479
- UDP 5349
- TCP 5349
- UDP 5350
- TCP 5350
- UDP 49152-50175
- TCP 49152-50175

5.3.4 External links

- Upstream project: https://github.com/coturn/coturn
- Upstream end user documentation: https://github.com/coturn/coturn/wiki/

5.4 Deluge (Distributed File Sharing via BitTorrent)



Available since: version 0.5

5.4.1 What is Deluge?

Deluge is a BitTorrent node (both, client and server at the same time).

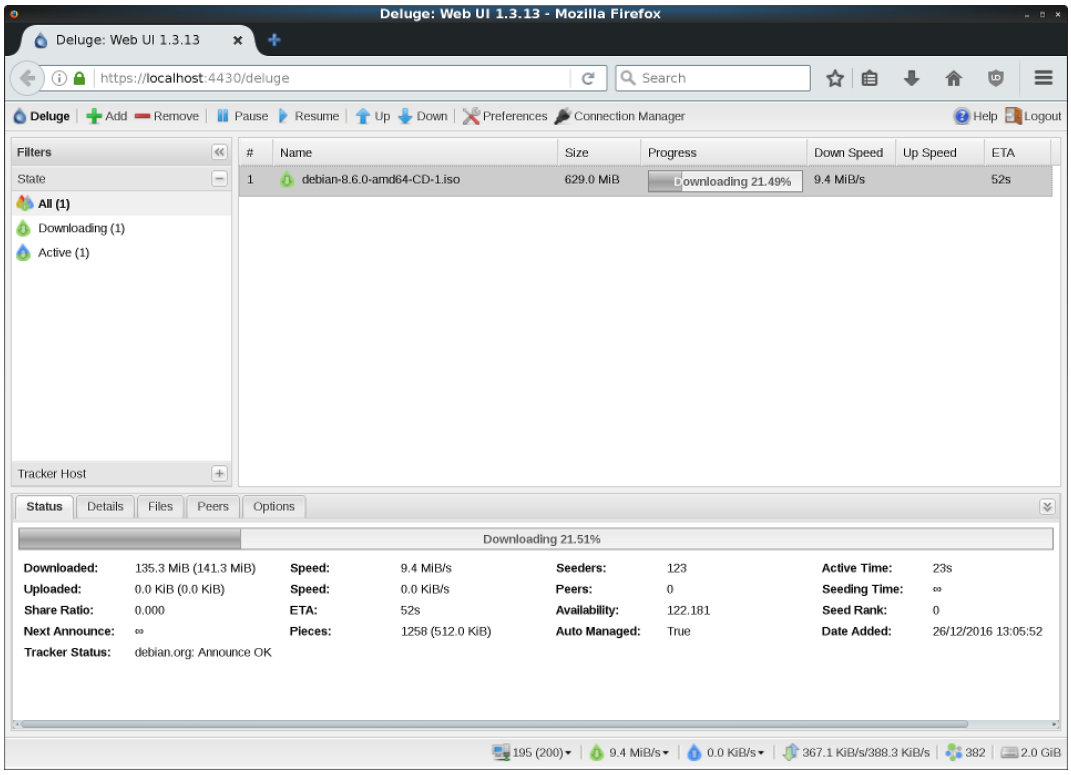
BitTorrent is a communications protocol for peer-to-peer (P2P) file sharing.

- It is **not anonymous**; you should assume that others can see what files you are sharing.
- This technology works best for big, popular files.

There are two BitTorrent web nodes available in FreedomBox: Transmission and Deluge. They have similar features, but you may prefer one over the other.

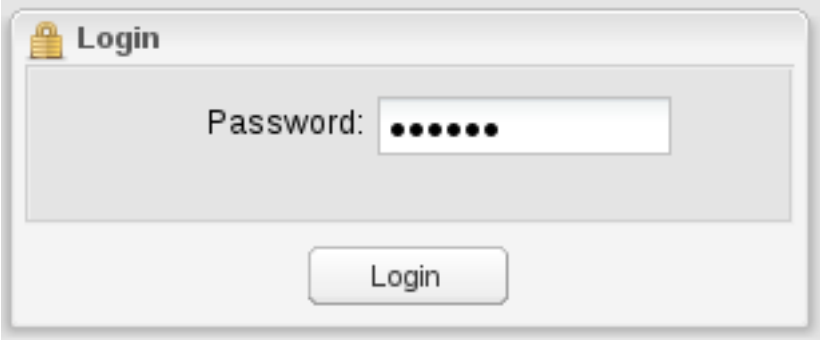
Deluge is a lightweight BitTorrent client that is highly configurable. Additional functionality can be added by installing plugins.

5.4.2 Screenshot



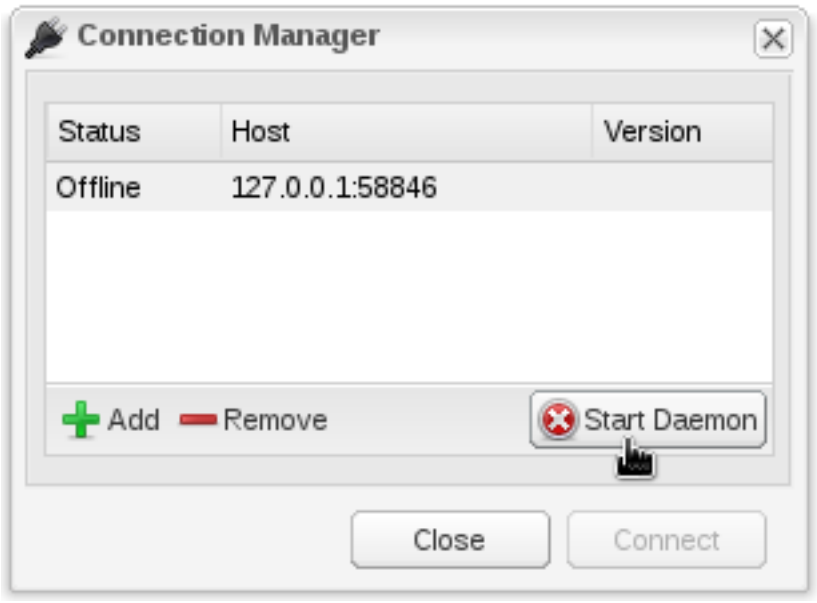
5.4.3 Initial Setup

After installing Deluge, it can be accessed by pointing your browser to `https://<your freedombox>/deluge`. You will need to enter a password to login:

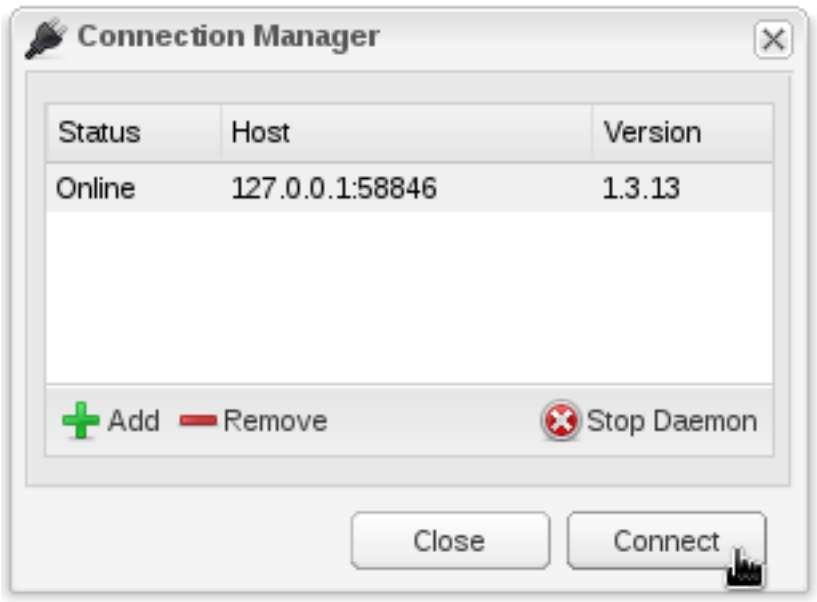


The initial password is "deluge". The first time that you login, Deluge will ask if you wish to change the password. You should change it to something that is harder to guess.

Next you will be shown the connection manager. Click on the first entry (Offline - 127.0.0.1:58846). Then click "Start Daemon" to start the Deluge service that will run in the background.



Now it should say "Online". Click "Connect" to complete the setup.



At this point, you are ready to begin using Deluge. You can make further changes in the Preferences, or add a torrent file or URL.

5.4.4 External links

- Upstream project: https://www.deluge-torrent.org
- Upstream end user documentation: https://dev.deluge-torrent.org/wiki/UserGuide

5.5 Ejabberd (Chat Server)



Available since: version 0.3

5.5.1 What is ejabberd?

Ejabberd is a chat server which uses the Extensible Messaging and Presence Protocol (XMPP).

5.5.2 What is XMPP?

XMPP is a federated server-client protocol for Instant Messaging. This means that users who have accounts on one server, can talk to users that are on another server.

XMPP can also be used for voice and video calls, if supported by the clients.

Currently FreedomBox offers both, a server (ejabberd) and a web client (JSXC) from its web interface.

5.5.3 Privacy

With XMPP, there are two ways that conversations can be secured:

1. TLS: This secures the connection between the client and server, or between two servers. This should be supported by all clients and is highly recommended.
2. End-to-end: This secures the messages sent from one client to another, so that even the server cannot see the contents. The latest and most convenient protocol is called OMEMO. There is another protocol called OTR that may be supported by some clients that lack OMEMO support. Both clients must support the same protocol for it to work.

5.5.4 Setting the Domain Name

For XMPP to work, your FreedomBox needs to have a Domain Name that can be accessed over the network.

If you only need the local network (LAN) users to chat with each other you can invent your domain name, but if you want users from the internet to join your rooms you need a public domain name. You can read more about obtaining a Domain Name in the Dynamic DNS section of this manual.

Once you have a Domain Name, you can tell your FreedomBox to use it by setting the Domain Name in the System Configuration.

Note: After changing your Domain Name, the Chat Server (XMPP) page may show that the service is not running. After a minute or so, it should be up and running again.

Please note that PageKite does not support the XMPP protocol at this time.

5.5.5 Registering FreedomBox users to use XMPP

Currently, all users created through FreedomBox will be able to login to the XMPP server. You can add new users through the System module Users and Groups. It does not matter which Groups are selected for the new user.

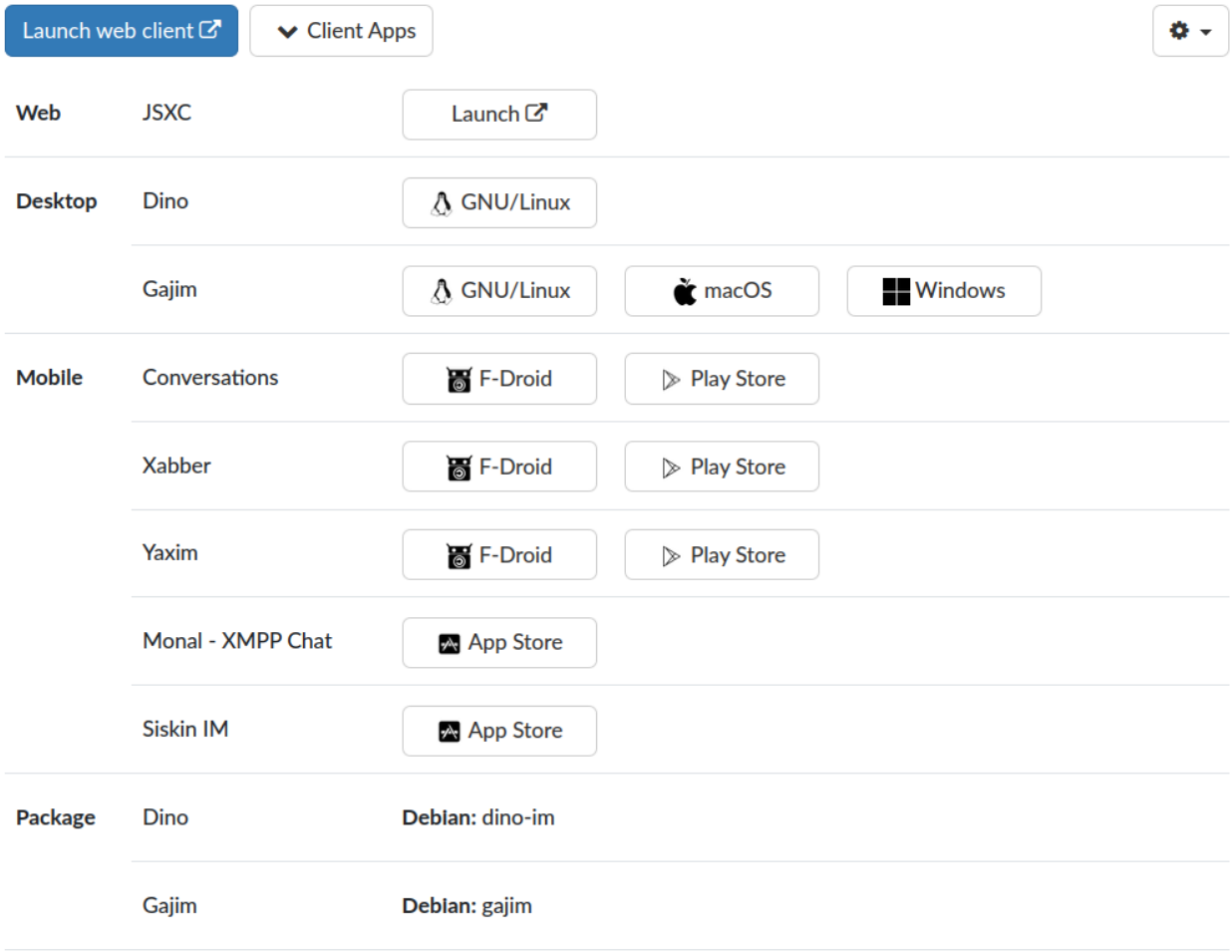
5.5.6 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for XMPP:

- TCP 5222 (client-to-server)
- TCP 5269 (server-to-server)

5.5.7 Compatible clients

- FreedomBox provides a web client: JSXC.
- XMPP clients are available for various desktop and mobile platforms. FreedomBox links to the download sources of some of them. Feel free to include more here (needs free registration). We'll notice and might list them in FreedomBox.



5.5.7.1 FreedomBox webclient

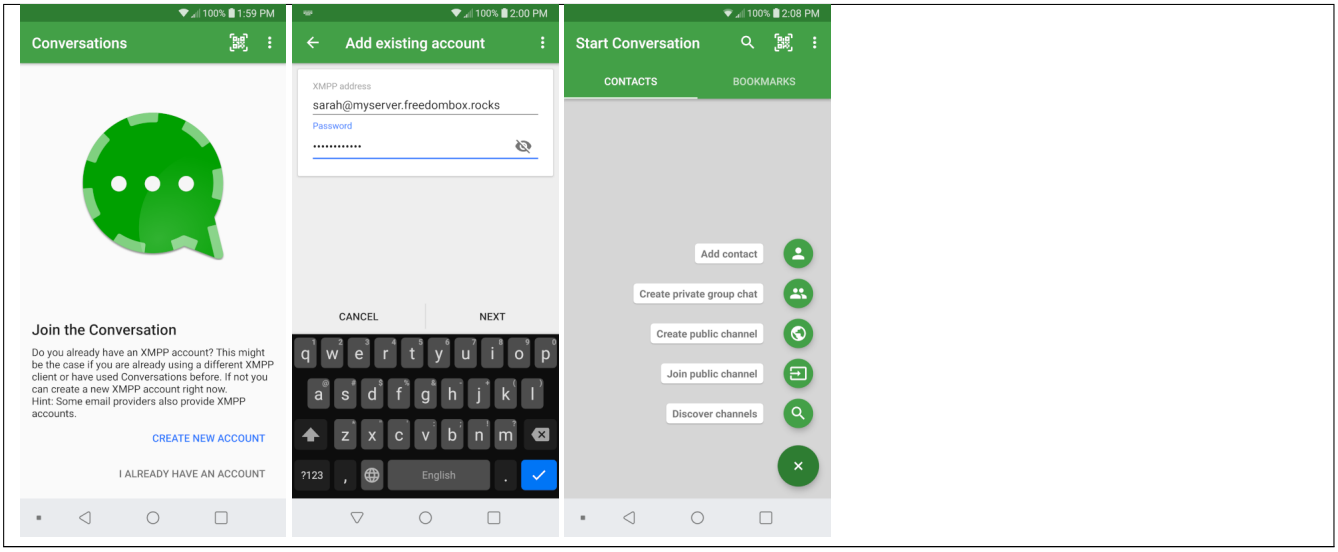
For maximum simplicity FreedomBox provides a web client: JSXC. No need for your users to install additional software on their side. They'll be able to just use their browser. This is usually the first choice for new and eventual users.

5.5.7.2 Mobile clients

You can download an XMPP client for your smartphone or tablet among the ones listed below.

5.5.7.2.1 Conversations (Android)

Conversations is an Android XMPP client with audio and video call support available on F-Droid or the Play Store. Audio and video calls also work between Conversations and Monal on iOS. In addition to text messaging, you can use Conversations to send images/files and have group chats.



From left to right: (1) First screen - (2) Login screen - (3) Add contacts.

When first starting the Conversations app, you will be asked whether you want to create a new account or if you want to use an existing account. Choose "I already have an account" (1)

With ejabberd installed, the FreedomBox provides an XMPP account for every FreedomBox user. Additional (non-admin) FreedomBox user accounts can be created under **System >** Users and Groups.

Once logged into a FreedomBox/XMPP account (2), the Conversation app provides a + button that brings up a few choices to contact other people (3).

5.5.7.2.2 Monal (iOS)

Monal is a free software XMPP client with audio and video call support available from the App Store. Audio and video calls also work between Conversations on Android and Monal. In addition to text messaging, you can use Monal to send images/files and have group chats.

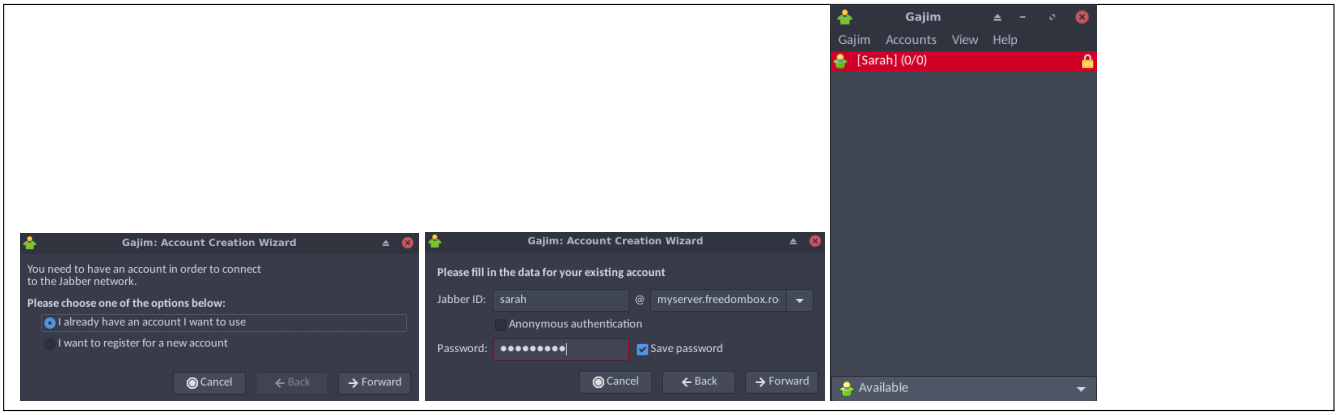
5.5.7.2.3 Siskin (iOS)

Siskin is a free software XMPP client with videochat support available from the App Store.

5.5.7.3 Desktop clients

5.5.7.3.1 Gajim (Windows, MacOS, Linux)

Gajim is a XMPP open-source client for the desktop, available for Windows, MacOS and Linux. This application is available in Debian, and for other operating systems you can download it from this page and find instructions about installation.



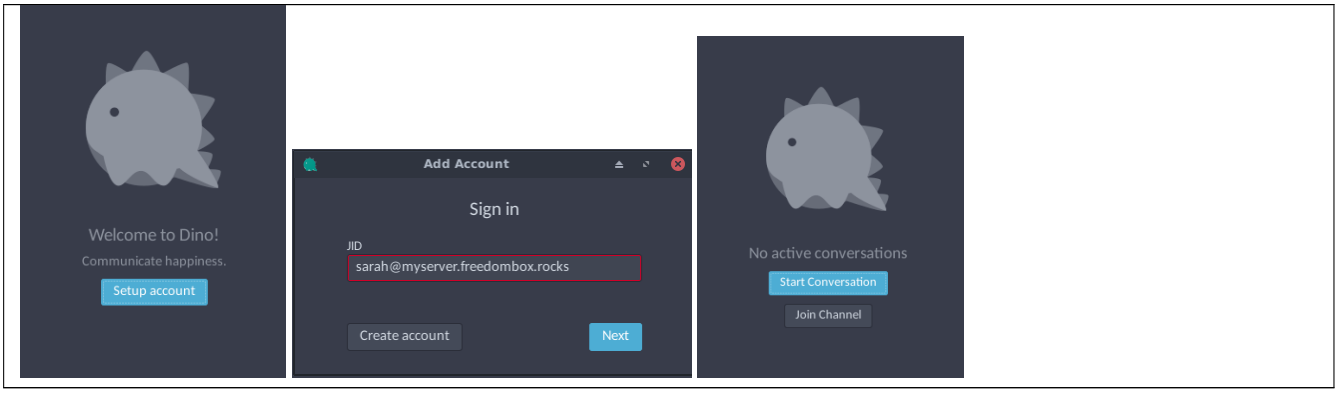
From left to right: (1) First screen - (2) Login screen - (3) Main window

A popup shows up right after you start Gajim for the first time (1), asking you to either login to your XMPP (FreedomBox) account or to register for a new account. When you choose to login, after clicking "Forward", you will be asked a Jabber ID and a password (2): you have to enter your FreedomBox account and password here.

Finally, after logging in successfully, you will see the main Gajim screen (3). From there, you can add a contact (*Account* > *Add contact...*) then you can start a conversation (*Gajim* > *Start chat*).

5.5.7.3.2 Dino (Linux)

Dino is another XMPP free software client for the desktop. It is available for `https://github.com/dino/dino/wiki/Distribution-Packages`.



From left to right: (1) First screen - (2) Login screen - (3) Start conversation

When first starting Dino after installation, click on the `Setup account` button. You will be then asked your `JID`: this is your FreedomBox account. Enter it then click `Next` (2). Alternatively, you can click on `Create account` if you don't have a FreedomBox account.

Once you have logged in, you will be able to either start a conversation with one of your XMPP contacts or to join a channel (3).

5.5.7.3.3 Movim (Linux)

Movim is a free software XMPP client with videochat support for Linux. The project provides an unofficial Debian package.

5.5.7.3.4 Monal (MacOS)

Monal is a free software XMPP client with videochat support available from the Mac App Store.

5.5.8 External links

- Upstream project: https://www.ejabberd.im
- Upstream end user documentation: https://docs.ejabberd.im

5.5.8.1 Client Apps

Links to recommended client apps, where applicable, can be found in FreedomBox by select the Apps page, selecting the application tile, and clicking the **> Client Apps** button.

- FreedomBox includes JSXC client software

5.6 Postfix/Dovecot/Rspamd (Email Server)

Available since: 22.6

5.6.1 About the Email Server

FreedomBox provides a complete email server solution using Postfix, Dovecot, and Rspamd. Postfix sends and receives emails. Dovecot allows email clients to access your mailbox using IMAP and POP3. Rspamd deals with spam. The following features are available:

- Send and receive email
  - Interoperate with other mail servers
  - Prevent others from spoofing your email addresses using SPF
  - Sign all outgoing email using DKIM
  - Receive reports of spoofing attempts using DMARC
- Access mails easily
  - Access mail from any device using Roundcube webmail
  - Configure email clients by just typing in email address and password
  - Auto-configuration works with clients using autoconf scheme such as Thunderbird
  - Auto-configuration works with clients using DNS scheme
  - Keep mails on server and access them with multiple clients using IMAP
  - Fetch mails to local machine using POP3
- Email address for all your users
  - Each user on your FreedomBox automatically gets an email address such as user@mydomain.example
  - Each user gets unlimited automatic aliases. user+purpose@mydomain.example points to user@mydomain.example
  - Users may themselves add more aliases. foo@mydomain.example can point to user@mydomain.example
  - Many common aliases such as info@mydomain.example and postmaster@mydomain.example point to administrator's email address.
- Filter messages on the server using sieve filters
  - Setup vacation auto-responders that work even when you are not using your email client
  - Forward to external/internal addresses, file in folders, delete, etc.
  - Manage filters on the server using email client (for example, Thunderbird with sieve add-on)
- Automatically setup and configure TLS certificates obtained by Let's Encrypt

  - All services (SMTP, IMAP, POP3 and `manageseive`) are configured to use TLS/STARTTLS
  - Certificates are renewed every 3 months or so
  - Upon renewal, certificates are installed and services are restarted
- Backup and restore emails, aliases and configuration
  - Set a schedule for periodic backups
- Scan incoming email for spam
  - Check the message against various block lists
  - Automatically move spam to the Junk folder
  - Sets a flag on the message when spam score reaches a threshold
  - Rejects the message during receiving when spam score reaches a high threshold
  - Understand spam decisions using extended spam headers added to a message
  - View details of spam processing and manage settings using Rspamd web interface
  - Admins can login to spam web interface using FreedomBox single-sign-on
  - Teach spam vs. not-spam using example messages

5.6.2 Prerequisites

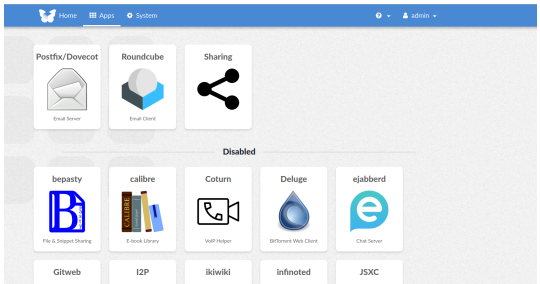
- You must own a domain on which you can configure advanced DNS records (MX, TXT and SRV). Such a domain can be obtained by buying one from a registrar or by obtaining a paid service from one of the Dynamic DNS providers (such as `freedns.afraid.org`). Currently, free subdomains provided by FreedomBox Foundation's free Dynamic DNS service at `ddns.freedombox.org` are **not** suitable. Support is planned in future.
- Your ISP or cloud provider, on your Internet connection, must not be blocking traffic to external mail servers. Quite a few of them block outgoing traffic on port 25. This will render the email server unable to send mails to external addresses. Many such providers allow you to request removing this restriction. To test whether this is a problem for your Internet connection, run the following command (you should see some text like this):

```
user@myserver:~$ nc freedombox.org 25
220 mx.sflc.info ESMTP Postfix (Debian/GNU)
^C
```

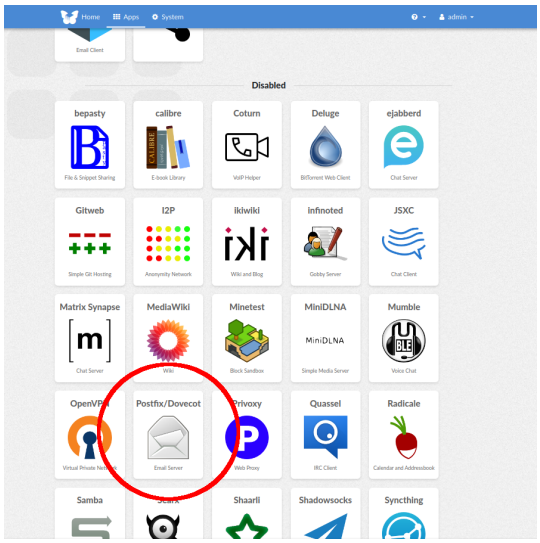
5.6.3 Installing

Go to the *Apps* menu.

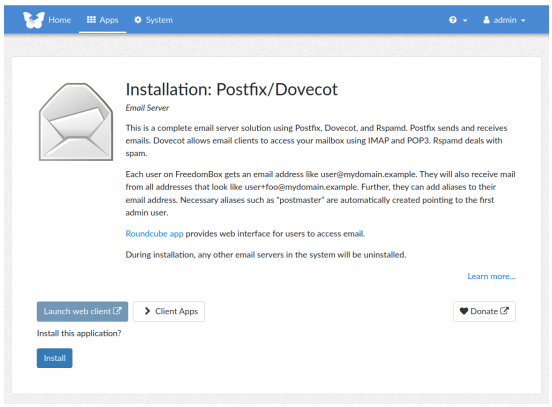
If already installed, the Email Server will be shown **above** the *Disabled* line. This is likely not your case, but if it is, that means that the Email Server is already installed, so skip this step and jump to the next one.

- 

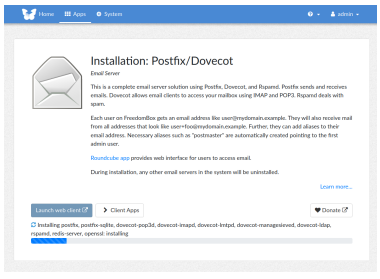
If the Email Server is shown among the icons **below** the *Disabled* line, it is either not yet installed or it is currently disabled. This is the usual starting status.

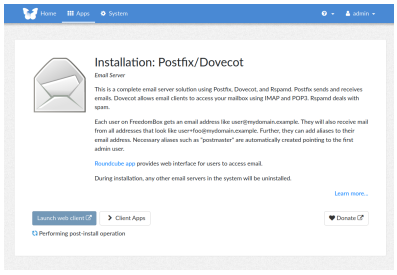
- 

Select the *Postfix/Dovecot* app. You are presented with the *Postfix/Dovecot* app page. If not installed yet you'll be shown the *Install* button. Click on it!

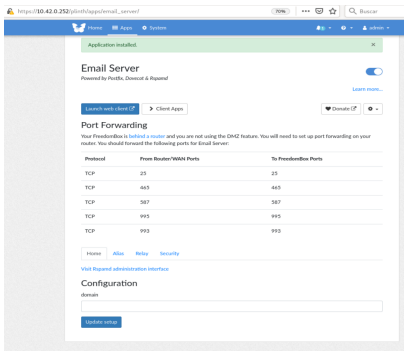
- 

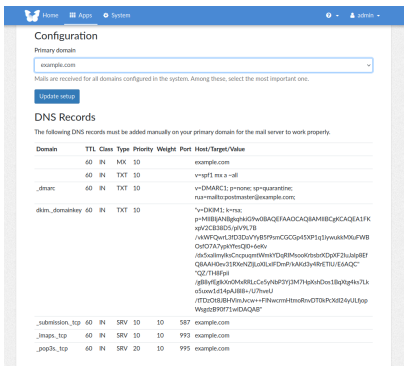
This will trigger the installation process.

- 



After installing all needed software packages and configuring them, FreedomBox will tell you that the installation is successful and the app page will show additional content such as port information, configuration form and DNS settings.

- 



Next time you go to the Apps Menu it will show the app enabled (above the *disabled* line).

5.6.4 Configuring the Email Server

1. If you wish to send email to and receive mails from users on other email servers on the Internet, you need a proper domain. As explained in the Prerequisites section, either buy a domain from a registrar or obtain one from a Dynamic DNS provider. If you purchased a domain from a registrar add it in the System, Configuration page. If it is a Dynamic DNS, configured it in System, Dynamic DNS Client page.

2. When a domain is added to FreedomBox, a TLS certificate is automatically obtained for the domain. This certificate is then used for encrypted communication with all the services that are configured with the domain. Go to System section, Let's Encrypt app page and verify that certificate has been successfully obtained for the domain. If not, click on the Obtain button and resolve any problems that show up. For successfully obtaining the certificate, your FreedomBox must be reachable from the Internet and your router, if any, must be configured to do port forwarding for the web ports (80, 443).

3. After adding a domain to FreedomBox, visit the Email app page. In the Configuration section, select the configured domain as the primary domain for the purposes of sending and receiving email.

4. After setting the primary domain, information will become available in the DNS Records section of the page. These are the records that must be manually configured on the domain. Login to your DNS provider's web interface for managing DNS records on your domain. There enter all the entries shown in the DNS records table.

   - The length of the value of DNS record for DKIM exceeds 255 characters in length. Typically, it must be broken into multiple values enclosed in the double quotes and separated by spaces. This is what FreedomBox does. If your DNS provider has a different way to enter these multiple values, consult their documentation.
   - All the records are assumed "under" the domain you are configuring but a full value can also be provided. For example, "Domain" value of "dkim._domainkey" means "dkim._domainkey.mydomain.example.". Use the latter form if necessary.

5. Install Roundcube app if you want to access emails using a web interface. In Roundcube configuration, be sure enable option to "Use only the local mail server". This removes the server field in the login page and makes the app work without any further configuration.

5.6.5 Using the Email Server

As a user you can:

- Start sending and getting emails using most email clients.
- Create and/or manage your email aliases in the *Aliases* tab of the *Email Server* app page in FreedomBox web interface.
- Manage filters on the server using sieve

Once an admin has set up RoundCube configuration for it to work with the FreedomBox Email server you can log into RoundCube and start sending emails without the need for other email clients. Use the same login credentials to RoundCube that you use to log into the FreedomBox web interface.

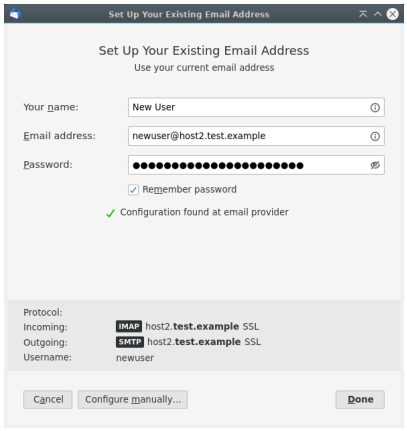
5.6.5.1 With FreedomBox Webmail Client (RoundCube)

RoundCube email client is provided by FreedomBox as an optional app. If RoundCube has been installed before the email server, there is an option to make it work with FreedomBox's email server. Once both apps are installed, you have a complete webmail setup ready.

5.6.5.2 With Thunderbird

Open Thunderbird. Go to *hamburger menu → New → Existing Mail Account*. Enter a display name, your FreedomBox email address, and your FreedomBox password. Click continue.

FreedomBox implements the Automatic Account Configuration endpoint which Thunderbird will make use of.

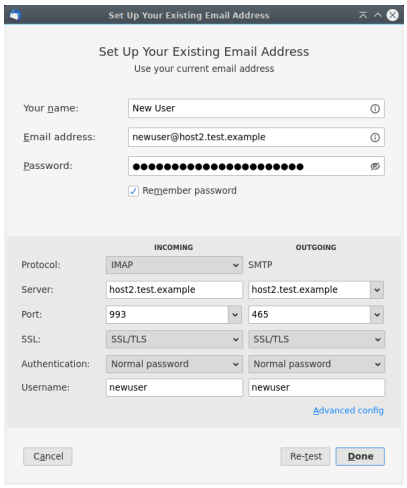
- 

5.6.5.3 Manual Configuration

Tell your email client to use these parameters:

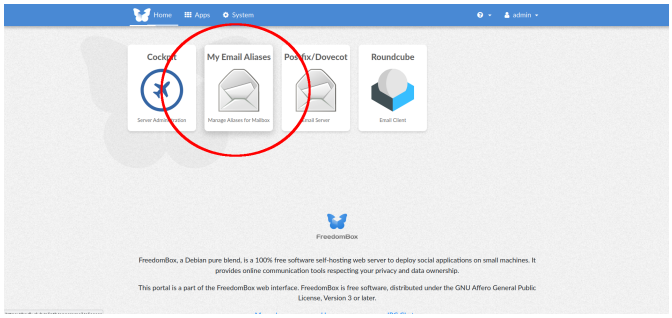
- **Username:** *your FreedomBox email address or just the username part*
- **Incoming mail:** IMAPS, port 993, forced SSL, normal password authentication
- **Outgoing mail:** SMTPS, port 465, forced SSL, normal password authentication

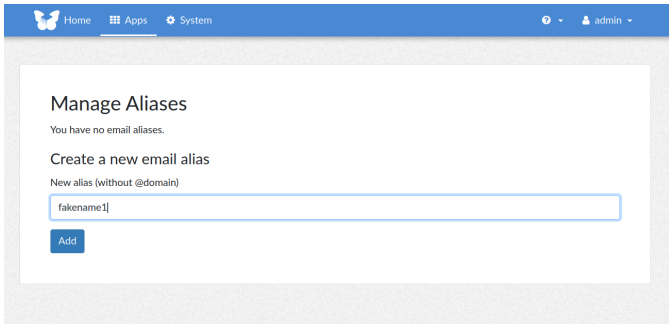
STARTTLS on the SMTP submission port is also supported.



5.6.5.4 Email Aliases

Email aliases are useful for privacy. Now as FreedomBox email user (you don't need to be an administrator) you can have temporary throw-away and specific email addresses under your control. You can list, create and delete email aliases from the *My Email Aliases* shortcut in FreedomBox home page.





Mails to non-existent users, non-existent aliases, or system users will be rejected at the SMTP connection level.

5.6.5.5 Automatic Email Aliases

In addition to allowing users to create their own aliases, FreedomBox also sets up automatic aliases by appending a string to your user name with a '+' sign. If your mail address is myname@mydomain.example, then all myname+anystring@mydomain.example is an automatic alias to your email address. For example, when subscribing to a mailing list call foolist, you can provide your email address as myname+foolist@mydomain.example. When mail is sent to that address, it ends up in your mailbox of myname@mydomain.example. This is primarily useful for mail sorting and spam control.

5.6.6 Advanced: Troubleshooting

5.6.6.1 How to debug an action script failure? How to access the system log?

Open a secure shell connection to your FreedomBox. Type `sudo journalctl -b -o short-monotonic --no-pager`

- `-b` show journal entries since boot
- `-o short-monotonic` use short timestamp format
- `--no-pager` make it easier to copy and paste

5.6.6.2 Why does the server say "relay access denied"?

This is because Postfix was not aware of the email domain. To fix that,

1. Ensure FreedomBox is aware of your internet domain name. If you don't have a domain name, skip to step 2.
   - Log into the FreedomBox web interface as an admin.
   - Go to *System → Name Services*
   - Add a domain name if you haven't done so.

5.6.6.3 Cannot send anything from Roundcube. It says "SMTP Error (250): Authentication failed".

Root cause: Roundcube tried to submit your email from an unencrypted connection, but ports 465 and 587 required SSL and STARTTLS encryption, respectively.

Solutions:

For RoundCube, edit the `/etc/roundcube/config.inc.php` file to make it use port 25 (unencrypted). Fix these settings:

```
$config['smtp_server'] = 'smtp://localhost';
$config['smtp_port'] = 25;
```

Notes:

- Access your FreedomBox via SSH.
- You can edit the file with `nano` text editor. The file is restricted, so you need to access it as superuser: `sudo nano /etc/roundcube/config.inc.php`.

If using another email client like Thunderbird, enforce SSL or STARTTLS usage by the email client.

5.6.7 Providing user feedback

Please provide your feedback on usage on this forum thread.

5.6.8 Technical info and discussion

FreedomBox email server was presented at Debconf21. Slides and video recording are available courtesy of the Debian Outreach team.

This salsa issue is driving the implementation. Feel free to join discussions and provide technical ideas.

5.6.9 External links

- Upstream projects:
  - https://www.dovecot.org
  - http://www.postfix.org
  - https://www.rspamd.com
- Debian community wiki:
  - Dovecot
  - Postfix

5.6.9.1 Client Apps

Links to recommended email client apps can be found in FreedomBox by select the Apps page, selecting the Dovecot/Postfix tile, and clicking the **> Client Apps** button.

- FreedomBox includes Roundcube client software.

5.7 Feather Wiki (Personal Notebooks)



Available since: version 24.17

Feather Wiki is a simple tool to create personal wikis on the web. Each wiki is self-contained within a single HTML file stored on your FreedomBox. Only FreedomBox users can read or write to the wiki files. It is also possible to "publish" a Feather Wiki file from within the application.

5.7.1 A Nest of Feather Wikis

Feather Wiki is a web application that runs entirely in the browser. FreedomBox implements a Feather Wiki Nest, i.e. an application that can store and manage several Feather Wiki files.

Feather Wiki is a web quine, like its more popular, older and more feature-rich cousin TiddlyWiki. Every wiki file contains an entire copy of the source code (HTML, CSS and JavaScript) along with your content, which makes it extremely portable.

The Nest is essentially a WebDAV server provided by FreedomBox's Apache web server, so Feather Wiki doesn't require installation of any additional packages. It requires no additional system resources either, since Apache is always running.

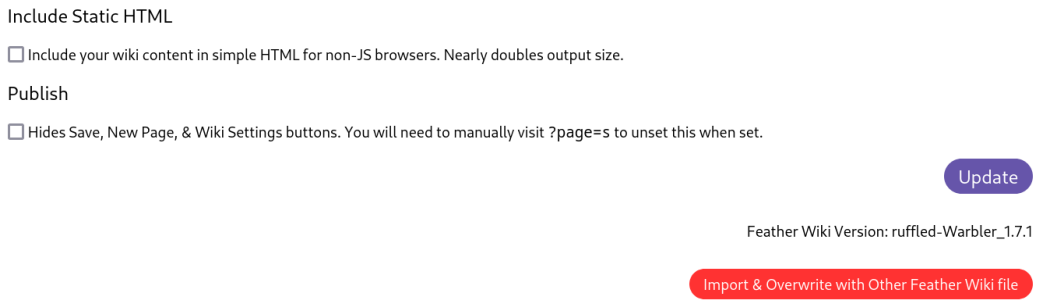
5.7.2 Setting up wikis

Creating and deleting wiki files is handled by the FreedomBox app itself. Every new wiki file is created from the latest available version of the empty Feather Wiki file downloaded from the official website. Deletion cannot be undone. You have to either restore the deleted file from a backup or import any saved local copy into a new wiki file. Uploading your existing Feather Wiki files is supported if they are at least v1.8.0 (Skylark). For older Feather Wiki files, you can use the import option as described below.

To import an old Feather Wiki file, first create a new wiki on your FreedomBox.The page called "Wiki Settings" in Feather Wiki provides a button that lets you import the contents of an existing wiki file into a new wiki by uploading the file from your local machine.

Upgrading Feather Wiki is a manual process that is very similar to importing.

1. Save a local copy of the wiki file.
2. Delete the wiki file from the server.
3. Create a new empty wiki file (this will be of the latest version by default).
4. Open the new wiki.
5. Import the contents of your local copy into the new wiki from the "Wiki Settings" page.



5.7.3 Publishing a wiki

The Feather Wiki app on FreedomBox doesn't provide a "public access" option, unlike MediaWiki. To share your wiki file with someone who doesn't have an account on your FreedomBox or to publish the file on the Internet, follow these steps:

1. Save a local copy of the wiki file.
2. Open the local copy in a web browser.
3. Go to "Wiki Settings" page and enable the "Publish" setting. (This makes the wiki non-editable. Hence the local copy.)
4. Save the local copy.

Now you can share the HTML file or publish it on the web using Apache userdir or placing it under the /var/www/html directory on your FreedomBox.

5.7.4 As many wikis as you need

The source code of Feather Wiki is tiny - less than 60 kilobytes. You are encouraged to create as many wiki files as you need, since the storage cost of each file is negligible, even on a FreedomBox running from an SD card.

The shortcut for Feather Wiki app opens a page with the list of wiki files stored on FreedomBox. Creating a front page shortcut for each wiki file (like Ikiwiki does) would clutter the front page, considering the number of Feather Wiki files people might create.

5.7.5 Use cases

- **Notebook:** A student can use Feather Wiki as a web notebook with one wiki file per course. About 40 files for a bachelor's degree!
- **Self-publishing:** An author can self-publish an online book using Feather Wiki as a GitBook alternative.
- **Refile:** One wiki file as a general notes dump that can be refiled later into more organized formats.
- One wiki file for each issue of a magazine (indie web zine) maybe with some custom CSS added.
- Simple internal websites for organizations to publish information.
- One Feather Wiki file for each of your projects. Keep all project-related notes and tasks in one file.
- Shipping documentation/user-guide with your product, in a single file.

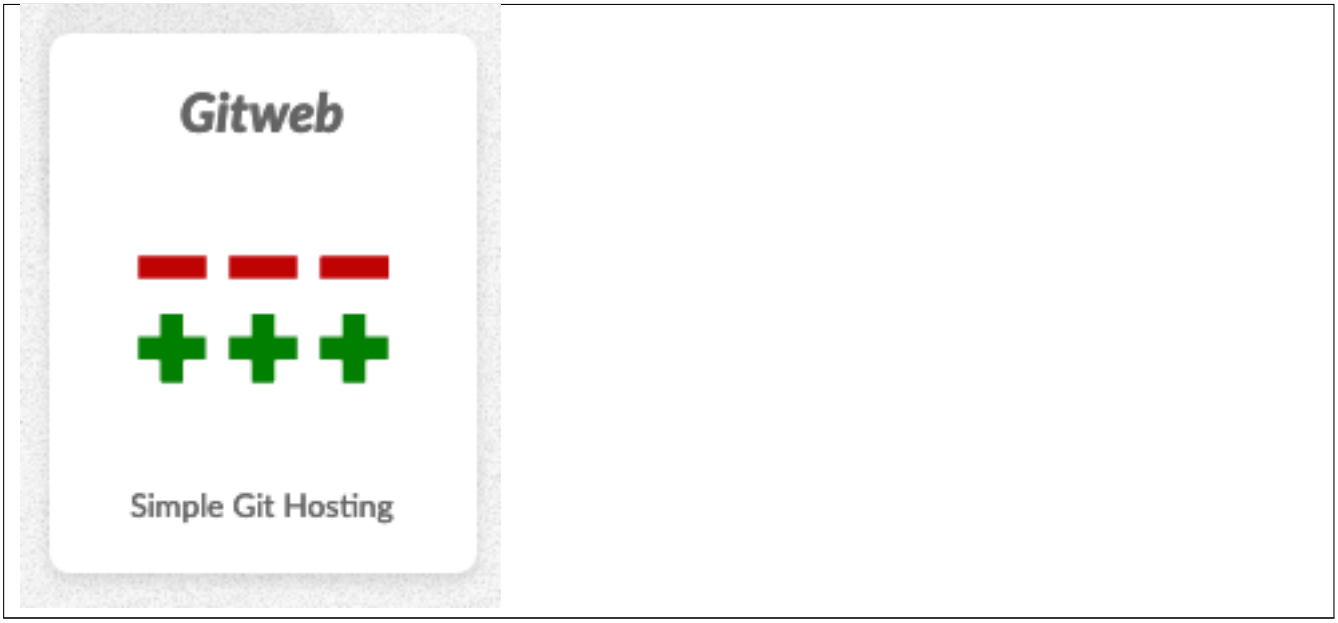
For more inspiration, see the Gallery on the official website.

You might want to configure your wiki with different extensions for Feather Wiki depending on the use case.

5.7.6 External links

- Website: `https://feather.wiki`
- Source code for Feather Wiki: `https://codeberg.org/Alamantus/FeatherWiki`

5.8 GitWeb (Simple Git Hosting)



Available since: version 19.19

5.8.1 What is Git

Git is a distributed version-control system for tracking changes in source code during software development. GitWeb provides a web interface to Git repositories. You can browse history and content of source code, use search to find relevant commits and code. You can also clone repositories and upload code changes with a command-line Git client or with multiple available graphical clients. And you can share your code with people around the world.

To learn more on how to use Git visit Git tutorial.

5.8.2 Managing the repositories

After installation of GitWeb, a new repository can be created. It can be marked as *private* to limit access.

5.8.3 Access

GitWeb can be accessed after installation e.g. by the web client through `https://<my_freedombox_name>/gitweb`.

5.8.4 HTTP basic auth

GitWeb on FreedomBox currently supports HTTP(S) remotes only (i.e. not SSH). To avoid having to enter the password each time you pull/push to the repository, you can edit your remote to include the credentials.

Example: `https://username:password@my.freedombox.rocks/gitweb/myrepo`

Your username and password will be encrypted. Someone monitoring the network traffic will notice the domain name only.

Note: If using this method, your password will be stored in plain text in the local repository's `.git/config` file. For this reason, you should create a FreedomBox user who has only access to the gitweb and never use an admin account.

For GNOME users (Advanced)

GNOME "Passwords and Keys" utility can be used to store the username and password. See this StackOverflow Answer for details on how to do it.

5.8.5 Mirroring

Though your repositories are primarily hosted on your own FreedomBox, you can configure a repository on another Git hosting system like GitLab as a mirror.

5.8.6 Enabling/Disabling Features

In FreedomBox, some of the default features of gitweb have been changed:

- Enabled: Blame feature to show what revision and author last modified each line of a file
- Enabled: Pickaxe feature to list the commits that introduced or removed a given string
- Enabled: Highlight feature to perform syntax highlighting on blobs
- Disabled: Snapshot feature that provides a download of a compressed tar file for a given revision (due to high resource usage).

These features can be changed on a per-repository bases by an administrator by editing the git configuration file for the repository on FreedomBox. See manual page for gitweb.conf(5) for more details on syntax and features. For example, to re-enable the snapshot feature on `myrepo` repository, login to a FreedomBox terminal via SSH or web console as administrator and edit the file `/var/lib/git/myrepo/config` to contain the following section:

```
[gitweb]
snapshot = tgz
```

5.8.7 External links

- Upstream project: https://git-scm.com/
- Upstream end user documentation: https://git-scm.com/docs/gitweb
- Debian community wikis: https://wiki.debian.org/CategoryGit

5.9 Home Assistant



Available since: version 25.10

Home Assistant is a home automation hub with emphasis on local control and privacy. It integrates with thousands of devices including smart bulbs, alarms, presence sensors, door bells, thermostats, irrigation timers, energy monitors, etc.

Please note that Home Assistant is installed and run inside a container provided by the Home Assistant project. Security, quality, privacy and legal reviews are done by the upstream project and not by Debian/FreedomBox. Updates are performed following an independent cycle.

Home Assistant app is considered experimental in FreedomBox as it is fairly new to our ecosystem. You may face issues working with it or the app may undergo major breaking changes.

5.9.1 Hardware

Home Assistant can detect, configure, and use various devices on the local network. For example, if a device is connected using Wi-Fi or LAN to the same network as FreedomBox, Home Assistant can detect, configure, and use the device. Other home automation protocols such as Thread, ZigBee, and Z-Wave are also supported but require additional hardware to be connected to your FreedomBox. For example, if you have a door sensor that speaks ZigBee, you need to attach a ZigBee USB dongle to your FreedomBox. Home Assistant can then detect and use the door sensor on the ZigBee network. **Note:** if you attach any such hardware to FreedomBox after you install the Home Assistant app, then you need to disable and re-enable the Home Assistant app before the hardware can be discovered and used by Home Assistant.

Home Assistant is a comprehensive solution for your home automation needs supporting thousands of devices. You can check whether a device is supported by Home Assistant by visiting the integrations page. Other devices which are not listed may also

be supported when they are similar to supported devices. It is recommended that you purchase hardware that advertises support for Home Assistant.

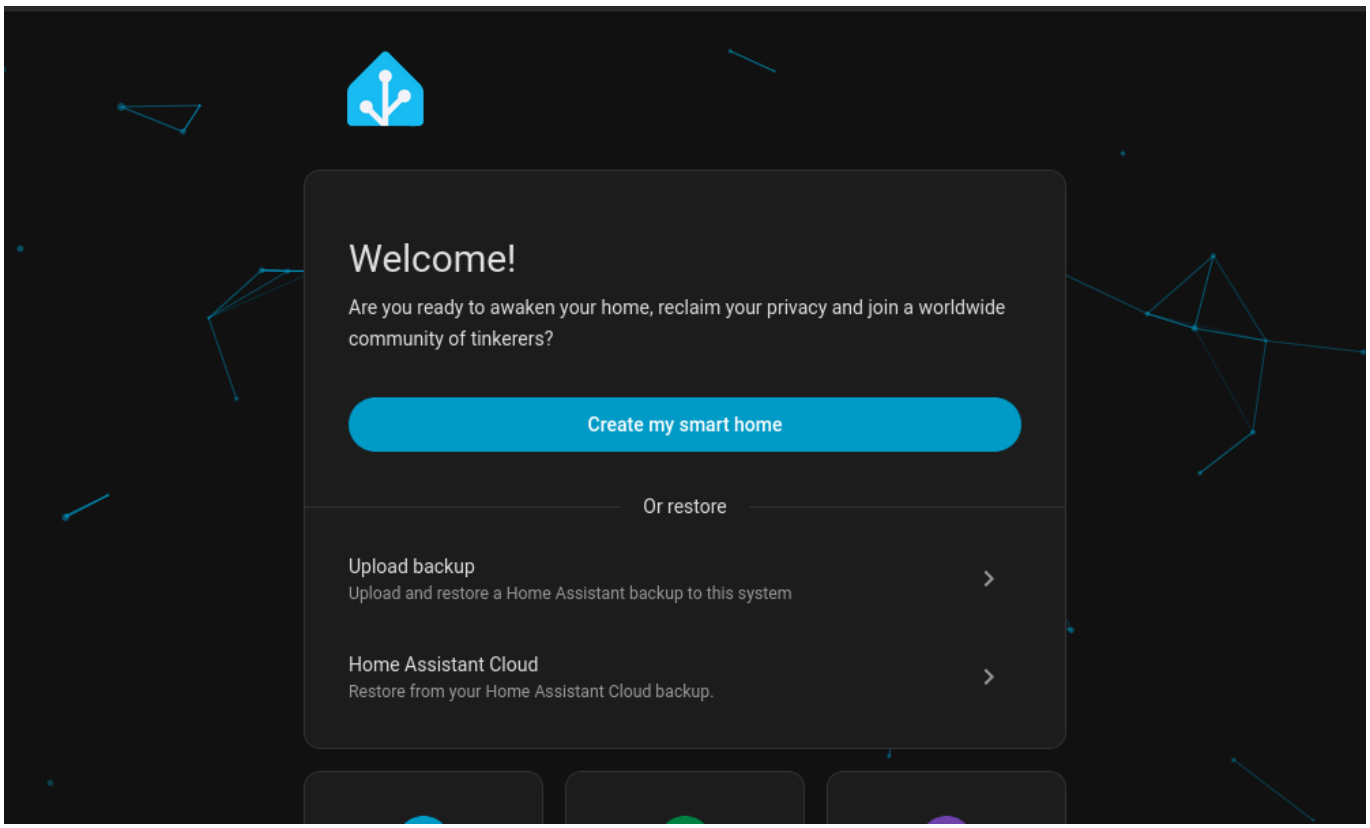
Many home automation devices connect to or send data to their manufacturers' servers. This can be a serious violation of your home's privacy. These devices may also stop working when their manufacturer goes bankrupt, decides to close the product line, or forces you to purchase newer models. It is strongly recommended that you purchase hardware that works locally with a home automation hub (in this case Home Assistant) without connecting to manufacturer's servers. These are listed under the categories "Local Push" or "Local Polling" in the Home Assistant's integrations page.

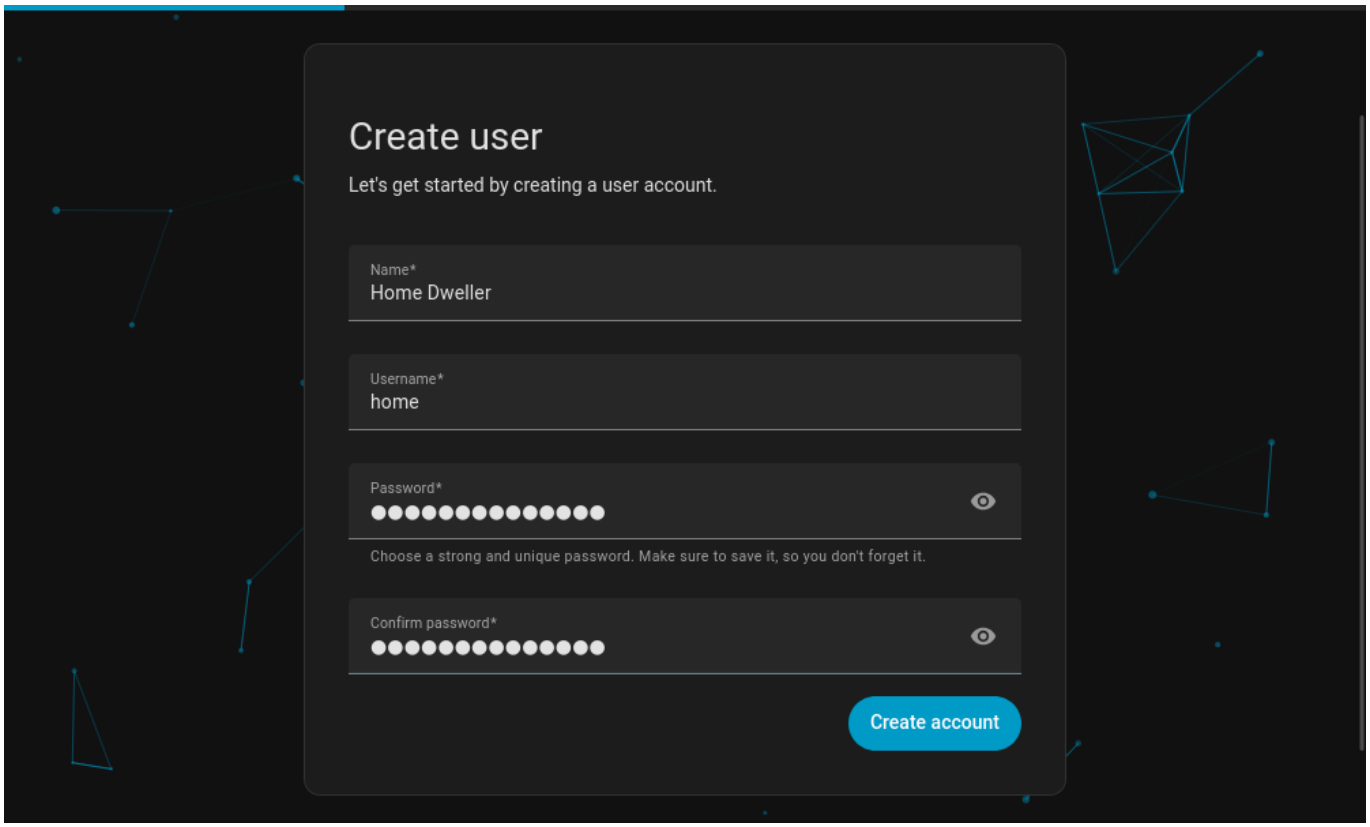
5.9.2 Need for Subdomain

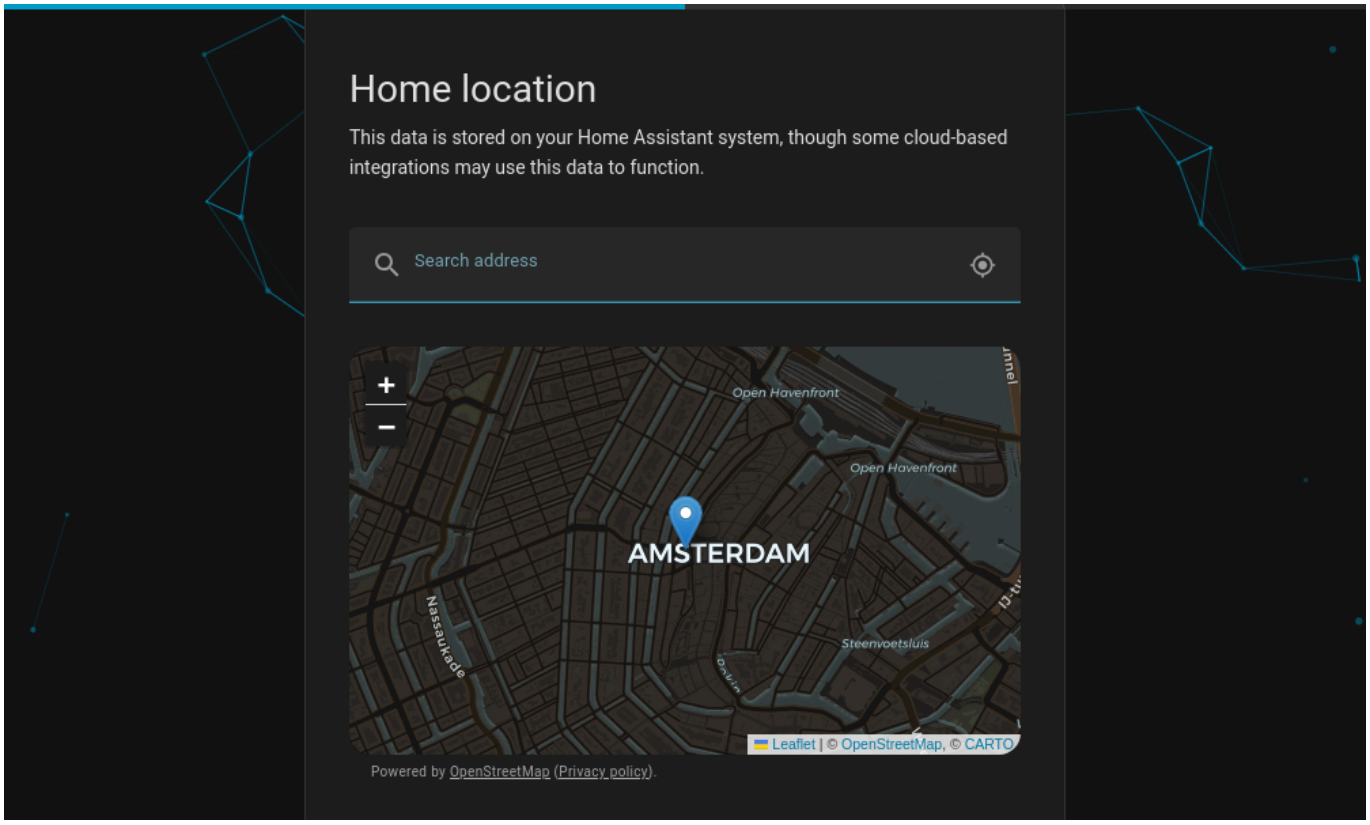
Unlike most other apps in FreedomBox which work on a path like `https://mydomain.example/app`, Home Assistant requires a separate dedicated subdomain such as `https://homeassistant.mydomain.example`. Before configuring the Home Assistant app, you must create a subdomain using System -> Name Services. See the documentation for Name Services to understand how you can acquire a subdomain.

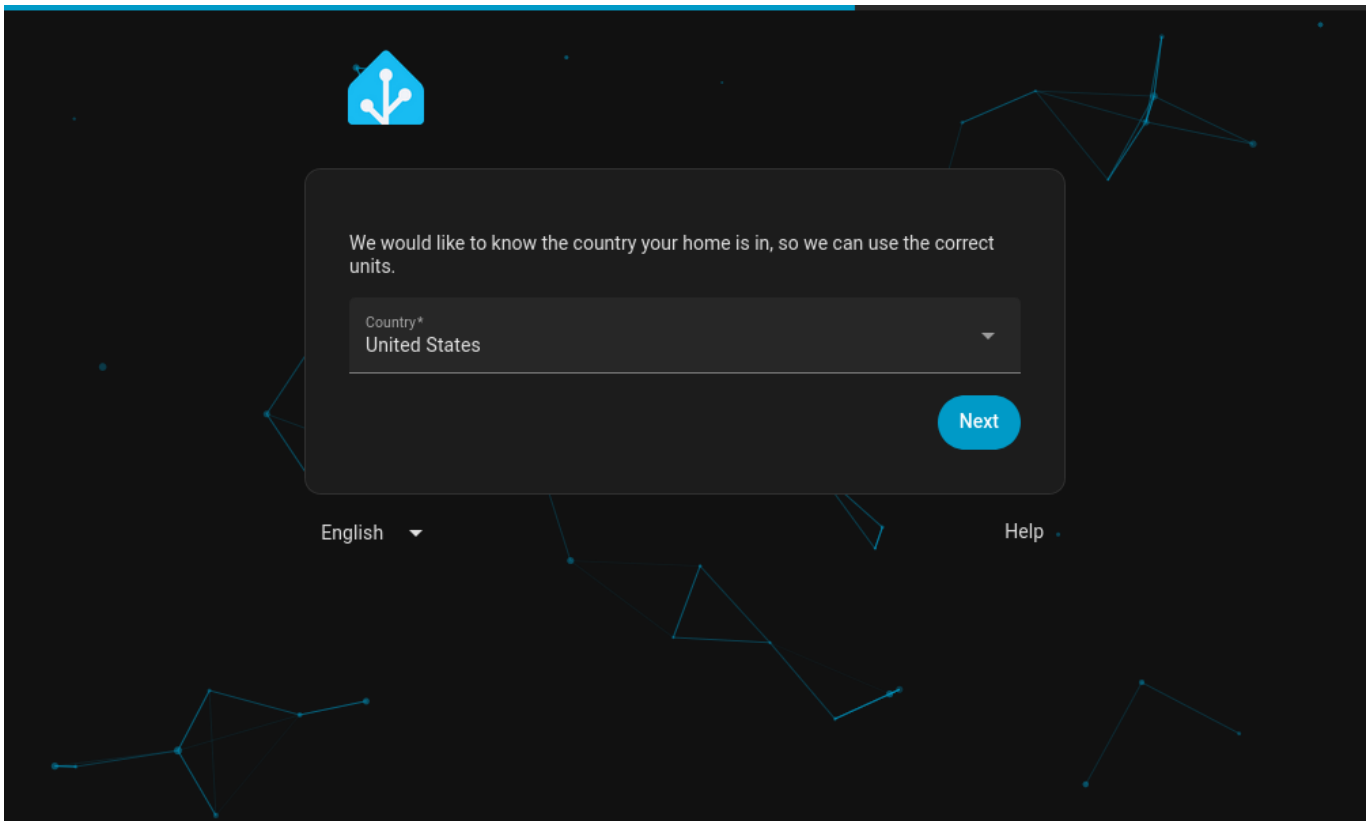
5.9.3 Setup

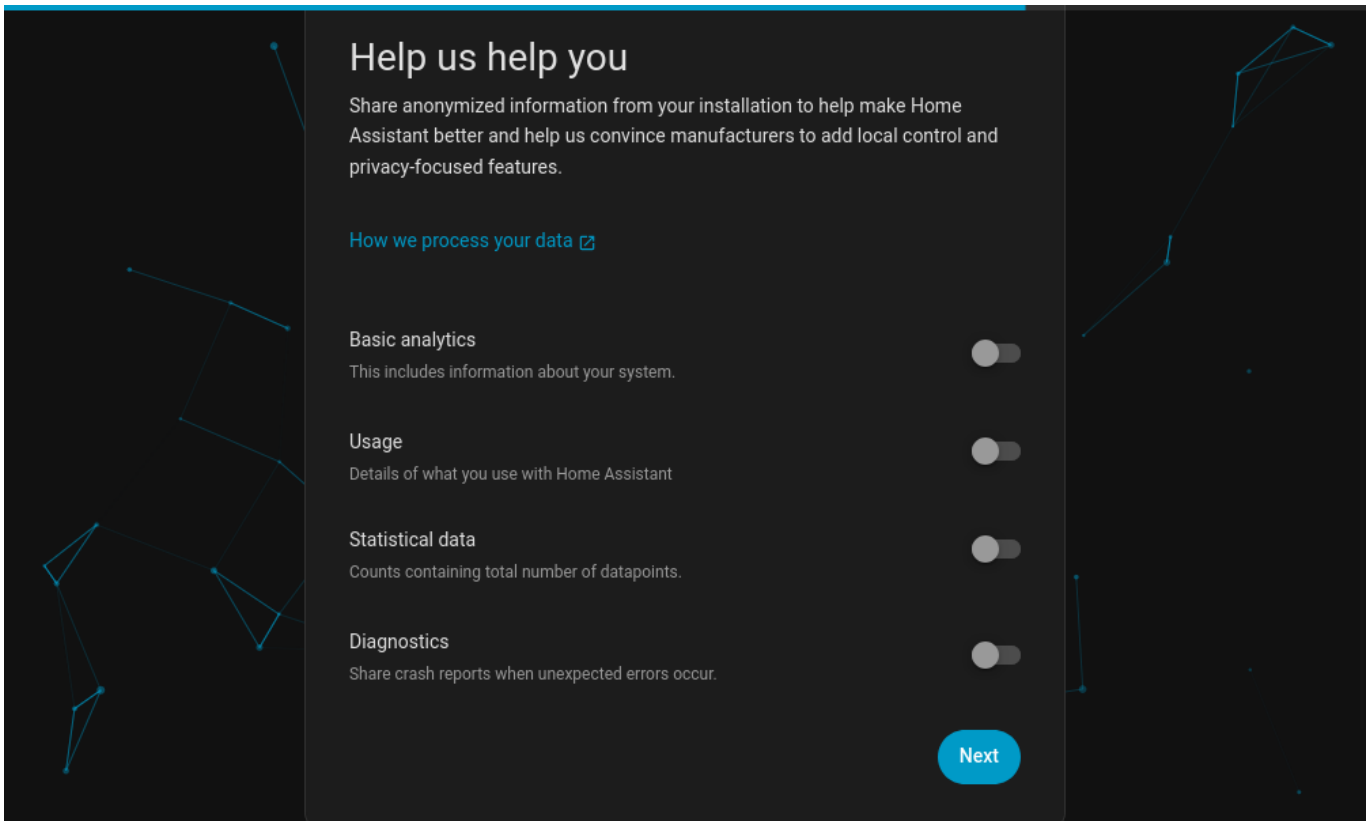
After the app is installed, Home Assistant web interface must be setup. An administrator account is created at this time. Home Assistant maintains its own user accounts separate from user accounts created on FreedomBox.




Create user
Let's get started by creating a user account.
Name*
Home Dweller
Username*
home
Password*
Choose a strong and unique password. Make sure to save it, so you don't forget it.
Confirm password*
Create account


Home location
This data is stored on your Home Assistant system, though some cloud-based integrations may use this data to function.
Search address
Open Havenfront
Open Havenfront
AMSTERDAM
Nassaukade
Steenvoetsluis
Leaflet | © OpenStreetMap, © CARTO
Powered by OpenStreetMap (Privacy policy).

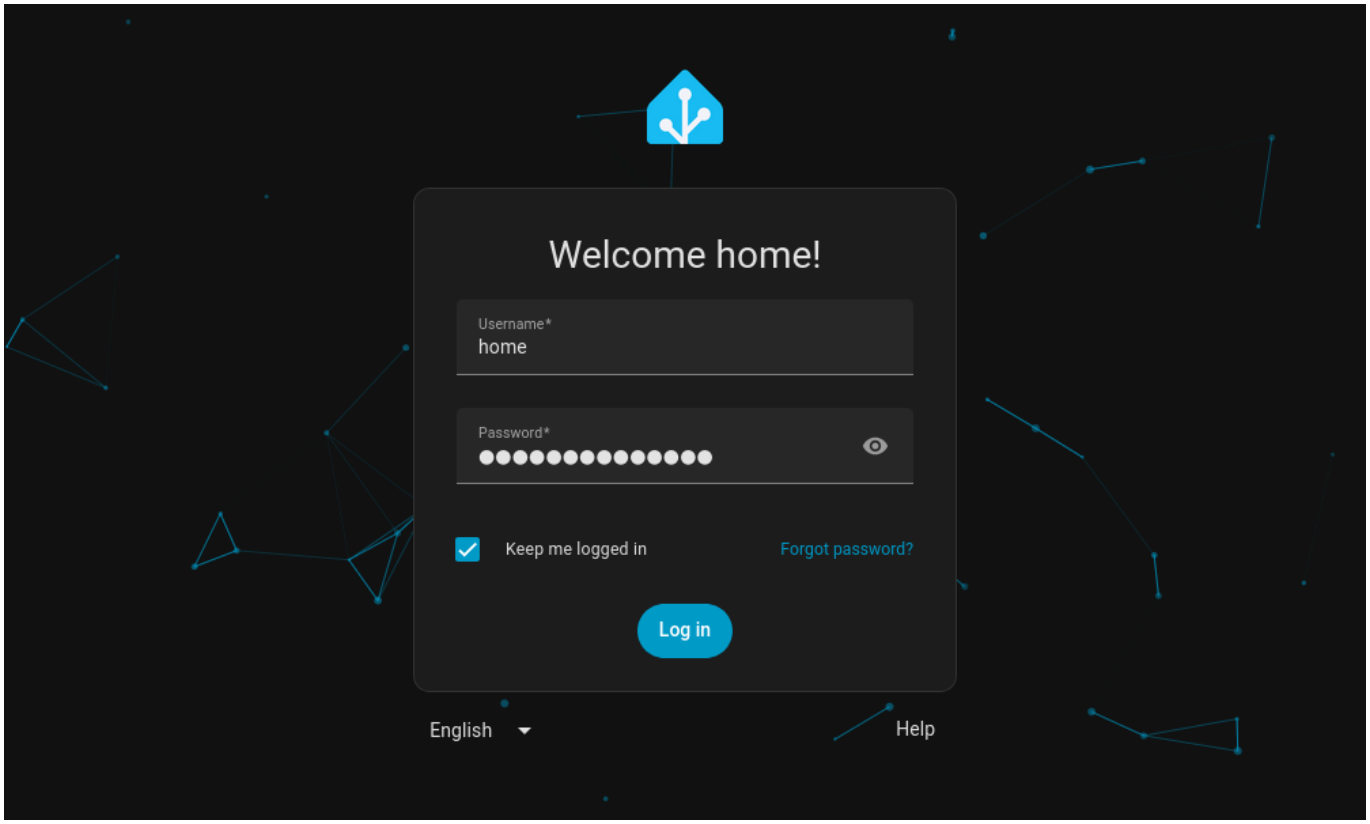


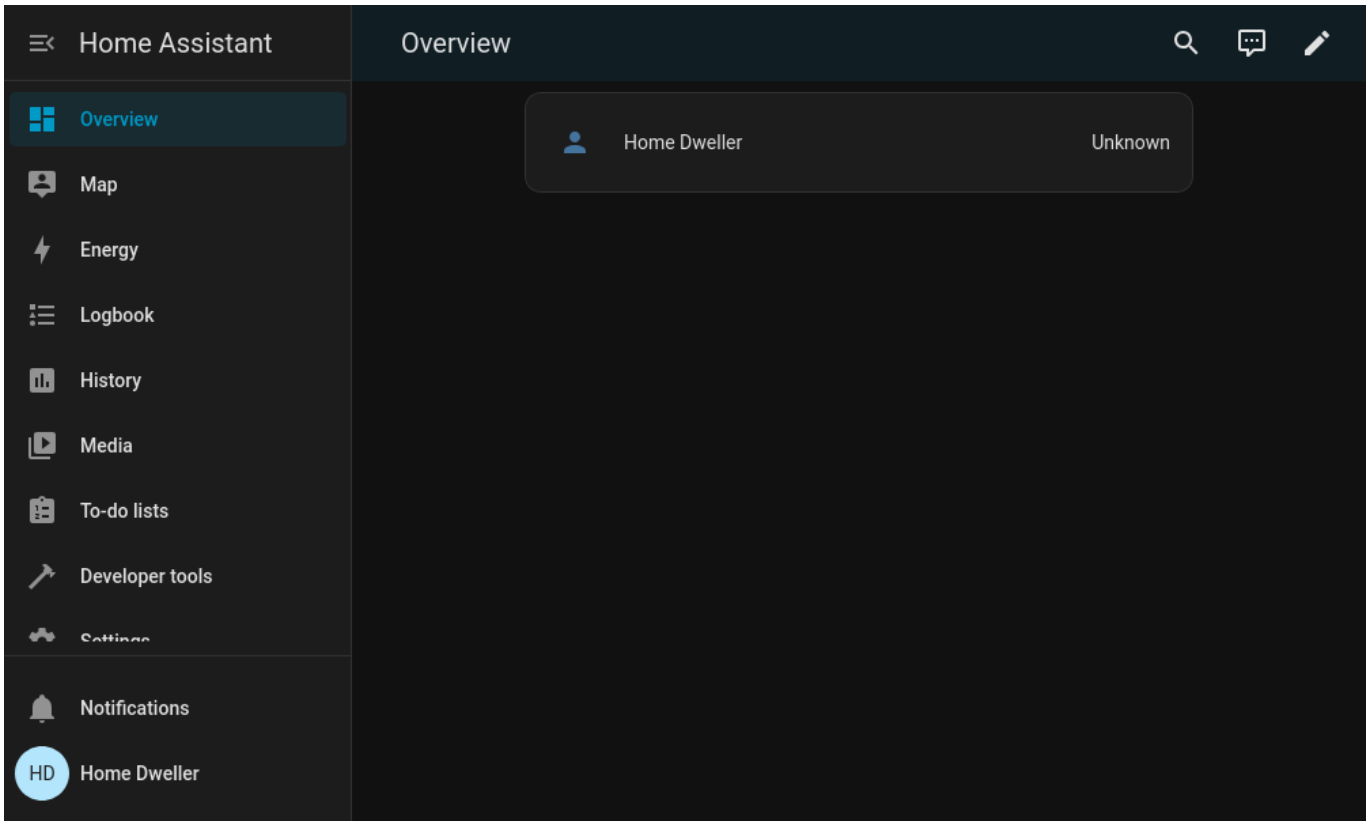


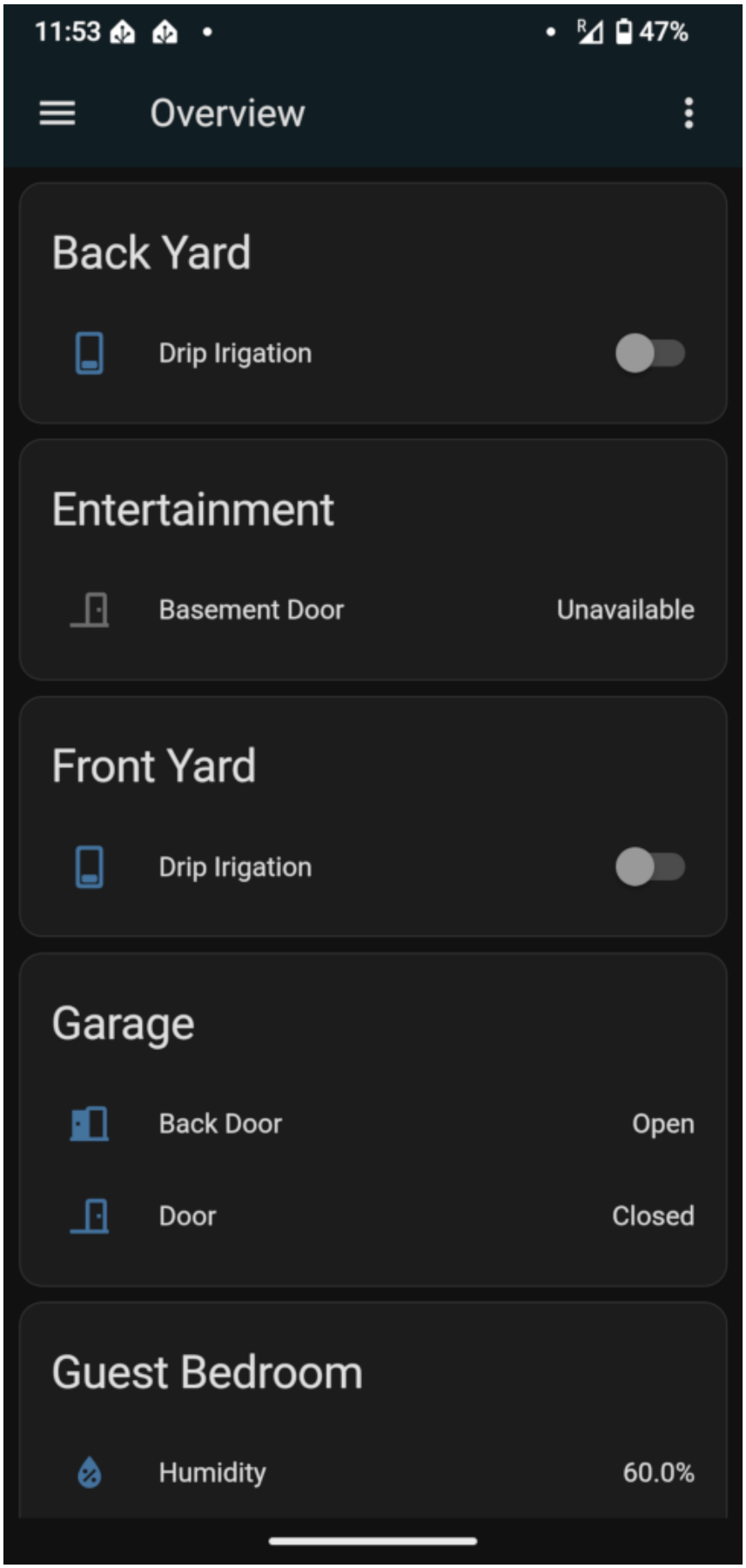
When new USB dongles are added to make Home Assistant talk to devices on networks like Thread, ZigBee, and Z-Wave, then setup operation must be re-run on the app. This operation is available in the Extras menu drop down with gears icon in the Home Assistant app page. Setup must also be re-run when the hardware is removed. Otherwise, Home Assistant service will fail to start.

5.9.4 Usage

Home Assistant provides a fully functional web application and a mobile application. Simply login and start using:






11:53
47%
Overview
Back Yard
Drip Irigation
Entertainment
Basement Door
Unavailable
Front Yard
Drip Irigation
Garage
Back Door
Open
Door
Closed
Guest Bedroom
Humidity
60.0%

5.9.5 Advanced Manual Configuration

If you wish to perform your own customization to Home Assistant setup, feel free to edit Home Assistant configuration files manually. Configuration, scenes, automations, scripts, secrets, database, and log files are all stored in the directory `/var/lib/home-assi` directory. You will need to disable and re-enable the app after configuration changes for them to take effect. Note that when FreedomBox itself modifies these files, it will not preserve comments and other formatting.

5.9.6 External links

- Website: https://www.home-assistant.io/
- Source code for Home Assistant: https://github.com/home-assistant/core
- Source code for Home Assistant container image: https://github.com/home-assistant/docker

5.10 Ikiwiki (Wiki and Blog)



Avaiable since: version 0.5

5.10.1 What is Ikiwiki?

Ikiwiki converts wiki pages into HTML pages suitable for publishing on a website. It provides particularly blogging, podcasting, calendars and a large selection of plugins.

5.10.2 Creating a wiki or blog

You can create a wiki or blog to be hosted on your FreedomBox through the Wiki & Blog (Ikiwiki) page in FreedomBox. The first time you visit this page, it will ask to install packages required by Ikiwiki.



After the package install has completed, select the "Create Wiki or Blog" button.

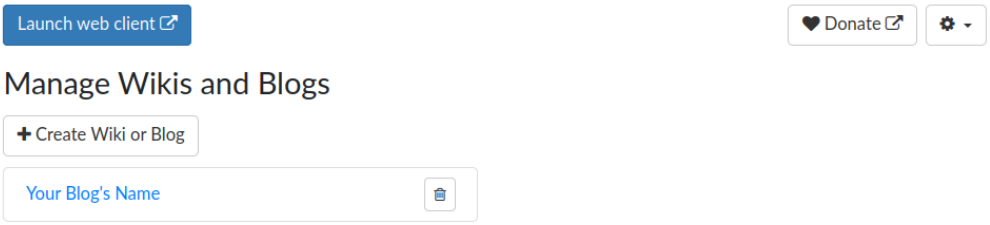
You can select the type to be Wiki or Blog. Also type in a name for the wiki or blog, and the username and password for the wiki's/blog's admin account. Then click Update setup and you will see the wiki/blog added to your list. Note that each wiki/blog



has its own admin account.

5.10.3 Accessing your wiki or blog

Your wikis and blogs are listed in the Ikiwiki app. Clicking on your site's name will bring you to its start page.



From here, if you click Edit or Preferences, you will be taken to a login page. To log in with the admin account that you created before, select the Other tab, enter the username and password, and click Login.

5.10.4 User login through SSO

Besides the wiki/blog admin, other FreedomBox users can be given access to login and edit wikis and blogs. However, they will not have all the same permissions as the wiki admin. They can add or edit pages, but cannot change the wiki's configuration.

To add a wiki user, go to the Users and Groups page in FreedomBox (under System configuration, the gear icon at the top right corner of the page). Create or modify a user, and add them to the wiki group. (Users in the admin group will also have wiki access.)

To login as a FreedomBox user, go to the wiki/blog's login page and select the Other tab. Then click the "Login with HTTP auth" button. The browser will show a popup dialog where you can enter the username and password of the FreedomBox user.

5.10.5 Adding FreedomBox users as wiki admins

1. Login to the site, using the admin account that was specified when the site was created.
2. Click "Preferences", then "Setup".
3. Under "main", in the "users who are wiki admins", add the name of a user on the FreedomBox.
4. (Optional) Under "auth plugin: passwordauth", uncheck the "enable passwordauth?" option. (Note: This will disable the old admin account login. Only SSO login using HTTP auth will be possible.)
5. At the bottom of the page click "Save Setup".
6. Click "Preferences", then "Logout".
7. Login as the new admin user using "Login with HTTP auth".

5.10.6 Avoiding Spam

By default, every wiki page also has a "Discussion" page, which can be edited anonymously, without logging in. To avoid spam, you may want to disable the Discussion feature all together, by unchecking the "enable Discussion pages?" option in the setup.

5.10.7 Theming

1. Login to the site, using the admin account that was specified when the site was created.
2. Click "Preferences", then "Setup".
3. Under "web plugin: theme", check "enable theme?"
4. Right under the checkbox, type in the name of the desired theme. You can choose from the following officially supported themes:
   - actiontabs - mobile friendly
   - blueview - non-mobile friendly
   - goldtype - non-mobile friendly
   - monochrome - mobile friendly
5. At the bottom of the page click "Save Setup".

For your changes to become visible, you might have to delete your browser's cache or wait a few minutes and refresh your ikiwiki's page.

It is also possible to install user-contributed themes from ikiwiki's Theme Market. Please note, that this requires additional technical knowledge.

5.10.8 External links

- Upstream project: https://ikiwiki.info
- Upstream project user documentation: https://ikiwiki.info
  - Upstream project user forum: https://ikiwiki.info/forum/
  - Upstream project theme market https://ikiwiki.info/theme_market/

5.11 Infinoted (Collaborative text edition with Gobby)



Available since: version 0.5

infinoted is a server for Gobby, a collaborative text editor.

To use it, download Gobby desktop client and install it. Then start Gobby and select "Connect to Server" and enter your FreedomBox's domain name.

5.11.1 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for infinoted:

- TCP 6523

5.11.2 Extenal links

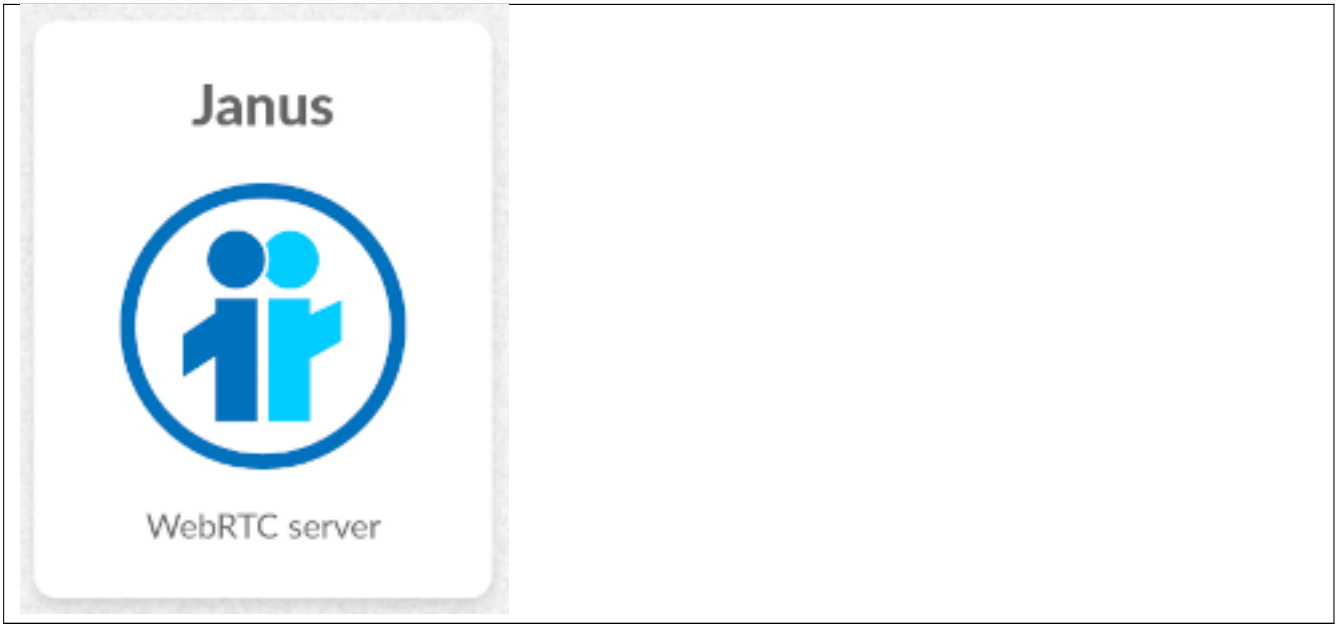
- Upstream project: https://gobby.github.io/libinfinity/
- Upstream end user documentation: https://github.com/gobby/gobby/wiki

5.11.2.1 Client apps

Client software from the Infinoted team is named, "Gobby." Links to recommended client apps, where applicable, can be found in FreedomBox by select the Apps page, selecting the application tile, and clicking the > Client Apps button.

- Gobby Client Software: https://gobby.github.io/
- Gobby Wiki: https://github.com/gobby/gobby/wiki

5.12 Janus (WebRTC server)



Available since version: 22.13

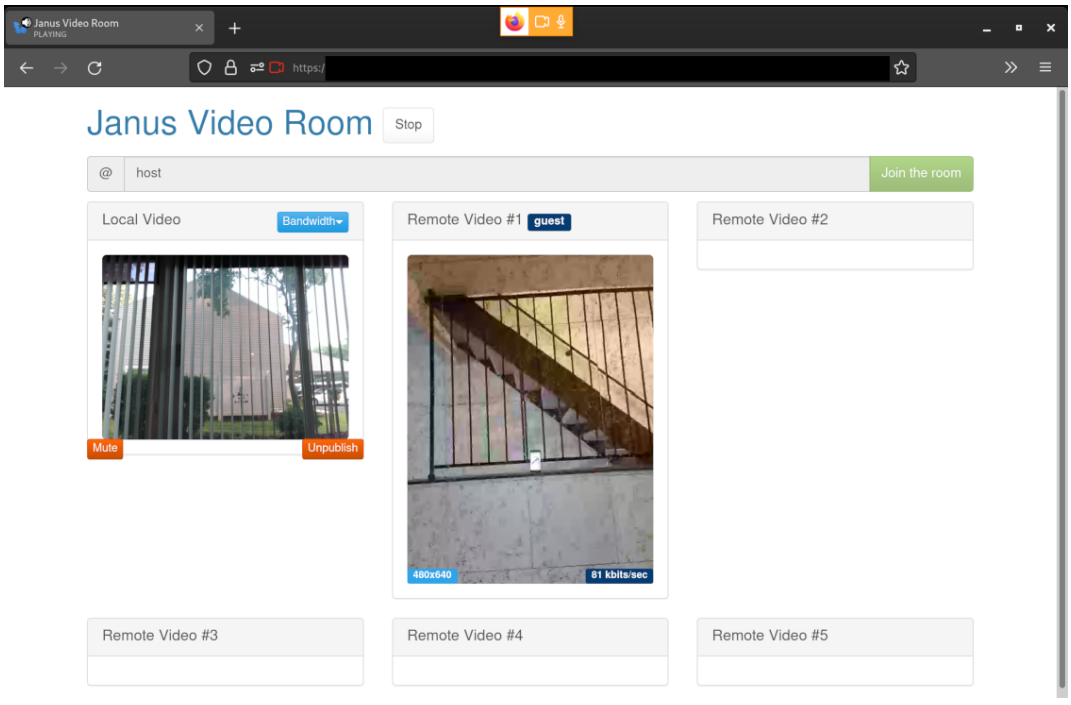
Janus is a lightweight, general purpose WebRTC server. It can support different kinds of real-time communication apps, such as video chat and streaming.

Currently, in FreedomBox, a simple video conference room is included with Janus. This video room can be accessed by anyone who visits your FreedomBox; it does not require logging in with a user account.

In the future, the simple video room app will be replaced by Jangouts, a fully-featured video conference app.

Coturn is required to use Janus, so it also needs to be installed and running on your FreedomBox.

5.12.1 Screenshot



5.12.2 Using Janus

The Janus shortcut will take you to the Janus Video Room page. From here, press the Start button at the top of the page.

Next, you will need to provide a display name. Any name can be used here. Press the "Join the room" button to enter the room.

The first time you enter the video room, your web browser will ask if this page has permission to access your camera and microphone. Press "Allow" to proceed.

Your own video will be displayed in the "Local Video" window. From here you can mute your audio, or use unpublish to stop sharing your video and audio. If other people join the video room, they will appear in the "Remote Video" windows.

5.12.3 External links

- Upstream project: https://janus.conf.meetecho.com
- Upstream end user documentation: https://janus.conf.meetecho.com/docs

5.13 JSXC (Web Chat Client)



Available since: version 0.11.0

JSXC is a web chat client. It can be used to join compatible chat servers.

FreedomBox offers both parties, a server (ejabberd) and a web client (JSXC), from its web interface.

5.13.1 Technical Specifications

JSXC features the XMPP over BOSH protocol and is implemented in HTML5.

XMPP is a federated server-client protocol for Instant Messaging. This means that users who have accounts on one server, can talk to users that are on another server.

XMPP can also be used for voice and video calls, if supported by the clients.

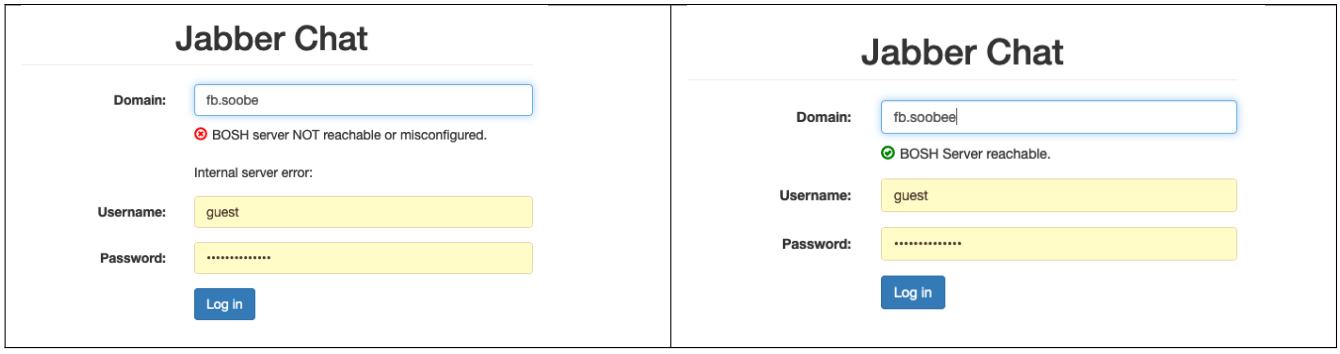
5.13.2 Installation

You can install JSXC through its icon in the Apps section of FreedomBox web interface. The ejabberd (XMPP server) icon also offers to launch the web client (and installs JSXC if not yet installed).

5.13.3 Usage

After the JSXC module install completes, the JSXC can be accessed through its icon in the Apps section of FreedomBox web interface. The ejabberd (XMPP server) icon also offers to launch the web client. Both will redirect you to `https://<your freedombox>/plinth/apps/xmpp/jsxc/`.

To use it, you need to input the domain name of the server to connect to. It will automatically check the BOSH server connection to the given domain name as you type it.



Videoconferencing and file transfer features are offered by JSXC but don't seem to work in FreedomBox yet.

5.13.4 Port Forwarding

If your FreedomBox is behind a router and you want to connect to other servers, you will need to set up port forwarding on your router. You should forward the following ports for XMPP:

- TCP 5222 (client-to-server)

5.13.5 External links

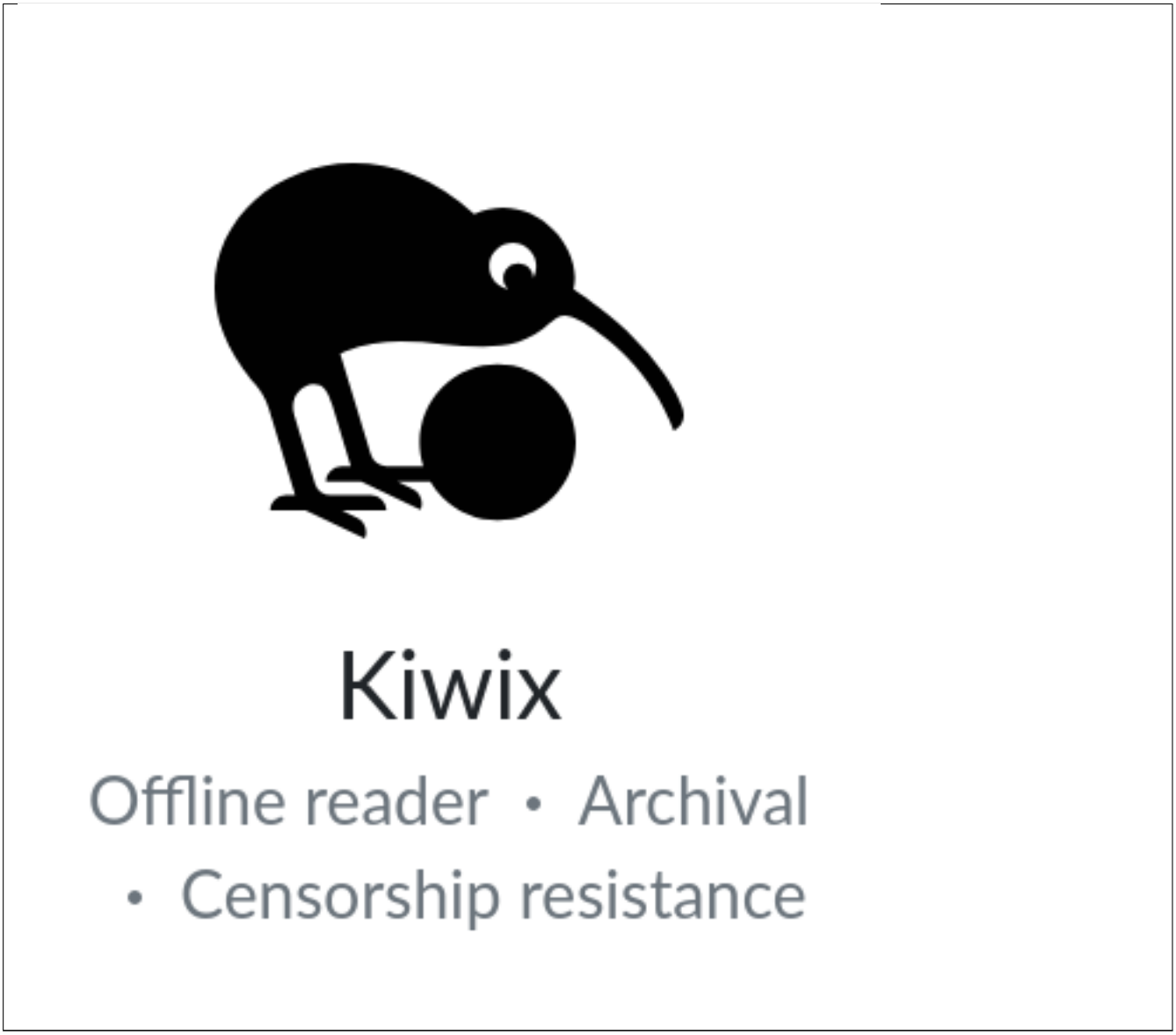
- Upstream project: `https://www.jsxc.org`
- Upstream end user documentation: `https://www.jsxc.org/manual.html`

5.13.5.1 Client apps

JSXC is the client app used with ejabberd server included in FreedomBox.

5.14 Kiwix (Offline Wikipedia)

Available since: version 23.19



Kiwix can serve archives of websites offline. It was originally created to serve an offline version of Wikipedia, the free encyclopedia.

5.14.1 Use Cases

FreedomBox for Communities was already using Kiwix years before it was officially included as an app. Kiwix was used to serve content over community Wi-Fi networks in villages of India with unreliable or insufficient Internet access. In addition to an offline Wikipedia, Kiwix served

- the archive "Wikipedia for Schools" which was used by school teachers and students for homework assignments and as a teaching aid.
- the archive "WikiSource (in Telugu)" which served as a digital library.

Kiwix is also useful when you have reliable Internet access.

1. Archival and preservation: Archives of websites can be created and preserved as ZIM files. These archives can be served using Kiwix.

2. Fast access to slow websites: An offline archive can be faster to access than a website.
3. Access to websites that can be temporarily unavailable. For example, the Low Tech Magazine is hosted on a solar-powered single-board computer and could go offline sometimes. There is a Kiwix archive available for it which can be hosted locally.

5.14.2 Kiwix on FreedomBox

FreedomBox serves the Debian package called kiwix-tools which allows managing and serving content archives in ZIM file format. ZIM files can be added or removed by a FreedomBox administrator or any user in the "kiwix" user group.

Kiwix on FreedomBox is available via web interface only. There are mobile and desktop clients for Kiwix, but they can only read local archives. They cannot pull content from Kiwix hosted on FreedomBox.

The web interface of Kiwix is available for public access by default.

5.14.3 Finding content archives

To find content for Kiwix, go to library.kiwix.org. You can download archives (called "books") using direct download or using BitTorrent. The website itself is a Kiwix installation, so you can have a preview of the content before downloading, to check that the archive is not broken.

An alternative source of the same files is wiki.kiwix.org, a wiki with download links and more information about the Kiwix project.

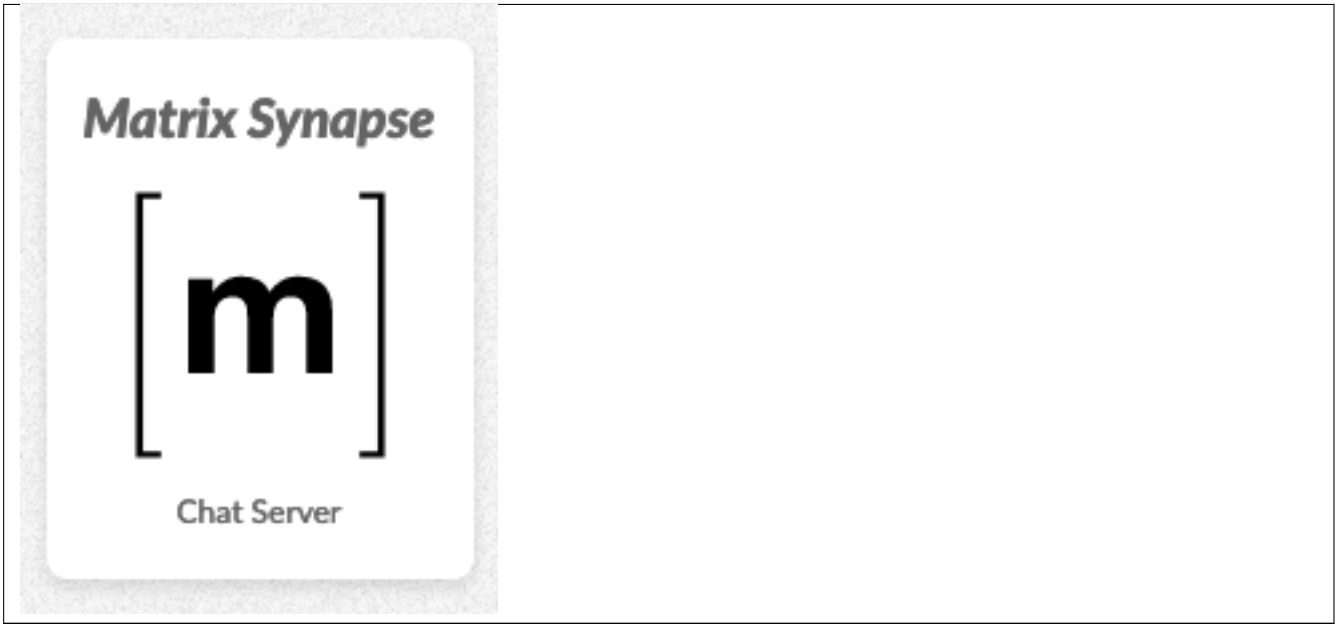
5.14.4 Creating content archives

The OpenZim project maintains tools to create ZIM files. The tool zimit is a complete solution that can scrape a website and create a ZIM file. You can also request a ZIM archive using the website youzim.it.

5.14.5 External links

- Upstream project: `https://kiwix.org/`
- Upstream documentation: `https://kiwix.org/frequently-asked-questions/`

5.15 Matrix Synapse (Chat Server)



Available since: version 0.14.0

5.15.1 What is Matrix?

Matrix is an open protocol for interoperable, decentralized, real-time communication over IP. Synapse is the reference implementation of a Matrix server. It can be used to setup instant messaging on FreedomBox to host chat rooms with end-to-end encrypted communication and audio/video calls. Matrix Synapse is a federated application where chat rooms can exist on any server and users from any server in the federated network can join them. Learn more about Matrix.

5.15.2 How to access your Matrix Synapse server?

We recommend the Element client to access the Matrix Synapse server. You can download Element for desktops. Mobile applications for Android and iOS are available from their respective app stores.

5.15.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Matrix:

- TCP 8448

5.15.4 Setting up Matrix Synapse on your FreedomBox

To enable Matrix Synapse, first navigate to the Chat Server (Matrix Synapse) page and install it. Matrix needs a valid domain name to be configured. After installation, you will be asked to configure it. You will be able to select a domain from a drop down menu of available domains. Domains are configured using System -> Configure page. After configuring a domain, you will see that the service is running. The service will be accessible on the configured FreedomBox domain. Currently, you will not be able to change the domain once is it configured.

Your router has to be configured to forward port 8448.

All the registered users of your FreedomBox will have their Matrix IDs as `@username:domain`. If public registration is enabled, also your chosen client can be used to register a user account.

5.15.5 Setting up Audio/Video calls

The Matrix Synapse server is only responsible for establishing calls between participants in rooms. Matrix clients such as Element are actually responsible for the transfer of the audio/video traffic. Element supports calling in both one-to-one conversations and in groups.

For one-to-one conversations, Element tries to make a peer-to-peer connection between the two participants. This works when both the participants are using Element on computers with a public IP address or if they're on the same network. If both the participants are behind different NAT devices, establishing a direct peer-to-peer connection between them will not be possible. This problem can be solved by configuring Matrix Synapse with a STUN/TURN server. FreedomBox provides an app called Coturn for this purpose. FreedomBox doesn't automatically install Coturn on installing Matrix Synapse. However, as soon as Coturn app is installed, FreedomBox automatically configures Matrix Synapse to use it for audio/video calls. It is possible to override this configuration with a different STUN/TURN server in the web interface.

For calling groups with more than two participants (i.e. not one-on-one conversations), Element uses an external Jitsi Meet instance. Element uses jitsi.riot.im as its default Jitsi Meet instance. If the Matrix Synapse server is configured to use a different Jitsi Meet instance as the default, Element will use it instead for all users on that server.

5.15.6 Federating with other Matrix instances

You will be able to interact with any other person running another Matrix instance. This is done by simply starting a conversation with them using their matrix ID which is of the format `@their-username:their-domain`. You can also join rooms which are in another server and have audio/video calls with contacts on other server.

5.15.7 Memory usage

The Synapse reference server implemented in Python is known to be quite RAM hungry, especially when loading large rooms with thousands of members like #matrix:matrix.org. It is recommended to avoid joining such rooms if your FreedomBox device only has 1 GiB RAM or less. Rooms with up to a hundred members should be safe to join. The Matrix team is working on a new implementation of the Matrix server written in Go called Dendrite which might perform better in low-memory environments.

Some large public rooms in the Matrix network are also available as IRC channels (e.g. #freedombox:matrix.org is also available as #freedombox on irc.debian.org). It is better to use IRC instead of Matrix for such large rooms. You can join the IRC channels using Quassel.

5.15.8 Advanced usage

1. If you wish to create a large number of users on your Matrix Synapse server, use the following commands on a remote shell as root user:

   ```
   cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 32 | head -n 1 | sed "s+^+ ↩
       registration_shared_secret: +" > /etc/matrix-synapse/conf.d/ ↩
       registration_shared_secret.yaml
   chmod 600 /etc/matrix-synapse/conf.d/registration_shared_secret.yaml
   chown matrix-synapse:nogroup /etc/matrix-synapse/conf.d/registration_shared_secret. ↩
       yaml
   systemctl restart matrix-synapse
   register_new_matrix_user -c /etc/matrix-synapse/conf.d/registration_shared_secret. ↩
       yaml
   ```

2. If you wish to see the list of users registered in Matrix Synapse, the following as root user:

   ```
   apt install sqlite3
   echo 'select name from users' | sqlite3 /var/lib/matrix-synapse/homeserver.db
   ```

3. If you wish to create a community in Matrix Synapse, a Matrix user with server admin privileges is needed. In order to grant such privileges to `username` run the following commands as root user:

- ```
  sudo apt install sqlite3
  echo "UPDATE users SET admin=1 WHERE name='@username:domainname'" | sudo sqlite3 /var ↩
      /lib/matrix-synapse/homeserver.db
  ```

5.15.9 External links

- Upstream project: https://matrix.org
- Upstream Synapse matrix server project: https://github.com/element-hq/synapse
- Upstream user documentation: https://matrix.org/docs/chat_basics/matrix-for-im/
- Debian community wiki: https://wiki.debian.org/Matrix
- Video tutorial for setting up Matrix Synapse on a Cloud instance: https://youtu.be/8snpMHHbymI

5.15.9.1 Client apps

Links to recommended client apps, where applicable, can be found in FreedomBox by select the Apps page, selecting the application tile, and clicking the > Client Apps button.

- Clients in Debian community wiki

5.16 MediaWiki (Wiki)



Available since: version 0.20.0

5.16.1 About MediaWiki

MediaWiki is the software that powers the Wikimedia suite of wikis.

Read more about MediaWiki on Wikipedia
  ```

5.16.2 MediaWiki on FreedomBox

MediaWiki on FreedomBox is configured to be publicly readable and privately editable. Only logged in users can make edits to the wiki. This configuration prevents spam and vandalism on the wiki.

5.16.2.1 User management

Users can be created by the MediaWiki administrator (user "admin") only. The "admin" user can also be used to reset passwords of MediaWiki users. The administrator password, if forgotten can be reset anytime from the MediaWiki app page in web interface.

5.16.2.2 Use cases

MediaWiki is quite versatile and can be put to many creative uses. It also comes with a lot of plugins and themes and is highly customizable.

5.16.2.2.1 Personal Knowledge Repository

- MediaWiki on FreedomBox can be your own personal knowledge repository. Since MediaWiki has good multimedia support, you can write notes, store images, create checklists, store references and bookmarks etc. in an organized manner. You can store the knowledge of a lifetime in your MediaWiki instance.

5.16.2.2.2 Community Wiki

- A community of users can use MediaWiki as their common repository of knowledge and reference material. It can used as a college notice board, documentation server for a small company, common notebook for study groups or as a fan wiki like wikia.

5.16.2.2.3 Personal Wiki-based Website

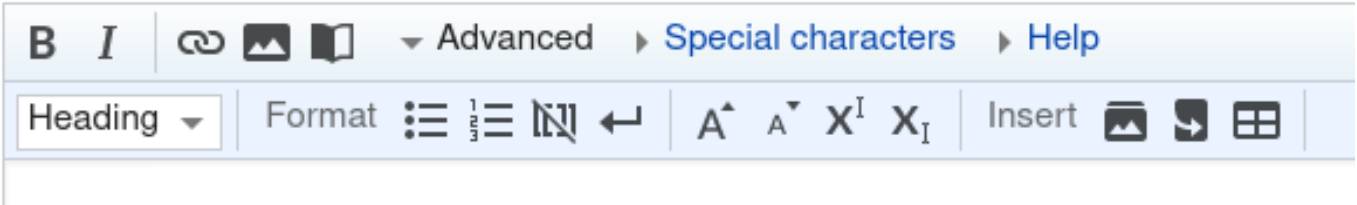
- Several websites on the internet are simply MediaWiki instances. MediaWiki on FreedomBox is read-only to visitors. Hence, it can be adapted to serve as your personal website and/or blog. MediaWiki content is easy to export and can be later moved to use another blog engine.

5.16.2.3 Editing Wiki Content

The MediaWiki installation on FreedomBox ships with two kinds of editors - WikiText editor and Visual editor.

5.16.2.3.1 WikiText Editor

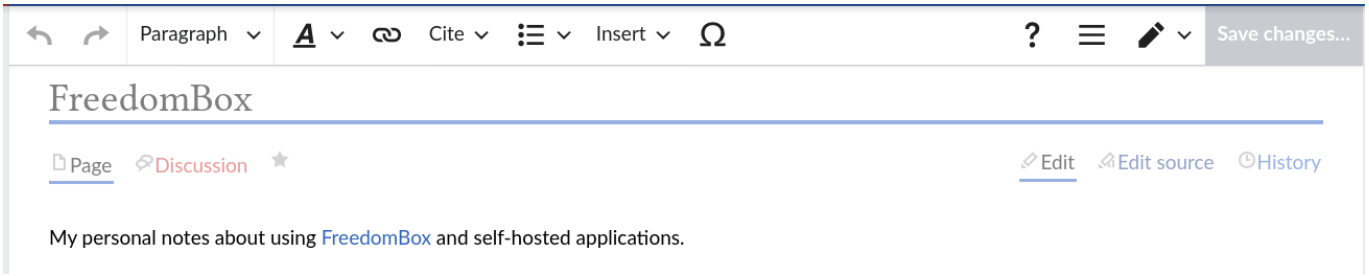
- This editor is for editing the wiki directly in MediaWiki's markup language. It has a toolbar for common options like Bold, Italics etc. Click on the Advanced section for more options like Headings, bullet lists etc.



5.16.2.3.2 Visual Editor

- MediaWiki's VisualEditor extension provides a WYSIWYG interface to editing wiki pages. This extension is bundled with MediaWiki from 1.35 and is enabled by default from FreedomBox 21.9.

  Since this is essentially a rich-text editor, knowledge of MediaWiki's markup language is not required. To use advanced features not available in the VisualEditor (yet), switch back to source editing.



5.16.2.3.3 Other Formats

- You don't have to necessarily learn the MediaWiki formatting language. You can write in your favorite format (Markdown, Org-mode, LaTeX etc.) and convert it to the MediaWiki format using Pandoc.

5.16.2.3.4 Image Uploads

- Image uploads have been enabled since FreedomBox version 0.36.0. You can also directly use images from Wikimedia Commons using a feature called Instant Commons.

5.16.2.4 Customization

5.16.2.4.1 Skins

MediaWiki's default skin is usually Vector. The default skin set by FreedomBox is Timeless.

Vector is a skin best-suited for viewing on desktop browsers. It is not suitable for mobile screen sizes. Wikimedia sites host a separate mobile site. It is not worth hosting a separate mobile site for small MediaWiki installations like those on FreedomBox. Using a mobile-friendly skin like Timeless is a cheaper way of solving the problem.

Administrators can choose a default skin from the app configuration. Users of the site also have the choice of viewing it with a different skin.

5.16.3 External links

- Upstream Project: `https://www.mediawiki.org/wiki/MediaWiki`
- User documentation: `https://www.mediawiki.org/wiki/Help:Contents`
- Debian MediaWiki wiki page: `https://wiki.debian.org/MediaWiki`

5.17 Minetest (Block Sandbox)



Available since: version 0.9

Minetest is a multiplayer infinite-world block sandbox. This module enables the Minetest server to be run on this FreedomBox, on the default port (30000). To connect to the server, a Minetest client is needed.

5.17.1 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Minetest:

- UDP 30000

5.17.2 Install and enable mods

After SSHing into your FreedomBox server, install (unzip or `git clone`) mods in `/var/games/minetest-server/.minetes` (for example, for the mobs_animal mod, you'd have the new `/var/games/minetest-server/.minetest/mods/mobs_ani` directory).

To enable a mod, first restart minetest:

```
sudo systemctl restart minetest-server.service
```

This will update the world config file, located in `/var/games/minetest-server/.minetest/worlds/world/world.mt`, with a line related to the added mod. Set that line from `false` to `true` in order to enable the new mod in your minetest instance. For example:

```
load_mod_mobs_animal = true
```

After that, save your changes, restart minetest one more time, then you should be all set.

5.17.3 External links

- Upstream Project: `https://www.minetest.net`
- Upstream Documentation: `https://wiki.minetest.net`
- Debian Minetest Wiki
- Using Minetest in Education: `https://www.minetest.net/education/`

5.17.3.1 Client Apps

- Debian: the client app package name is DebianPkg:minetest.
- Other platforms: https://www.minetest.net/downloads/

5.18 MiniDLNA / ReadyMedia (Simple Media Server)



Available since: version 19.23

MiniDLNA is a media server with the aim to be compliant with DLNA/UPnP clients.

Please know that the maintainers of MiniDLNA software are changing the name of this program to ReadyMedia.

Note: This service is available only on networks configured as "internal" zone. It is not available when connected via OpenVPN.

5.18.1 What is UPnP/DLNA?

Universal plug & play is a set of networking protocols that allow devices within a network such as PCs, TVs, printers etc. to seamlessly discover each other and establish communication for data sharing. It is zero configuration protocol and requires only a media server and a media player that are compliant with the protocol.

DLNA is derived from UPnP as a form of standardizing media interoperability. It forms a standard/certification which many consumer electronics conform to.

5.18.2 Setting up MiniDLNA on your FreedomBox

To install/enable the media server you need to navigate at MiniDLNA page and enable it. The application is intended to be available in the internal (home) network and therefore it requires a network interface configured for internal traffic.

After installation a web page becomes available in the internal network at http://<your-freedombox-IP-address>:8200 (note that using mDNS name like freedombox.local doesn't work here). The web page includes information for how many files the server is detecting, how many connections exist etc. This is very useful if plugging external disks with media to check if the new media files are detected properly. If that is not happening, disabling and enabling the server will fix it.

5.18.3 Using MiniDLNA to play media on your devices

Any DLNA compliant device or media player should be able to automatically detect, browse and play media from MiniDLNA on FreedomBox. The following devices and media players have been tested:

- **GNOME Videos**: Videos is the default media player on the popular GNU/Linux desktop environment GNOME. Open Videos, switch to 'Channels'. You should see a channel named 'freedombox: minidlna'. You will be able to browse and play media from it.
- **VLC media player**: VLC is a very popular media player for GNU/Linux, Android, Windows and macOS. Open VLC and click on 'View -> Playlist'. In the playlist sidebar that appears, select 'Universal Plug'n'Play'. You should see an item named 'freedombox: minidlna'. You should be able to browse and play media from it.
- **Kodi**: Kodi is a popular media centre software with user interface designed for Televisions. Open Kodi, goto 'System -> Service settings -> UPnP/DLNA' and 'Enable UPnP support'. Then visit 'Home -> Videos -> Files -> Add videos... -> Browse -> UPnP devices'. You should see 'freedombox: minidlna'. Select it and choose 'OK'. Then choose 'OK in the 'Add video source' dialog. From now on, you should see 'freedombox: minidlna' in 'Videos -> Files' section. You should be able to browse and play media from it. See Kodi documentation for more information.
- **Roku**: Roku is an appliance connected to a TV for playing Internet streaming services. Many TVs also have Roku built into them. In Roku interface, find a channel called 'Roku Media Player' and open it. You should see an item called 'freedombox: minidlna'. You should be able to browse and play media from it.
- **Rhythmbox**: Rhythmbox is the default audio player on the popular GNU/Linux desktop environment GNOME. Open Rhythmbox and ensure that the side pane is open by clicking on 'Application menu -> View -> Side Pane'. In the side pane you should see 'freedombox:minidlna' under the 'Shared' section. You should be able to browse and play audio files from it. Video files will not show up.

5.18.4 Supported media formats

MiniDLNA supports a wide variety of video and audio file formats.

- **Video**: Files ending with .avi, .mp4, .mkv, .mpg, .mpeg, .wmv, .m4v, .flv, .mov, .3gp, etc.
- **Audio**: Files ending with .mp3, .ogg, .flac, .wav, .pcm, .wma, .fla, .aac, etc.
- **Image**: Files ending with .jpg, .jpeg
- **Playlist**: Files ending with .m3u, .pls
- **Captions**: Files ending with .srt, .smi

Notably, it does **not** support the following file extensions. Renaming the file to a known extension seems to work in most cases.

- **Video**: Files ending with .webm

In addition to file format support from MiniDLNA, your media player or device needs to support the audio/video codecs with which the media has been encoded. MiniDLNA does not have the ability to translate files into a codec understood by the player. If you face problems with media playback, use the VLC player to find the codecs used in the media and the check your device or media player documentation on whether the codecs are supported.

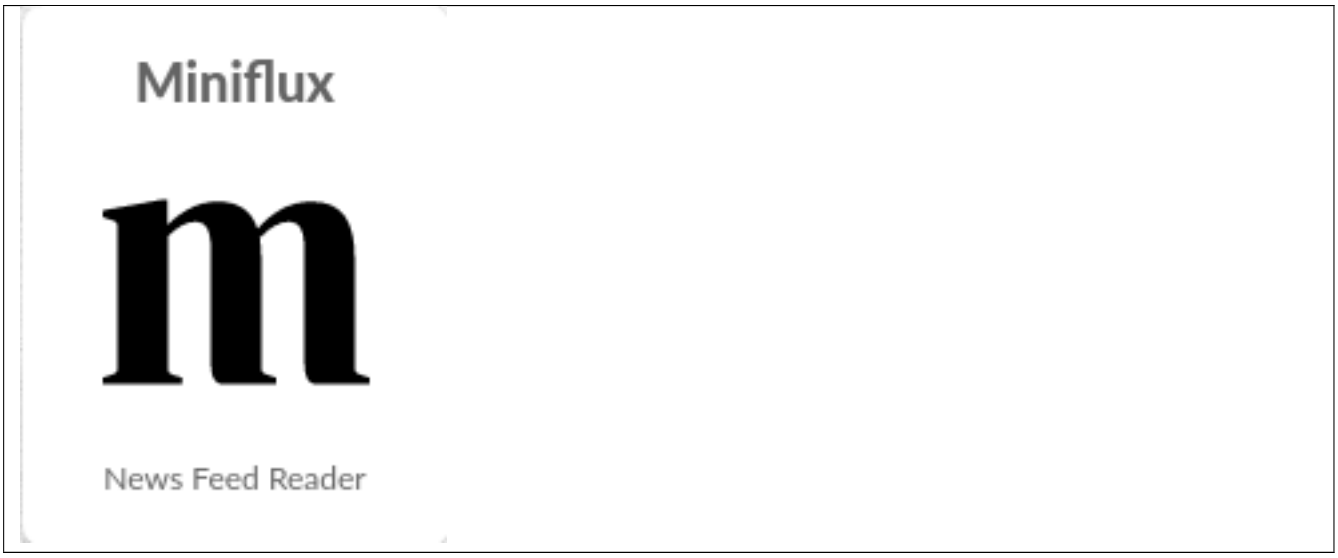
5.18.5 File systems for external drives

If using an external drive that is used also from a Windows system the preferred filesystem should be NTFS. NTFS will keep Linux file permissions and UTF8 encoding for file names. This is useful if file names are in your language.

5.18.6 External links

- Upstream project site: http://minidlna.sourceforge.net
- Upstream documentation and support: https://sourceforge.net/projects/minidlna/support
- Debian MiniDLNA wiki: https://wiki.debian.org/minidlna
- About DLNA: https://en.wikipedia.org/wiki/Digital_Living_Network_Alliance

5.19 Miniflux (News Feed Reader)



Available since: FreedomBox on Debian 13 (Trixie)

Miniflux is a minimalist and opinionated news feed reader supporting a wide variety of feed formats - RSS, ATOM, RDF and JSON. It is an alternative to Tiny Tiny RSS on FreedomBox.

5.19.1 Clients

The primary web interface is mobile-friendly and optimized for readability. Miniflux is a progressive web application. It can be added to the home screen on mobile and accessed like a native application.

Miniflux server provides an API for clients. It is also compatible with existing applications built for Fever or Google Reader API. There is a listing of clients for various desktop and mobile platforms on the official website.

5.19.2 Reader

Miniflux's reader supports several content types. In addition to the usual blog posts and comics which most feed readers support, you can also listen to podcasts and play videos from within Miniflux. YouTube videos are played using youtube-nocookie.com or (optional) a proxy like invidious.io to prevent tracking your preferences across websites. Miniflux also removes tracking pixels and all external JavaScript.

Miniflux has a content scraper that allows you to fetch and read the full article even if the site only publishes a summary in their feed. Only the relevant content is scraped, providing a reading experience similar to Firefox Reader View. This scraper is also capable of fetching the video duration from YouTube or Invidious.

5.19.3 Other Features

Miniflux provides some quality-of-life features like full-text search (powered by PostgreSQL database), filtering rules, keyboard shortcuts and touch events. There are integrations with several applications like read-it-later solutions, bookmarking tools (Shaarli) and messaging apps (Matrix). If you don't use Shaarli, starring an article within Miniflux is equivalent to bookmarking it.

Miniflux doesn't permit installing additional plugins or themes. You can pick one of the existing light/dark themes or install a browser extension to read in a different theme.

5.19.4 Miniflux on FreedomBox

5.19.4.1 User Management

Miniflux does not integrate with the existing single sign-on mechanisms in FreedomBox. The FreedomBox app for Miniflux provides an option to create administrator users. Once an administrator user is created, any further users (administrator or not) can be created and managed from within Miniflux settings.

The FreedomBox app also provides an option to reset any Miniflux user's password. This is especially convenient in the case where there is only one person using Miniflux. Without this option, in case the only user of Miniflux forgets their password, another administrator user must be created to reset this user's password from Miniflux settings.

Since Miniflux accounts are not tied to FreedomBox accounts, you can provide Miniflux accounts to your friends who do not have accounts on your FreedomBox.

5.19.4.2 Migrating from Tiny Tiny RSS

TODO

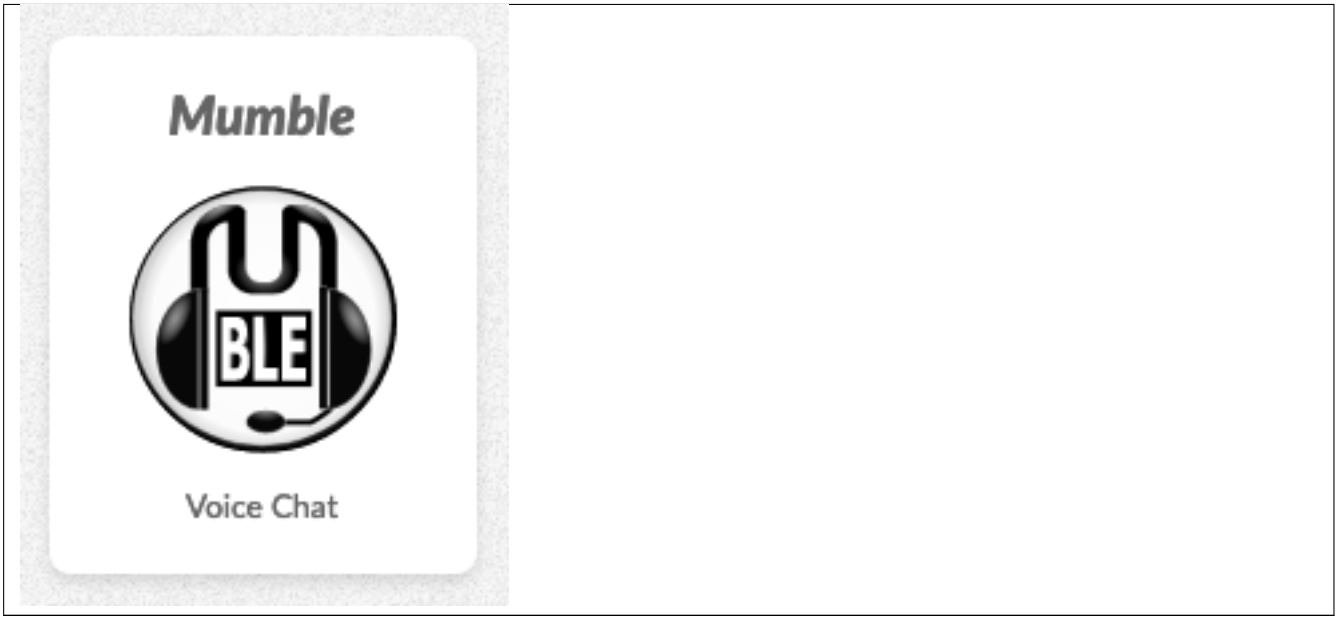
5.19.4.3 Integrations

1. Adding a bookmark to Shaarli
2. Posting feeds to a Matrix channel
3. RSS-Bridge

5.19.5 External links

- Upstream project: https://miniflux.app
- Upstream documentation: https://miniflux.app/docs/index.html

5.20 Mumble (Voice Chat) Server



Available since: version 0.5

5.20.1 What is Mumble?

Mumble is a voice chat software. Primarily intended for use while gaming, it is suitable for simple talking with high audio quality, noise suppression, encrypted communication, public/private-key authentication by default, and "wizards" to configure your microphone for instance. A user can be marked as a "priority speaker" within a channel.

5.20.2 Using Mumble

FreedomBox includes the Mumble server. Clients are available for desktop and mobile platforms. Users can download one of these clients and connect to the server.

5.20.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Mumble:

- TCP 64738
- UDP 64738

5.20.4 Managing Permissions

A super user in Mumble has the ability to create administrator accounts who can in turn manage groups and channel permissions. This can be done after logging in with the username "SuperUser" using the super user password. See Mumble Guide for information on how to do this. The SuperUser password can be set through the FreedomBox interface.

5.20.5 External links

- Upstream Project: `https://www.mumble.info`
- User documentation: `https://www.mumble.info/documentation`

5.21 Nextcloud (File Storage & Collaboration)



Available since: FreedomBox 24.8

5.21.1 What is Nextcloud?

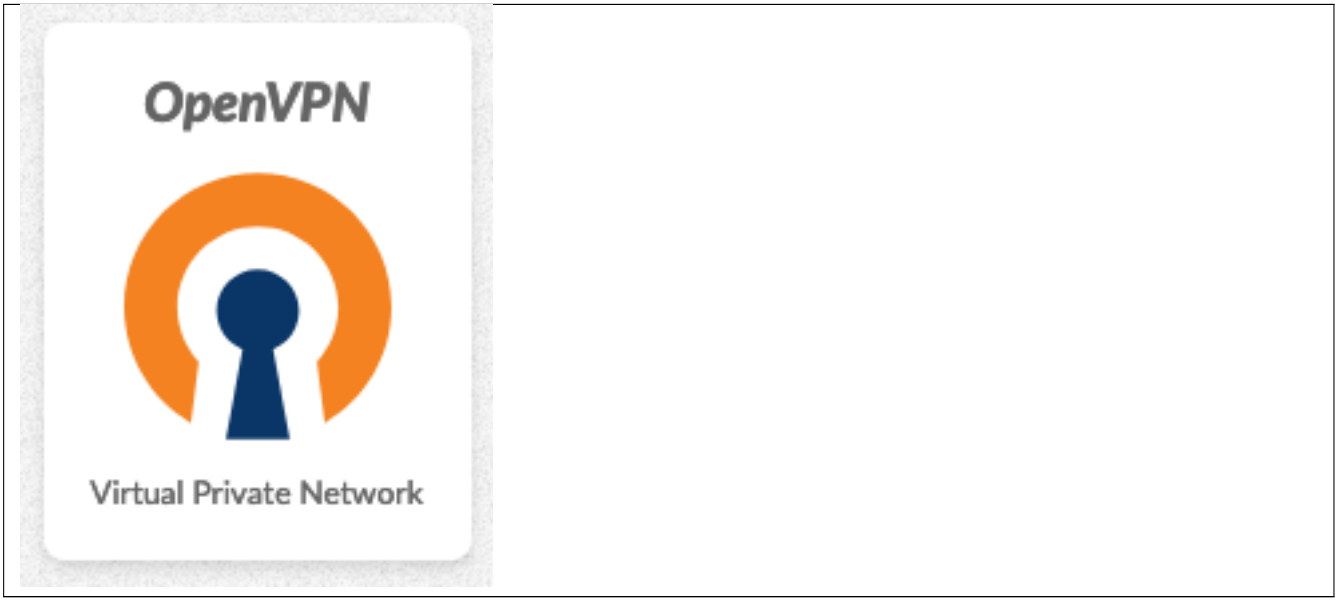
This page is new and needs a Nexcloud user to help with the content. This could be you!

Nextcloud is a self-hosted productivity platform which provides private and secure functions for file sharing, collaborative work, and more. Nextcloud includes the Nextcloud server, client applications for desktop computers, and mobile clients. The Nextcloud server provides a well integrated web interface.

All users of FreedomBox can use Nextcloud. To perform administrative actions, use the "nextcloud-admin" user after setting a password here.

Please note that Nextcloud is installed and run inside a container provided by the Nextcloud community. Security, quality, privacy and legal reviews are done by the upstream project and not by Debian/FreedomBox. Updates are performed following an independent cycle.

5.22 OpenVPN (Virtual Private Network)



Available since: version 0.7

5.22.1 What is OpenVPN?

OpenVPN provides to your FreedomBox a virtual private network service. You can use this software for remote access, site-to-site VPNs and Wi-Fi security. OpenVPN includes support for dynamic IP addresses and NAT.

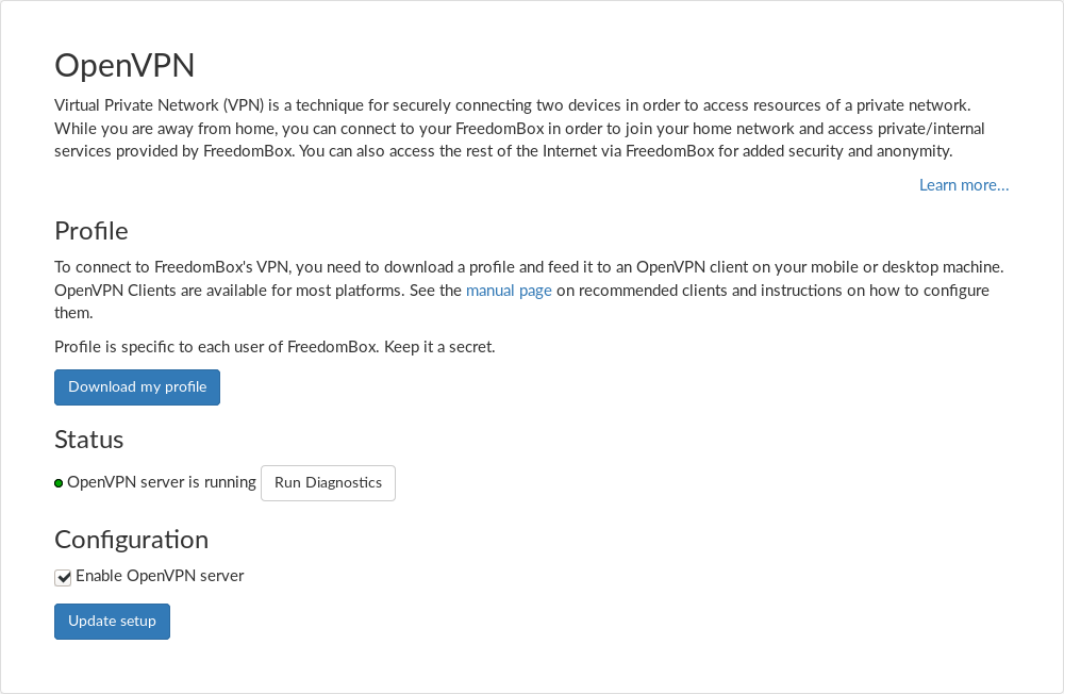
5.22.2 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for OpenVPN:

- UDP 1194

5.22.3 Setting up

1. In FreedomBox apps menu, select *Virtual Private Network (OpenVPN)* and click Install.
2. After the module is installed, there is an additional setup step that may take a long time to complete. Click "Start setup" to begin.



3. Wait for the setup to finish. This could take a while.
4. Once the setup of the OpenVPN server is complete, you can download your profile. This will download a file called <USER>.ovpn, where <USER> is the name of a FreedomBox user. Each FreedomBox user will be able to download a different profile. Users who are not administrators can download the profile from home page after login.
5. The ovpn file contains all the information a vpn client needs to connect to the server.
6. The downloaded profile contains the domain name of the FreedomBox that the client should connect to. This is picked up from the domain configured in 'Config' section of 'System' page. In case your domain is not configured properly, you may need to change this value after downloading the profile. If your OpenVPN client allows it, you can do this after importing the OpenVPN profile. Otherwise, you can edit the .ovpn profile file in a text editor and change the 'remote' line to contain the WAN IP address or hostname of your FreedomBox as follows.

```
client
remote mybox.freedombox.rocks 1194
proto udp
```

5.22.4 Troubleshooting

If your network doesn't support IPv6, you might have to remove the following line from your OpenVPN client configuration. This is especially in cases where your server supports IPv6 but client does not thus confusing the OpenVPN client on which protocol to use.

```
proto udp6
```

To connect via IPv4, ensure that the following line is present.

```
proto udp
```

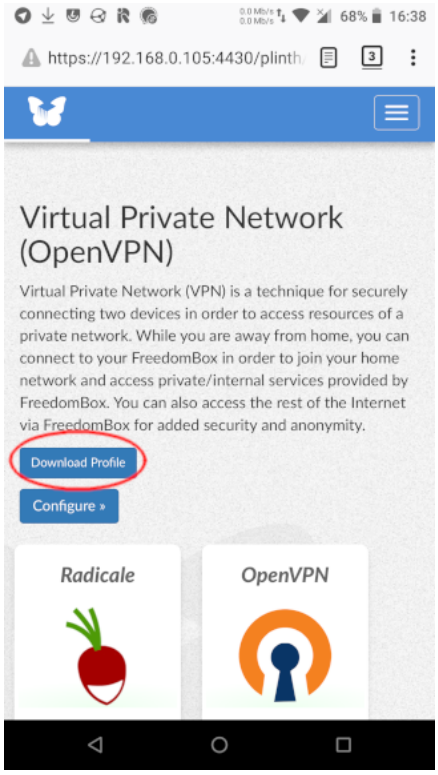
5.22.5 Browsing Internet after connecting to VPN

After connecting to the VPN, the client device will be able to browse the Internet without any further configuration. However, a pre-condition for this to work is that you need to have at least one Internet connected network interface which is part of the 'External' firewall zone. Use the networks configuration page to edit the firewall zone for the device's network interfaces.

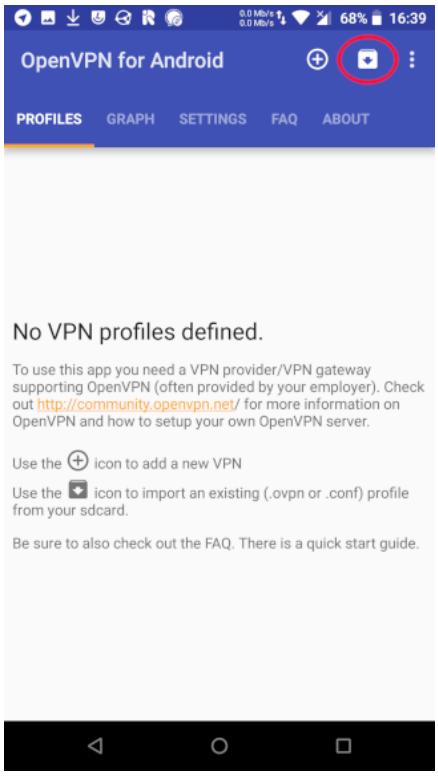
5.22.6 Usage

5.22.6.1 On Android/LineageOS

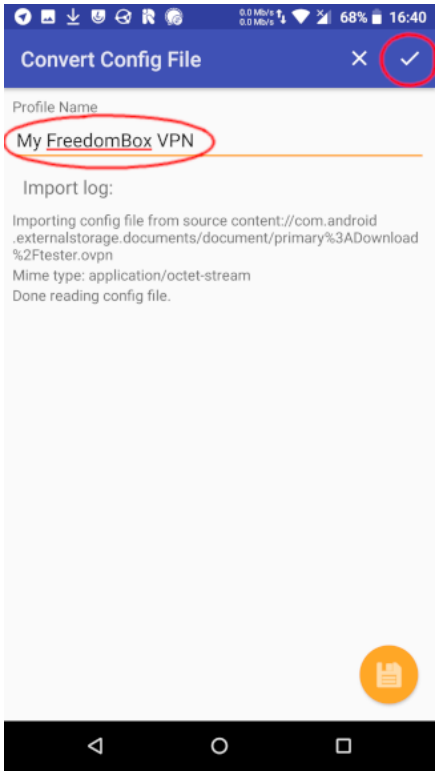
1. Visit FreedomBox home page. Login with your user account. From home page, download the OpenVPN profile. The file will be named *username*.ovpn.



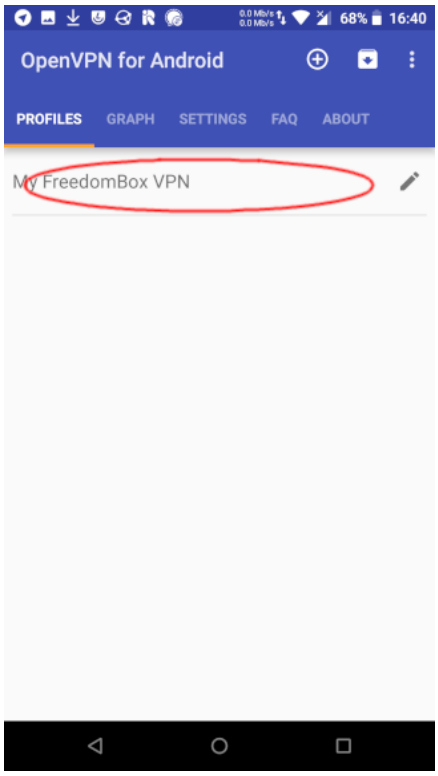
2. Download an OpenVPN client such as *OpenVPN for Android*. F-Droid repository is recommended. In the app, select import profile.

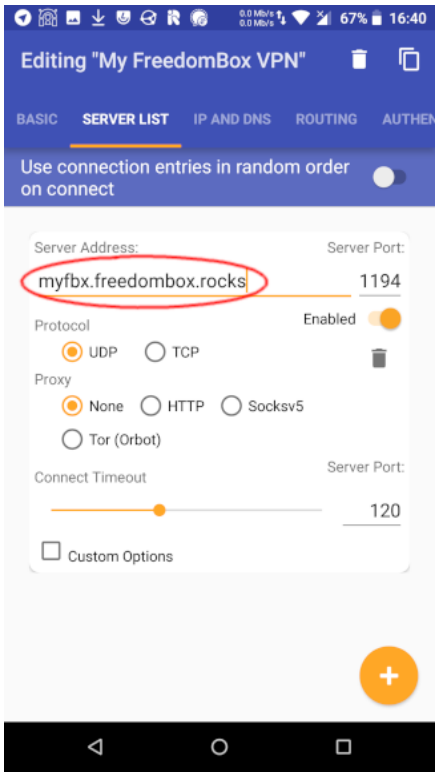


3. In the select profile dialog, choose the *username*.opvn file you have just downloaded. Provide a name for the connection and save the profile.

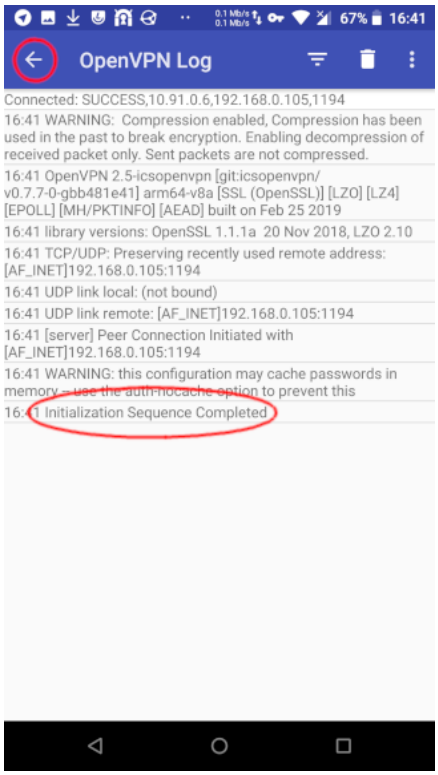


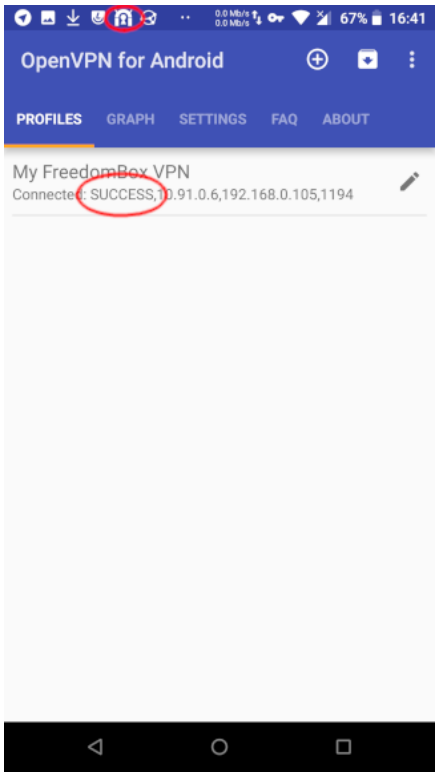
4. Newly created profile will show up. If necessary, edit the profile and set the domain name of your FreedomBox as the server address.

- 

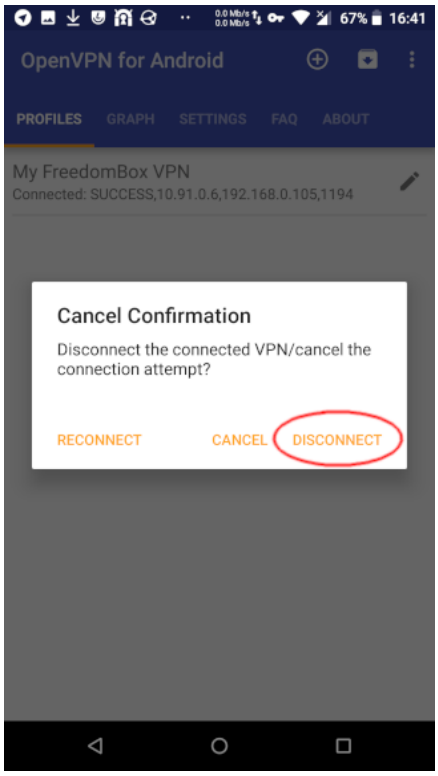


5. Connect by tapping on the profile.

- 

- 

6. When done, disconnect by tapping on the profile.

- 

5.22.6.2 On Debian

Install an OpenVPN client for your system

```
$ sudo apt install openvpn
```

Open the ovpn file with the OpenVPN client.

```
$ sudo openvpn --config /path/to/<USER>.ovpn
```

If you use Network Manager, you can create a new connection by importing the file:

```
$ sudo apt install network-manager-openvpn-gnome
$ sudo nmcli connection import type openvpn file /path/to/<USER>.ovpn
```

If you get an error such as `configuration error: invalid 1th argument to "proto" (line 5)` then edit the .ovpn file and remove the line `proto udp6`.

5.22.7 Checking if you are connected

5.22.7.1 On Debian

1. Try to ping the FreedomBox or other devices on the local network.
2. Running the command `ip addr` should show a `tun0` connection.
3. The command `traceroute freedombox.org` should show you the ip address of the VPN server as the first hop.

5.22.8 Accessing internal services

After connecting to OpenVPN, you will be able to access FreedomBox services that are only meant to be accessed on internal networks. This is in addition to being able to access external services. This can be done by using the IP address 10.91.0.1 as the host name for these services (for example, use smb://10.91.0.1 instead of smb://freedombox.local to access Samba shares).

The following services are known to **work**:

- Privoxy,
- Tor Socks,
- Shadowsocks,
- Samba.

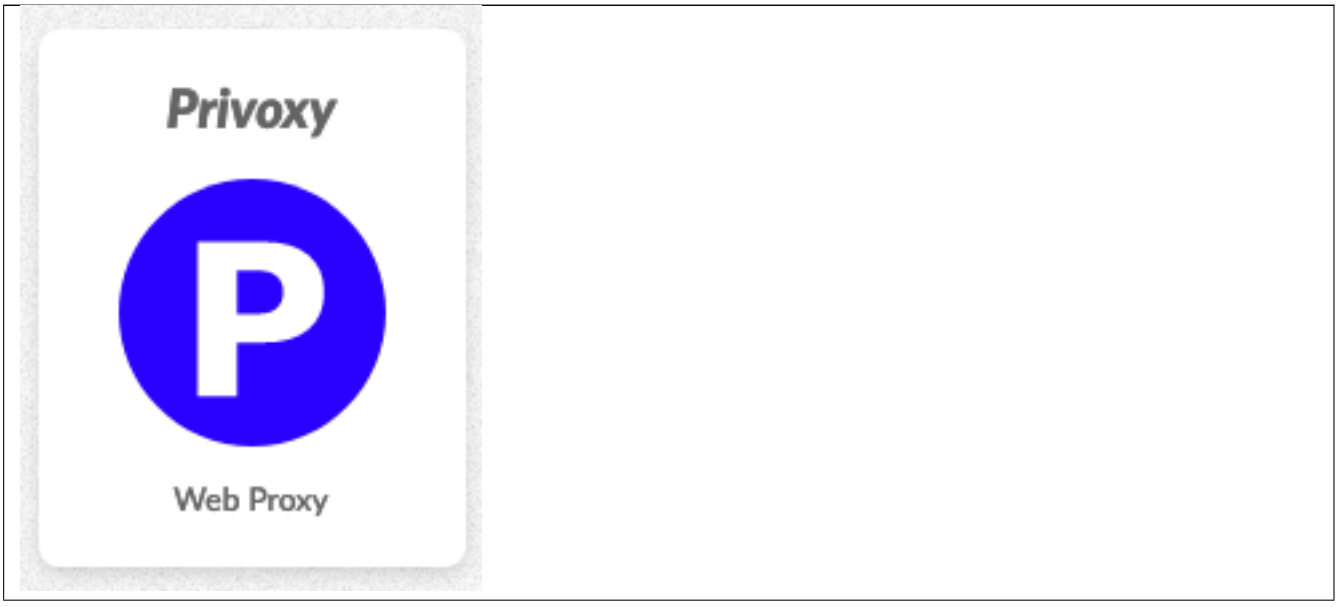
Some services are known **not** to work at this time:

- Avahi,
- bind and
- MiniDLNA.

5.22.9 External Links

- Upstream Project: https://openvpn.net/
- Upstream documentation: https://openvpn.net/connect-docs/
- Debian OpenVPN wiki: https://wiki.debian.org/OpenVPN
- Wiki / Tracker: https://community.openvpn.net/openvpn

5.23 Privoxy (Web Proxy)



Available since: version 0.1

A web proxy acts as a filter for incoming and outgoing web traffic. Thus, you can instruct any computer in your network to pass internet traffic through the proxy to remove unwanted ads and tracking mechanisms.

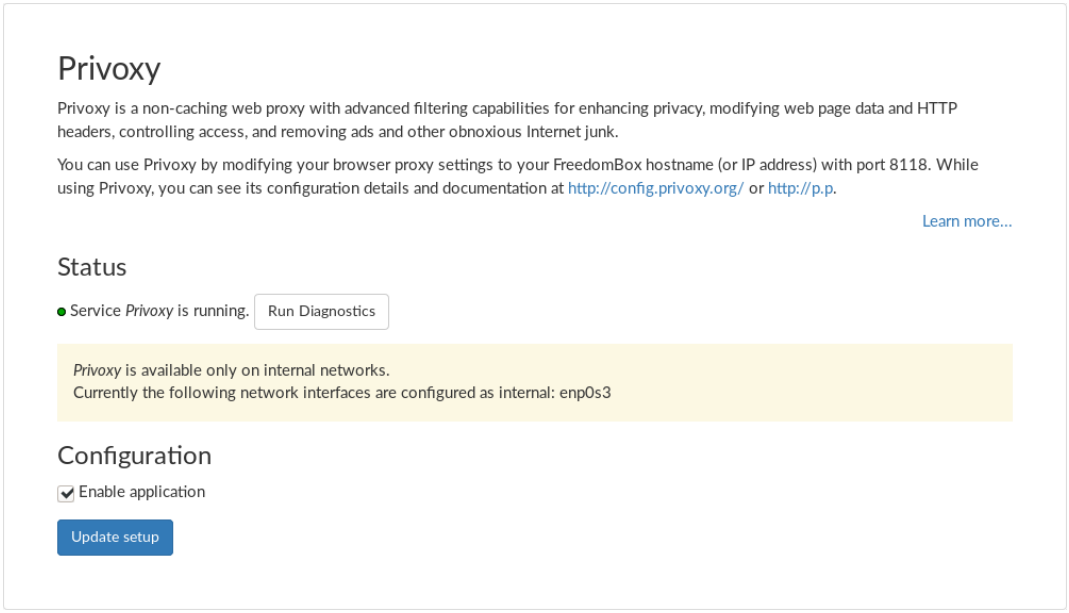
Privoxy is a software for security, privacy, and accurate control over the web. It provides a much more powerful web proxy (and anonymity on the web) than what your browser can offer. Privoxy "is a proxy that is primarily focused on privacy enhancement, ad and junk elimination and freeing the user from restrictions placed on his activities" (source: Privoxy FAQ).

5.23.1 Screencast

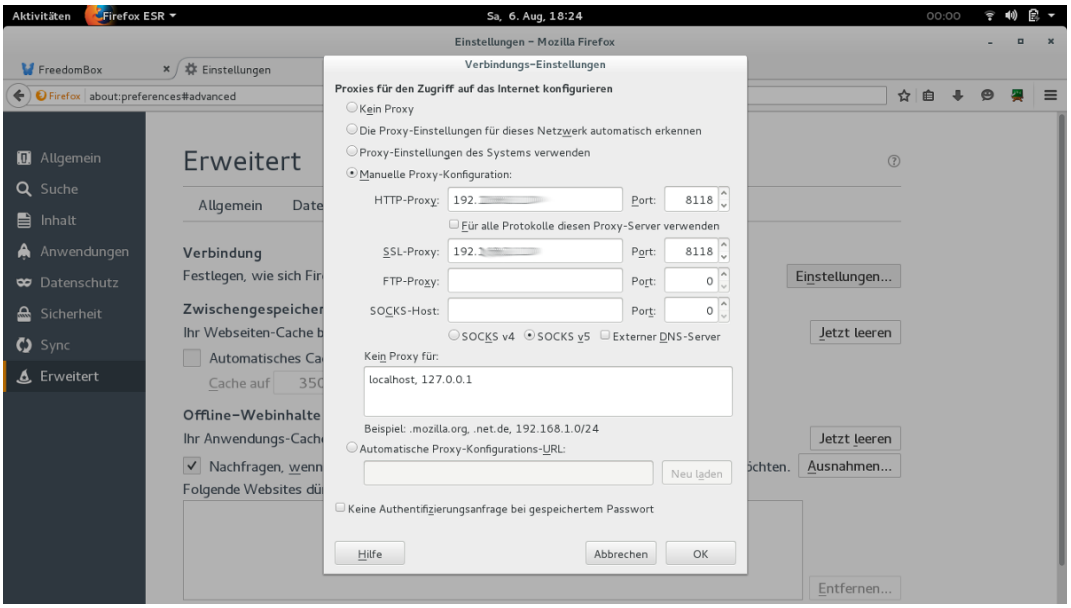
Watch the screencast on how to setup and use Privoxy in FreedomBox.

5.23.2 Setting up

1. In FreedomBox, install *Web Proxy (Privoxy)*



2. Adapt your browser proxy settings to your FreedomBox hostname (or IP address) with port 8118. Please note that Privoxy can only proxy HTTP and HTTPS traffic. It will not work with FTP or other protocols.



3. Go to page `http://config.privoxy.org/` or `http://p.p`. If Privoxy is installed properly, you will be able to configure it in detail; if not you will see an error message.

4. If you are using a laptop that occasionally has to connect through other routers than yours with the FreedomBox and Privoxy, you may want to install a proxy switch add-on that allows you to easily turn the proxy on or off.

5.23.3 Advanced Users

The default installation should provide a reasonable starting point for most. There will undoubtedly be occasions where you will want to adjust the configuration, that can be dealt with as the need arises.

1. Plan first:
   - While using Privoxy, you can see its configuration details and documentation at `http://config.privoxy.org/` or `http://p.p`.
   - The Quickstart is a good starting point to read on how to define own blocking and filtering rules.
   - Read carefully the manual, especially this **security warning**: *Access to the editor can not be controlled separately by "ACLs" or HTTP authentication, so that everybody who can access Privoxy can modify its configuration for all users. This option is not recommended for environments with untrusted users. Note that malicious client side code (e.g Java) is also capable of using the actions editor and you shouldn't enable this options unless you understand the consequences and are sure your browser is configured correctly.*

2. Only when you are ready, perform the changes:
   1. To enable changing these configurations, you first have to change the value of `enable-edit-actions` in `/etc/privox` to `1`.
   2. Now you find an `EDIT` button on the configuration screen in `http://config.privoxy.org/`.

5.23.4 External links

- Upstream project: `https://www.privoxy.org`
- Upstream documentation: `https://www.privoxy.org/user-manual/index.html`

5.24 Quassel (Text Chat Client via IRC)



Available since: version 0.8

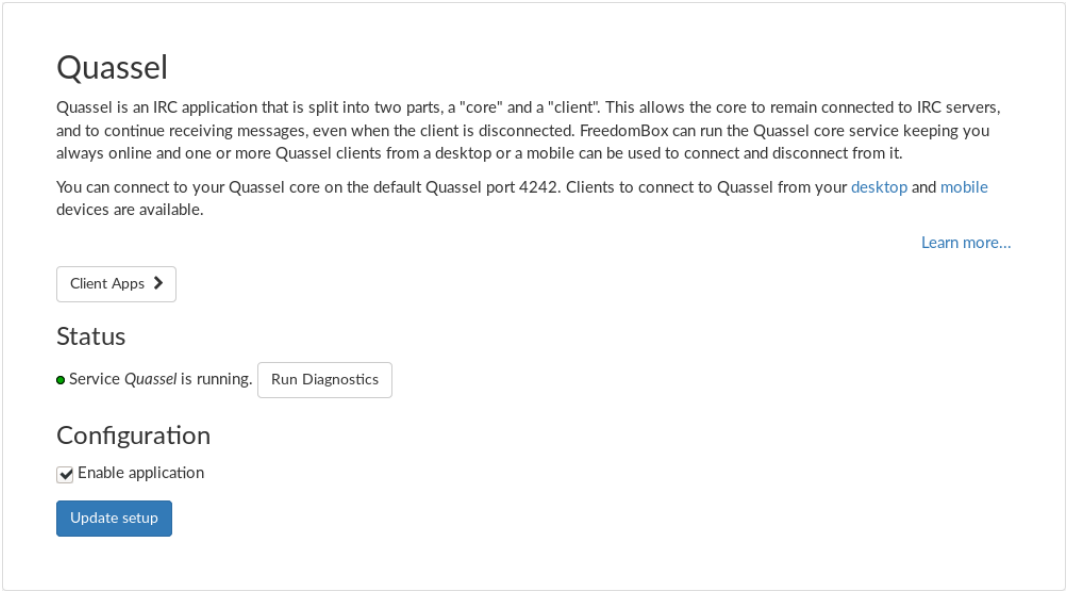
Quassel is an IRC application that is split into two parts, a "core" and a "client". This allows the core to remain connected to IRC servers, and to continue receiving messages, even when the client is disconnected. FreedomBox can run the Quassel core service keeping you always online and one or more Quassel clients from a desktop or a mobile device can be used to connect and disconnect from it.

5.24.1 Why run Quassel?

Many discussions about FreedomBox are being done on the IRC-Channel `irc://irc.debian.org/freedombox`. If your FreedomBox is running *Quassel*, it will collect all discussions while you are away, such as responses to your questions. Remember, the FreedomBox project is a worldwide project with people from nearly every time zone. You use your *client* to connect to the *Quassel* core to read and respond whenever you have time and are available.

5.24.2 How to setup Quassel?

- Within FreedomBox's web interface
  1. select *Applications*
  2. go to *IRC Client (Quassel)* and
  3. install the application and make sure it is enabled

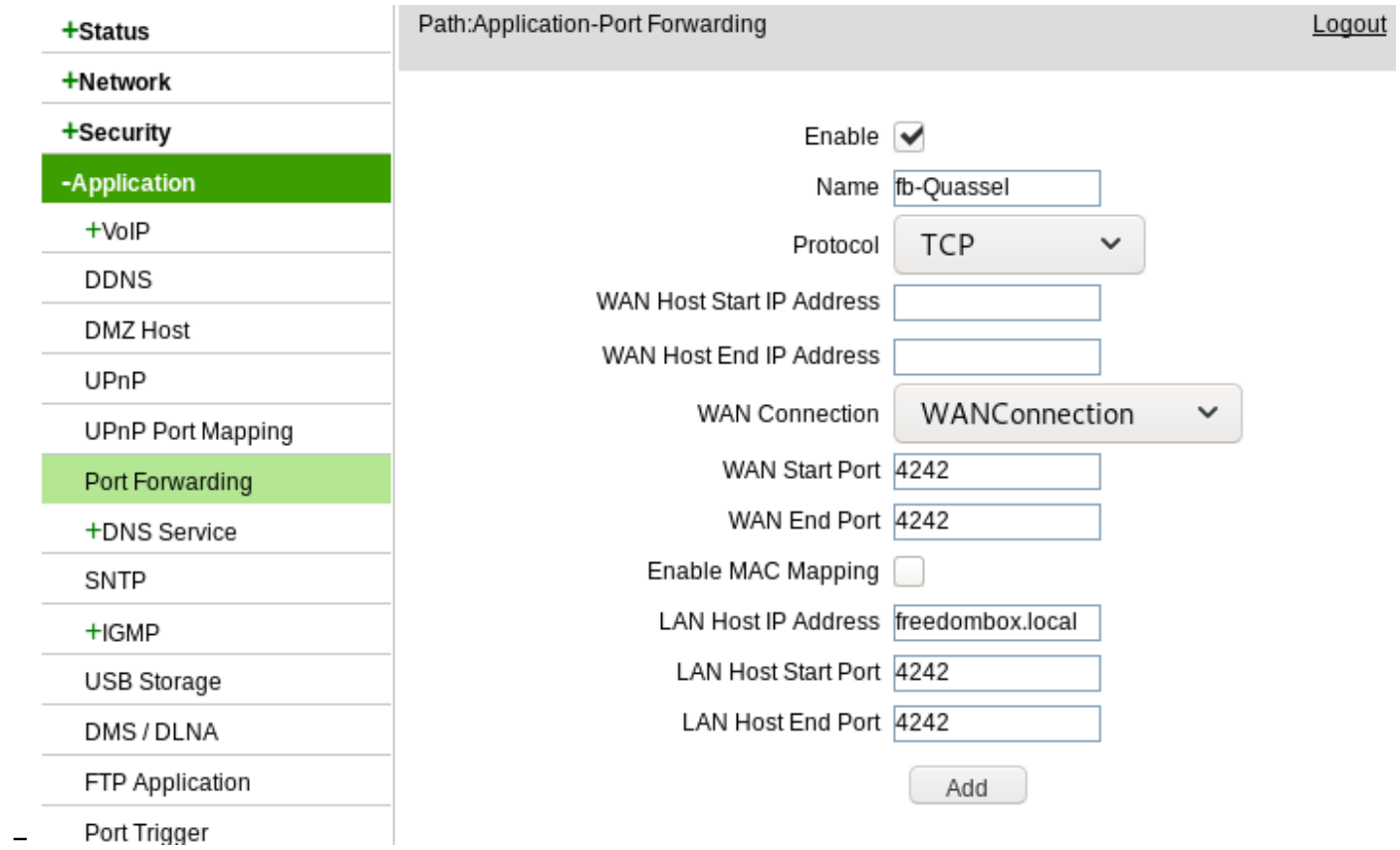


4. now your Quassel core is running

5.24.3 Port Forwarding

If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Quassel:

- TCP 4242
- Example configuration in router:



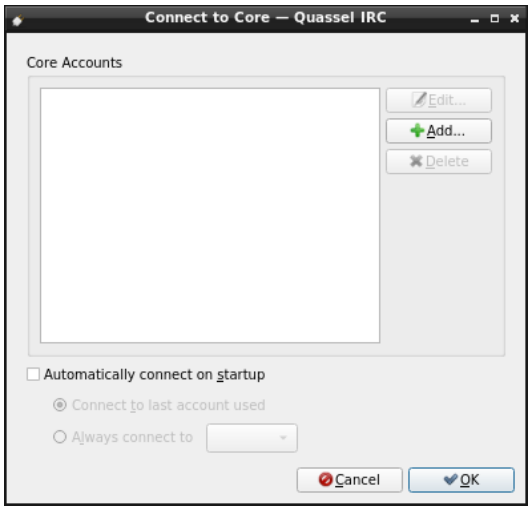
5.24.4 Clients

Clients to connect to Quassel from your desktop and mobile devices are available.

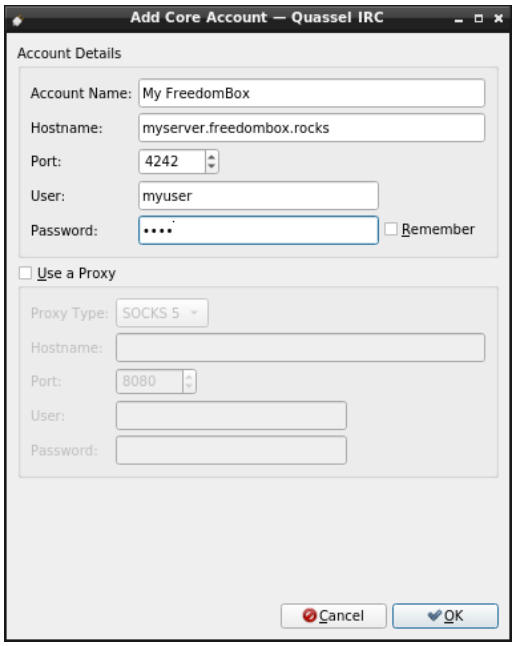
5.24.4.1 Desktop

In a Debian system, you can e.g. use quassel-client. The following steps describe how to connect Quassel Client with Quassel Core running on a FreedomBox. The first time you do this connection, Quassel Core will be initialized too.

1. Launch Quassel Client. You will be greeted with a wizard to `Connect to Core`.

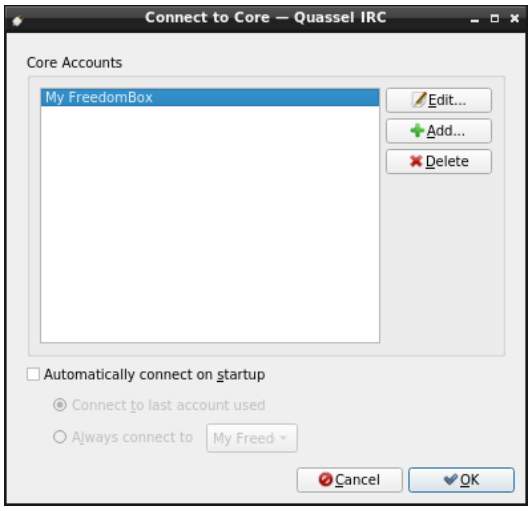


2. Click the `Add` button to launch `Add Core Account` dialog.

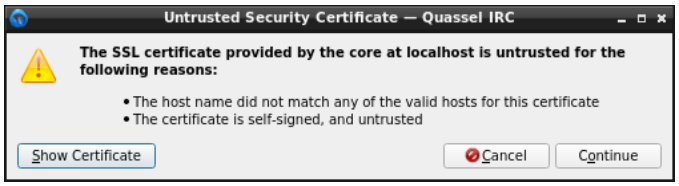


3. Fill any value in the `Account Name` field. Fill proper DNS hostname of your FreedomBox in `Hostname` filed. `Port` field must have the value `4242`. Provide the username and password of the account you wish to create to connect to the Quassel Core in the `User` and `Password` fields. Choose `Remember` if don't wish to be prompted for a password every time you launch Quassel client.

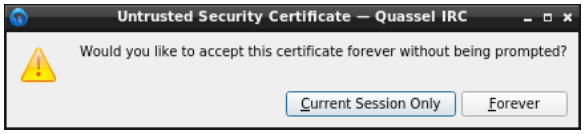
4. After pressing `OK` in the `Add Core Account` dialog, you should see the core account in the `Connect to Core` dialog.



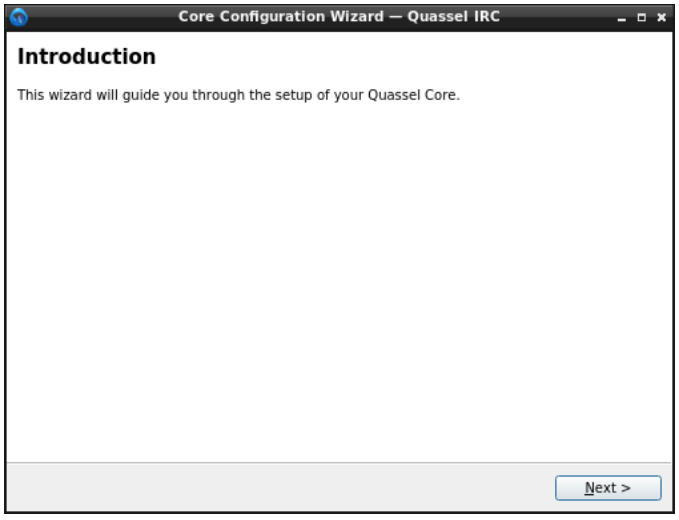
5. Select the newly created core account and select `OK` to connect to it.
6. If this is the first time you are connecting to this core. You will see an `Untrusted Security Certificate` warning and need to accept the server certificate.



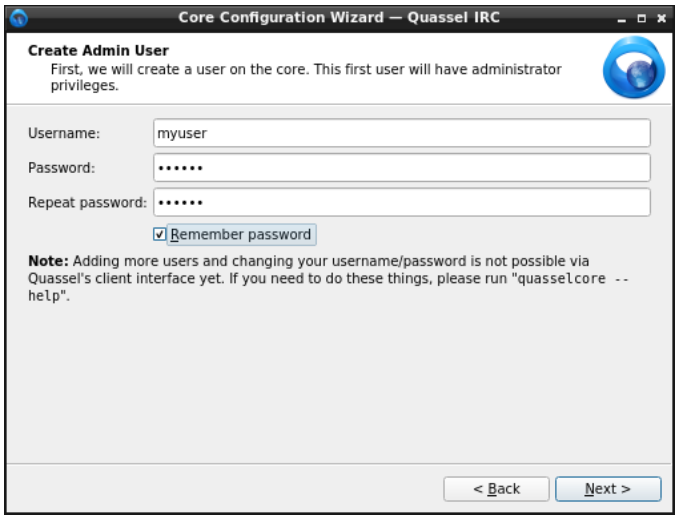
7. Select `Continue`. Then you will be asked if you wish to accept the certificate permanently. Select `Forever`.



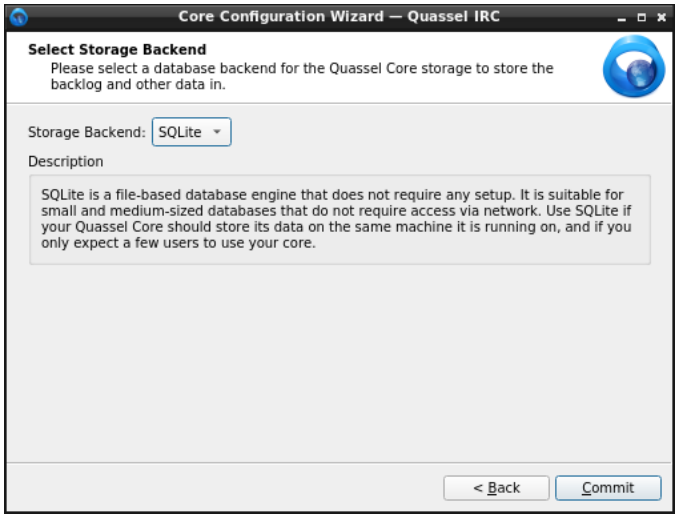
8. If this Quassel Core has not been connected to before, you will then see a `Core Configuration Wizard`. Select `Next`.



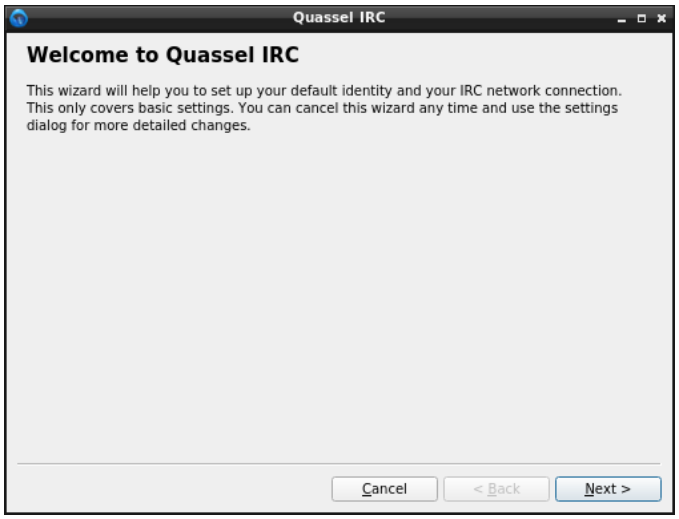
9. In the `Create Admin User` page, enter the username and password you have used earlier to create the core connection. Select `Remember password` to remember this password for future sessions. Click `Next`.



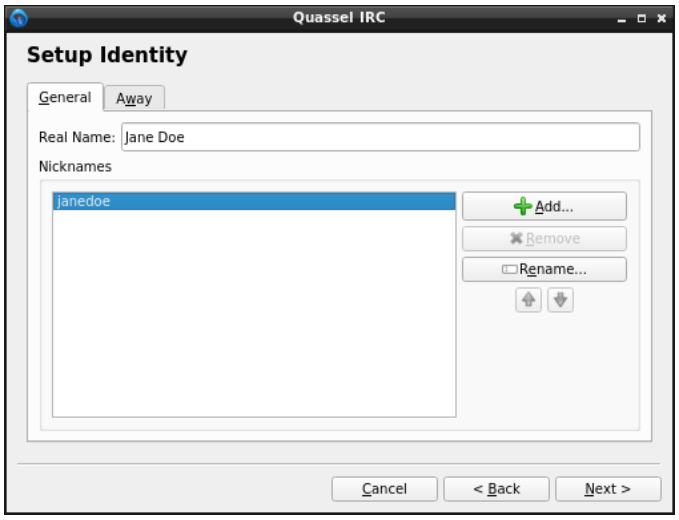
10. In the `Select Storage Backend` page, select `SQLite` and click `Commit`.



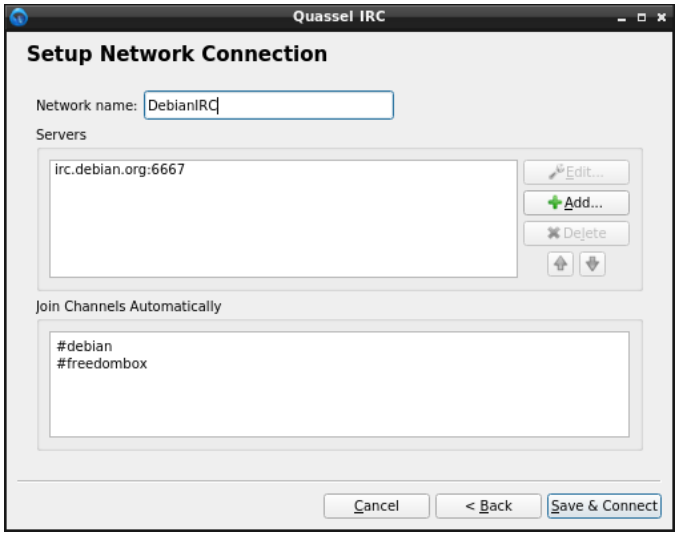
11. The core configuration is then complete and you will see a `Quassel IRC` wizard to configure your IRC connections. Click `Next`.



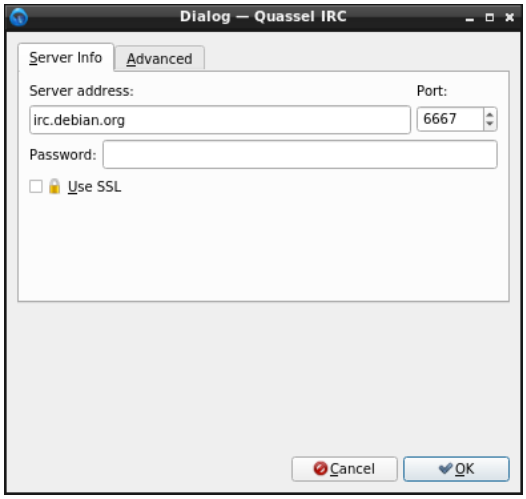
12. In `Setup Identity` page next, provide a name and multiple nicknames. This is how you present yourself to other users on IRC. It is not necessary to give your real world name. Multiple nicknames are useful as fallback nicknames when the first nickname can't be used for some reason. After providing the information click `Next`.



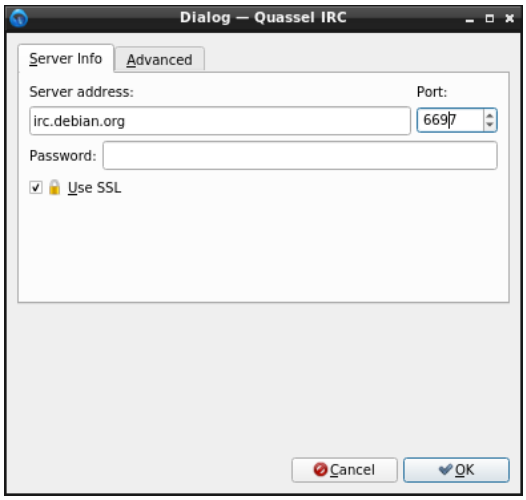
13. In `Setup Network Connection` page next, provide a network name of your choice. Next provide a list of servers to which Quassel Core should connect to in order to join this IRC network (such as irc.debian.org:6667).



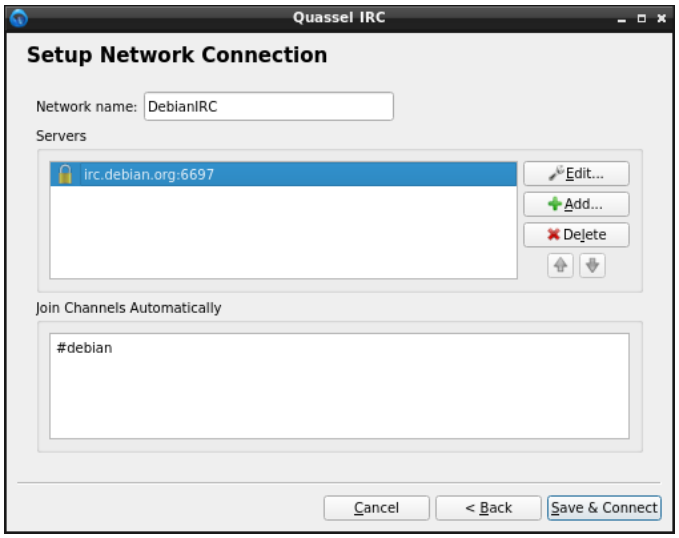
14. Select the server in the servers list and click `Edit`. In the `Server Info` dialog, set the port `6697` (consult your network's documentation for actual list of servers and their secure ports) and click `Use SSL`. Click `OK`. This is to ensure that communication between your FreedomBox and the IRC network server is encrypted.



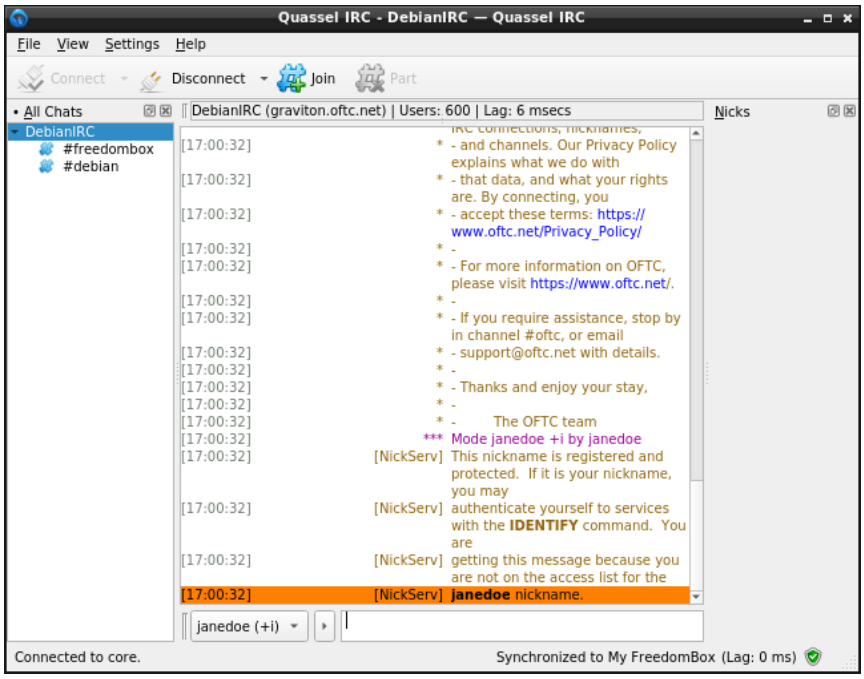
<
>



15. Back in the `Setup Network Connection` dialog, provide a list of IRC channels (such as #freedombox) to join upon connecting to the network. Click `Save & Connect`.



16. You should connect to the network and see the list of channels you have joined on the `All Chats` pane on the left of the Quassel Client main window.

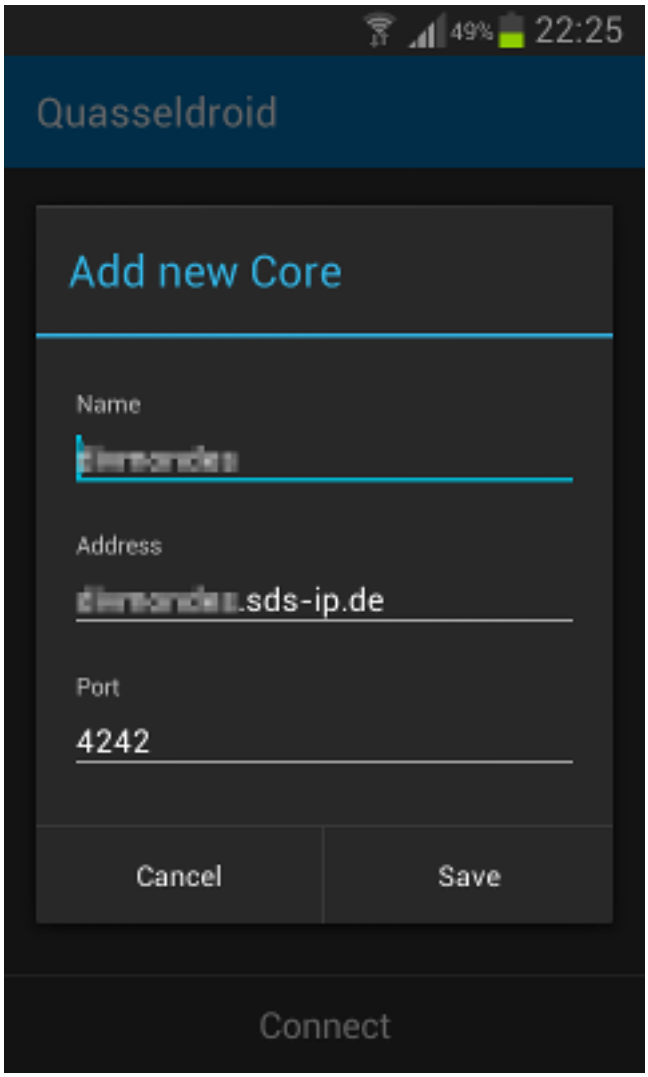


17. Select a channel and start seeing messages from others in the channel and send your own messages.

5.24.4.2 Android

For Android devices you may use e.g. *Quasseldroid* from F-Droid

- enter core, username etc. as above

– 

By the way, the German verb *quasseln* means *talking a lot*, to *jabber*.

5.24.5 External links

- Upstream Project: https://quassel-irc.org
- Upstream documentation: https://bugs.quassel-irc.org/projects/quassel-irc/wiki

5.25 Radicale (Calendar and Addressbook)



Available since: version 0.9

With Radicale, you can synchronize your personal calendars, ToDo lists, and addressbooks with your various computers, tablets, and smartphones, and share them with friends, without letting third parties know your personal schedule or contacts.

5.25.1 Why should I run Radicale?

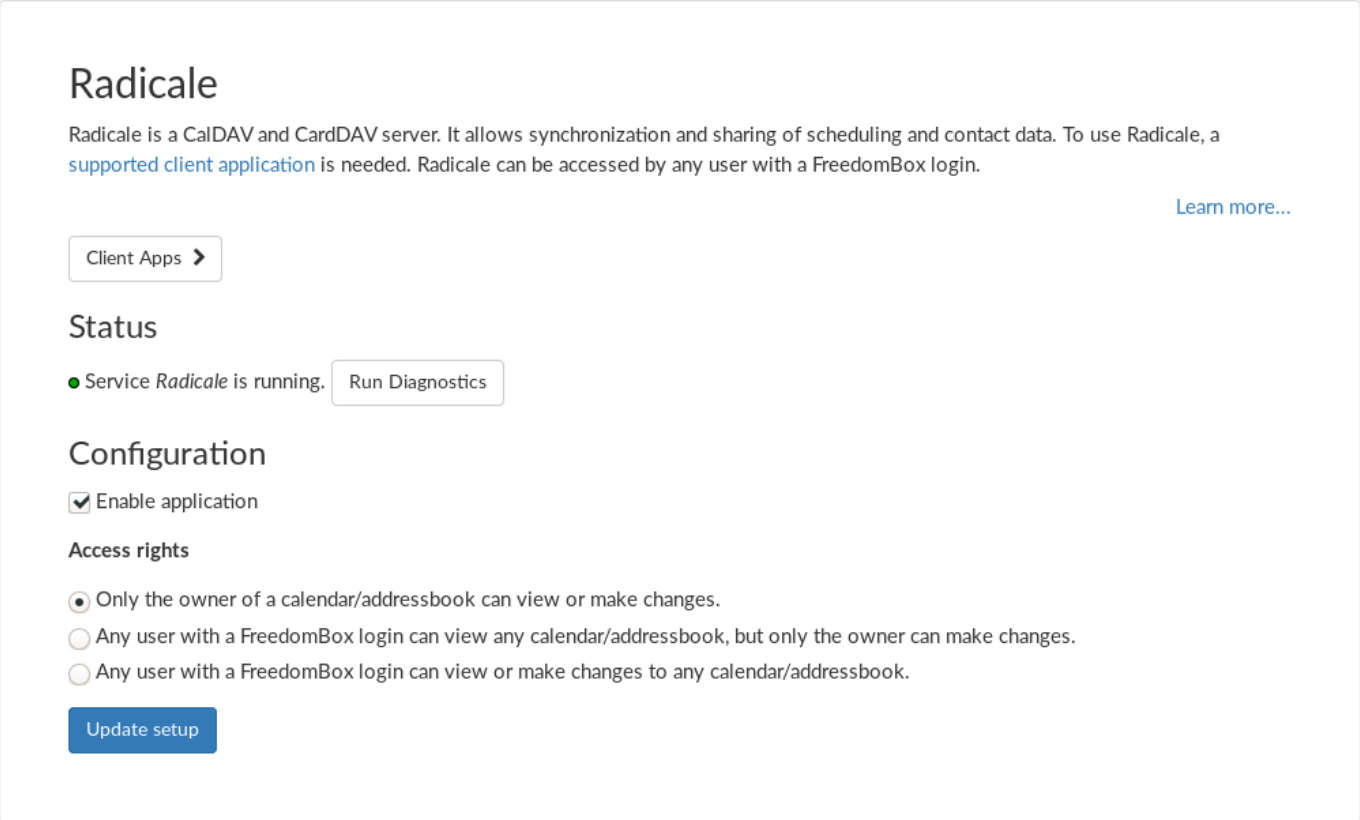
Using Radicale, you can get rid of centralized services like Google Calendar or Apple Calendar (iCloud) data mining your events and social connections.

5.25.2 How to setup Radicale?

First, the Radicale server needs to be activated on your box.

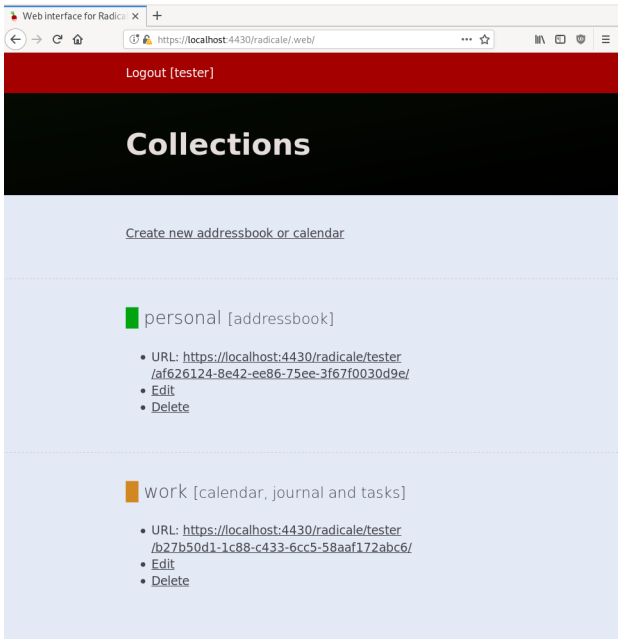
- Within FreedomBox Service:
  1. select *Apps*
  2. go to *Radicale (Calendar and Addressbook)* and
  3. install the application. After the installation is complete, make sure the application is marked "enabled" in the FreedomBox interface. Enabling the application launches the Radicale CalDAV/CardDAV server.
  4. define the access rights:
     - Only the owner of a calendar/addressbook can view or make changes
     - Any user can view any calendar/addressbook, but only the owner can make changes
     - Any user can view or make changes to any calendar/addressbook

Note, that only users with a FreedomBox login can access Radicale.



If you want to share a calendar with only some users, the simplest approach is to create an additional user-name for these users and to share that user-name and password with them.

Radicale provides a basic web interface, which only supports creating new calendars and addressbooks. To add events or contacts, an external supported client application is needed.

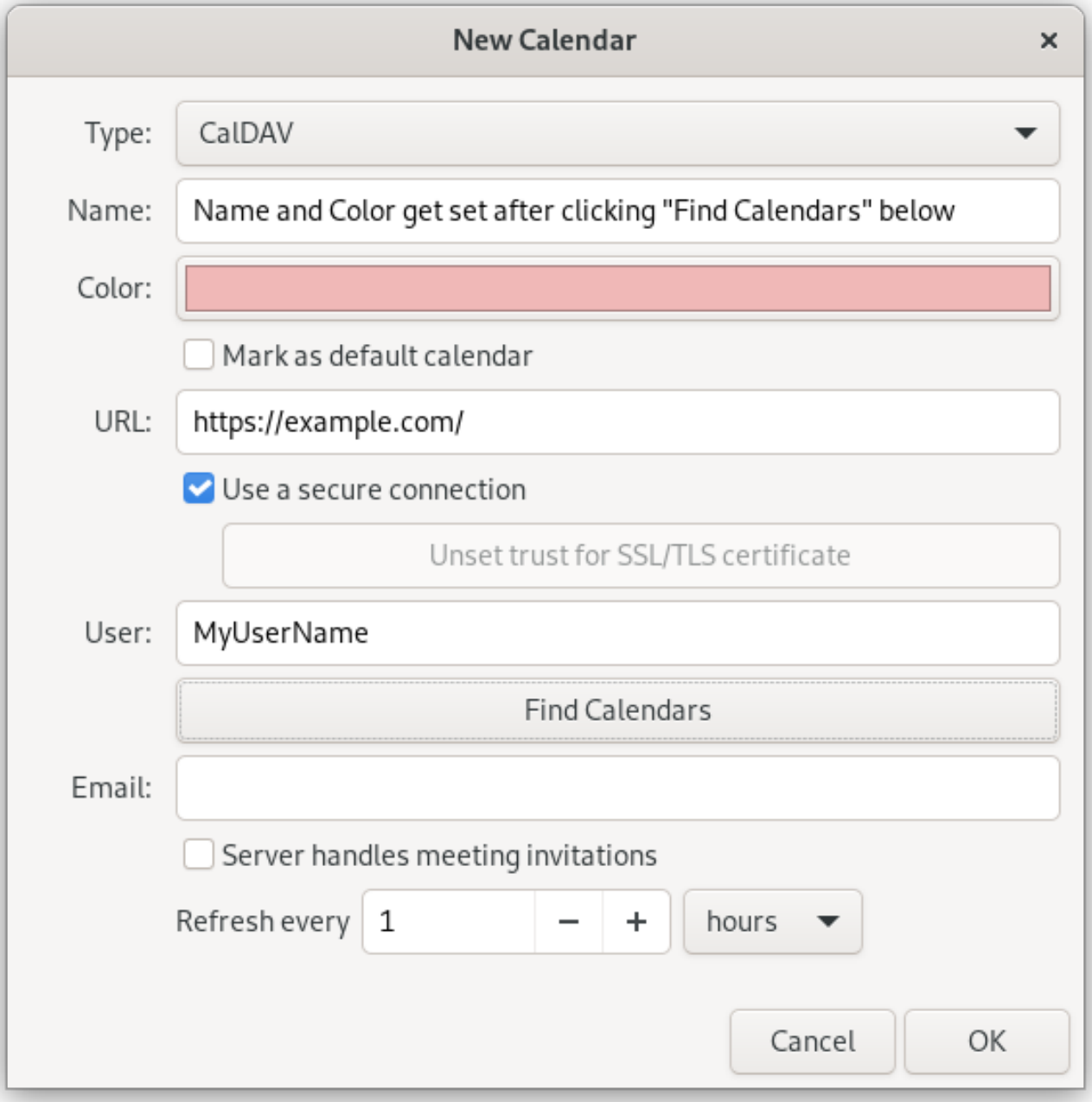


- Creating addressbook/calendar using the web interface
  - Visit `https://`*IP-address-or-domain-for-your-server*`/radicale/`
  - Log in with your FreedomBox account

  - Select "Create new addressbook or calendar"
  - Provide a title and select the type
  - Optionally, provide a description or select a color
  - Click "Create"
  - The page will show the URL for your newly created addressbook or calendar

Now open your client application to create new calendar and address books that will use your FreedomBox and Radicale server. The Radicale website provides an overview of supported clients, but do *not* use the URLs described there; FreedomBox uses another setup, follow this manual. Below are the steps for two examples:

- Example of setup with Evolution client:
  - Calendar
    1. Create a new calendar
    2. For "Type," select "CalDAV"
    3. When "CalDAV" is selected, additional options will appear in the dialogue window.
    4. URL: `https://`*IP-address-or-domain-for-your-server*. Items in *italics* need to be changed to match your settings.
    5. Enable "Use a secure connection."
    6. User: *USERNAME*. Your Freedombox user-name.
    7. Click on "Find Calendars"
    8. Enter your password and select a calendar



- TODO/Tasks list: Adding a TODO/Tasks list is basically the same as a calendar.
- Contacts
  * Follow the same steps described above and replace CalDAV with WebDAV.

5.25.3 Synchronizing over Tor

In FreedomBox, setting up a calendar with Radicale over Tor is the same as over the clear net. Here is a short summary:

1. When logged in to FreedomBox interface over Tor, click on Radicale, and at the prompt provide your FreedomBox user name and password.
2. In the Radicale web interface, log in using your FreedomBox user name and password.

3. Click on "Create new address book or calendar", provide a title, select a type, and click "Create".
4. Save the URL, e.g., `https://`*ONION-ADDRESS-FOR-YOUR-SERVER*`.onion/radicale/`*USERNAME*`/`*CALENDAR-CODE*`/`. Items in *italics* need to be changed to match your settings.

These instructions are for Thunderbird/Lightning. Note that you will need to be connected to Tor with the Tor Browser Bundle.

1. Open Thunderbird, install the Torbirdy add-on, and restart Thunderbird. (This may not be necessary.)
2. In the Lightning interface, under Calendar/Home in the left panel right click with the mouse and select "New calendar".
3. Select the location of your calendar as "On the Network".
4. Select CalDAV and for the location copy the URL, e.g., `https://`*ONION-ADDRESS-FOR-YOUR-SERVER*`.onion/radical`*CODE*`/`. Items in *italics* need to be changed to match your settings.
5. Provide a name, etc. Click "Next". Your calendar is now syncing with your FreedomBox over Tor.
6. If you have not generated a certificate for your FreedomBox with "Let's Encrypt", you may need to select "Confirm Security Exception" when prompted.

5.25.4 Synchronizing with your Android phone

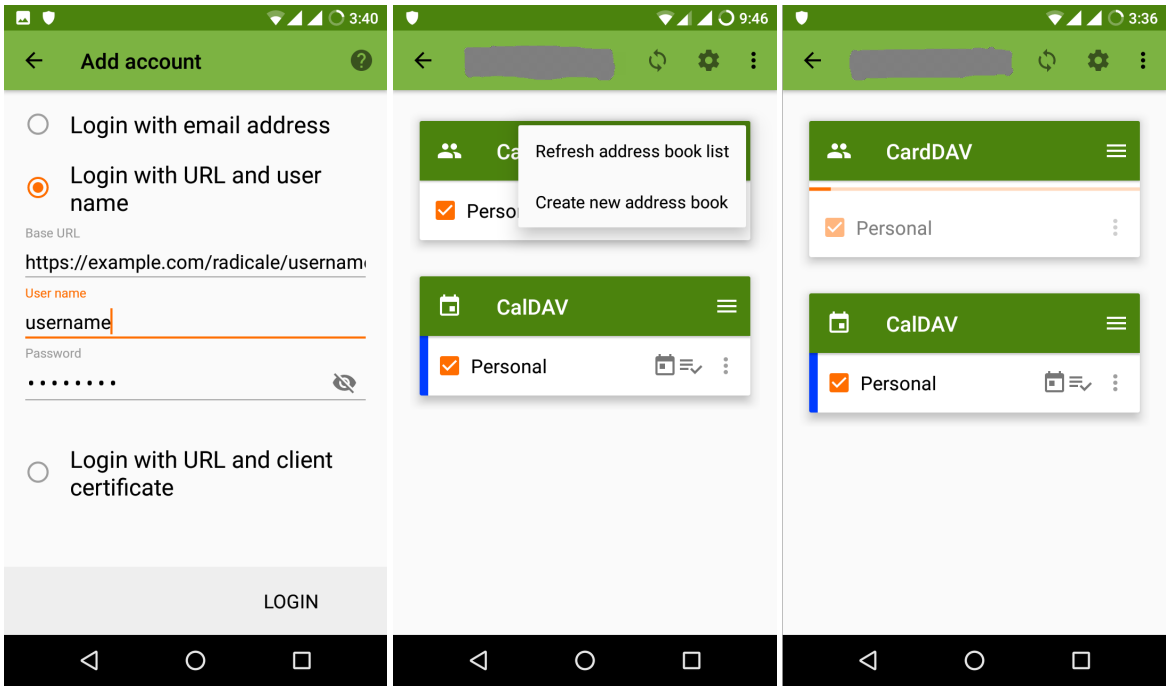
There are various Apps that allow integration with the Radicale server. This example uses DAVx5, which is available e.g. on F-Droid. If you intend to use ToDo-Lists as well, the compatible app OpenTasks has to be installed first.

Follow these steps for setting up your account with the Radicale server running on your FreedomBox.

1. Install DAVx5
2. Create a new account on DAVx5 by clicking on the floating + button.
3. Select the second option as shown in the first figure below and enter the base url as *https://<your.freedombox.address>/radicale/use*
(don't miss the / at the end). DAVx5 will be able to discover both CalDAV and WebDAV accounts for the user.
4. Follow this video from DAVx5 FAQ to learn how to migrate your existing contacts to Radicale.

Synchronizing contacts

1. Click on the hamburger menus of CalDAV and CardDAV and select either "Refresh ..." in case of existing accounts or "Create ..." in case of new accounts (see the second screenshot below).
2. Check the checkboxes for the address books and calendars you want to synchronize and click on the sync button in the header. (see the third screenshot below)



5.25.5 Advanced Users

5.25.5.1 Sharing resources

Above was shown an easy way to create a resource for a group of people by creating a dedicated account for all. Here will be described an alternative method where two users `User1` and `User2` are granted access to a calendar. This requires SSH-access to the FreedomBox.

1. create a file `/etc/radicale/rights`
   - ```
     [friends_calendar]
     user: ^(User1|User2)$
     collection: ^.*/calendar_of_my_friends.ics$
     permission: rw

     # Give write access to owners
     [owner-write]
     user: .+
     collection: ^%(login)s/.+$
     permission: rw
     ```
   - `[friends_calendar]` is just an identifier, can be any name.
   - The `[owner-write]` section makes sure that owners have access to their own files
2. edit file `/etc/radicale/config` and make the following changes in section `[rights]`
   - ```
     [rights]
     type = from_file
     file = /etc/radicale/rights
     ```
3. Restart the radicale server or the FreedomBox

5.25.5.2 Importing files

If you are using a contacts file exported from another service or application, it should be copied to: /var/lib/radicale/collections/*user*/*conta*
file name.vcf.

5.25.6 External links

- Upstream Project: https://radicale.org/3.0.html
- Upstream Documentation: https://radicale.org/v3.html#documentation-1
- Debian Radicale wiki: https://wiki.debian.org/Radicale

5.26 Roundcube (Email Client)



Available since: version 0.5

5.26.1 What is Roundcube?

Roundcube is a browser-based multilingual email client with an application-like user interface. Roundcube is using the Internet Message Access Protocol (IMAP) to access e-mail on a remote mail server. It supports MIME to send files, and provides particularly address book, folder management, message searching, and spell checking.

Roundcube app can be used to read and send emails in one of the two ways: using an email account you have elsewhere (other than FreedomBox), like in Riseup or in Gmail, or an account on FreedomBox. The latter is possible if an email server app, such as Postfix/Dovecot/Rspamd, is setup and enabled on FreedomBox. At present, Roundcube can only be setup for one of these two ways. This is controlled by the *Use only the local mail server* option in Roundcube app configuration page.

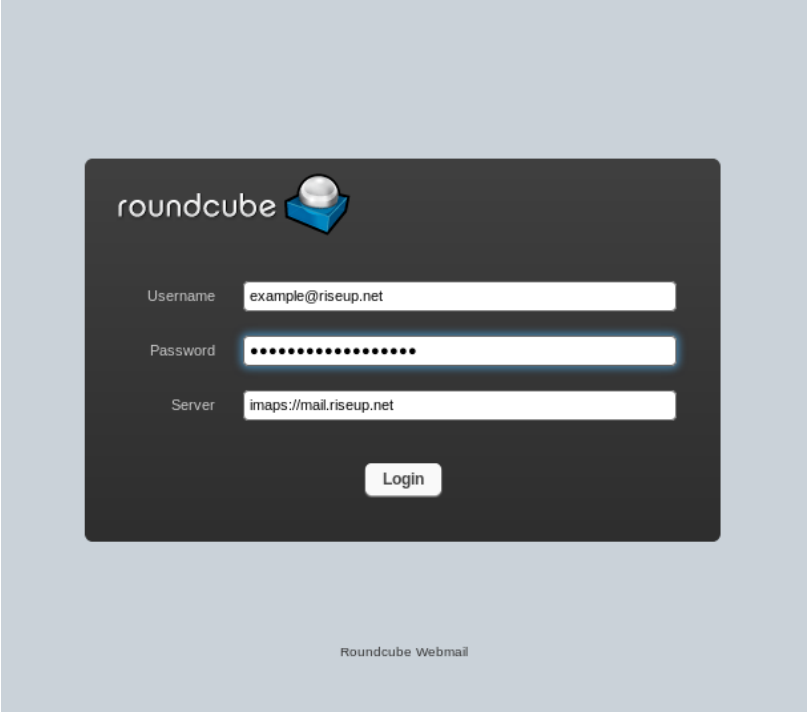
5.26.2 Email privacy

In the first case, you only get privacy over your drafts. Once you send the email, a copy will reside in the external services (Riseup, Gmail, etc) unless you explicitly remove it. In any case, your traffic transits through them.

In the second (self-hosted) case, your mail copies reside on your FreedomBox server. But you only keep privacy as long as the recipient also uses a private system and doesn't disclose your content.

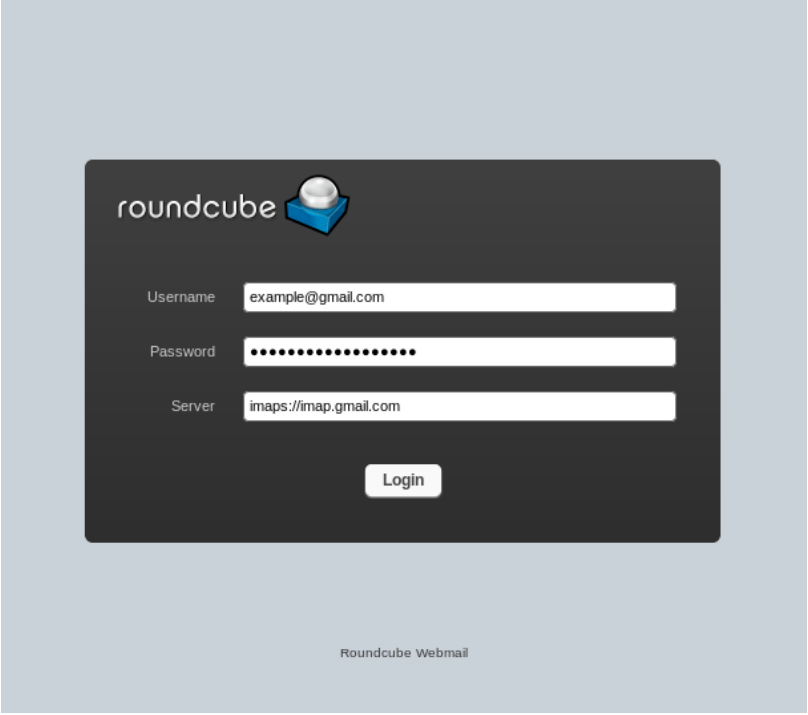
5.26.3 Using Roundcube

After Roundcube is installed, it can be accessed at `https://<your freedombox>/roundcube`. Enter your username and password. The username for many mail services will be the full email address such as *exampleuser@example.org* and not just the username like *exampleuser*. Enter the address of your email service's IMAP server address in the *Server* field. You can try providing your domain name here, such as *example.org* for email address *exampleuser@example.org* and if this does not work, consult your email provider's documentation for the address of the IMAP server. Using encrypted connection to your IMAP server is strongly recommended. To do this, prepend 'imaps://' at the beginning of your IMAP server address. For example, *imaps://imap.example.org*.



5.26.4 Using Gmail with Roundcube

If you wish to use Roundcube with your Gmail account, you need to first enable support for password based login in your Google account preferences. This is because Gmail won't allow applications to login with a password by default. To do this, visit Google Account preferences and enable *Less Secure Apps*. After this, login to Roundcube by providing your Gmail address as *Username*, your password and in the server field use *imaps://imap.gmail.com*.



5.26.5 External links

- Upstream Project: https://roundcube.net
- User Documentation: https://roundcube.net/support/

5.27 RSS Bridge (RSS Feed Generator)



Available since: version 22.16

5.27.1 What is RSS Bridge?

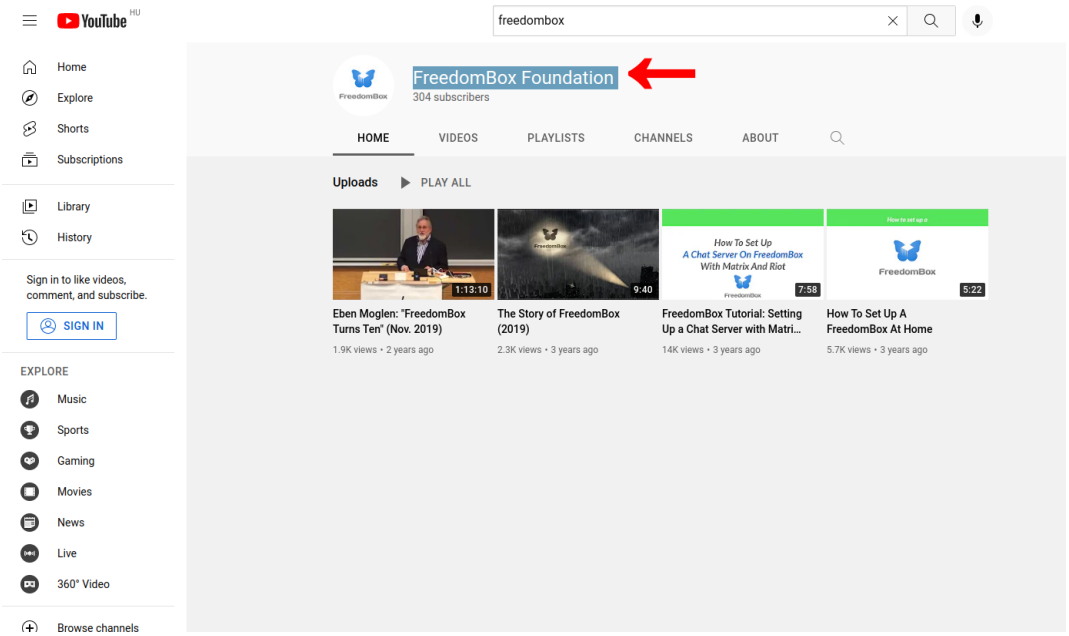
RSS-Bridge is a web application capable of generating RSS and Atom feeds for websites that don't have one. For example, with the help of RSS Bridge you can subscribe to YouTube channels without having to have a YouTube account.

5.27.2 Usage Example

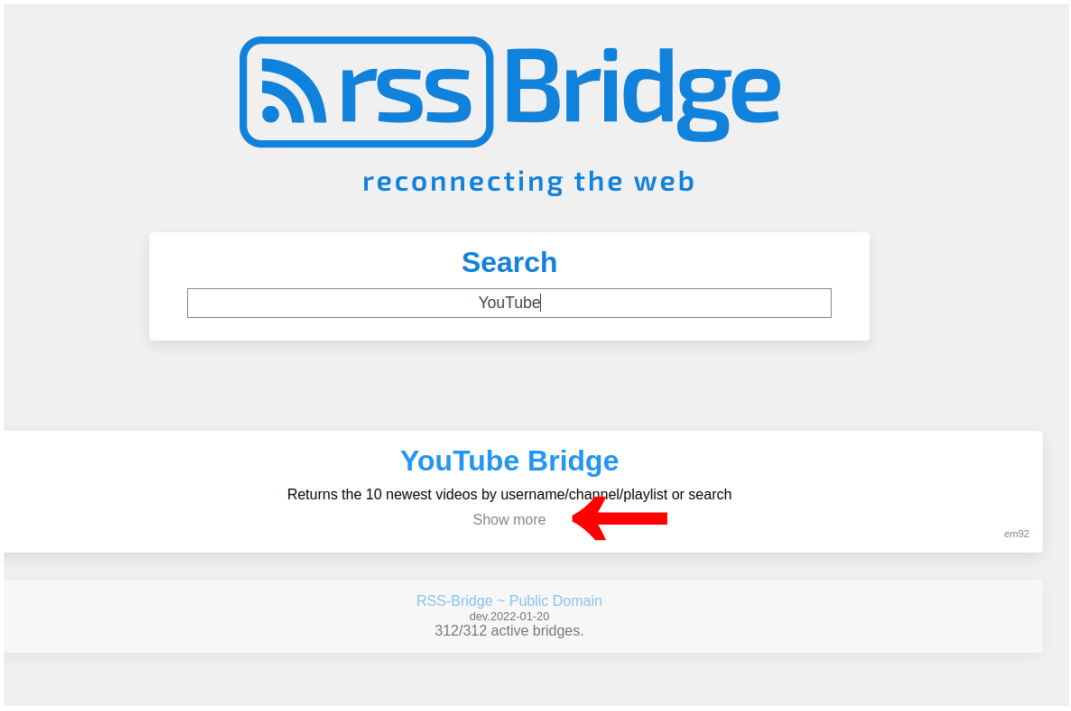
5.27.2.1 Subscribing to a YouTube account

In this example, we will see one of the ways to subscribe to a given YouTube channel.

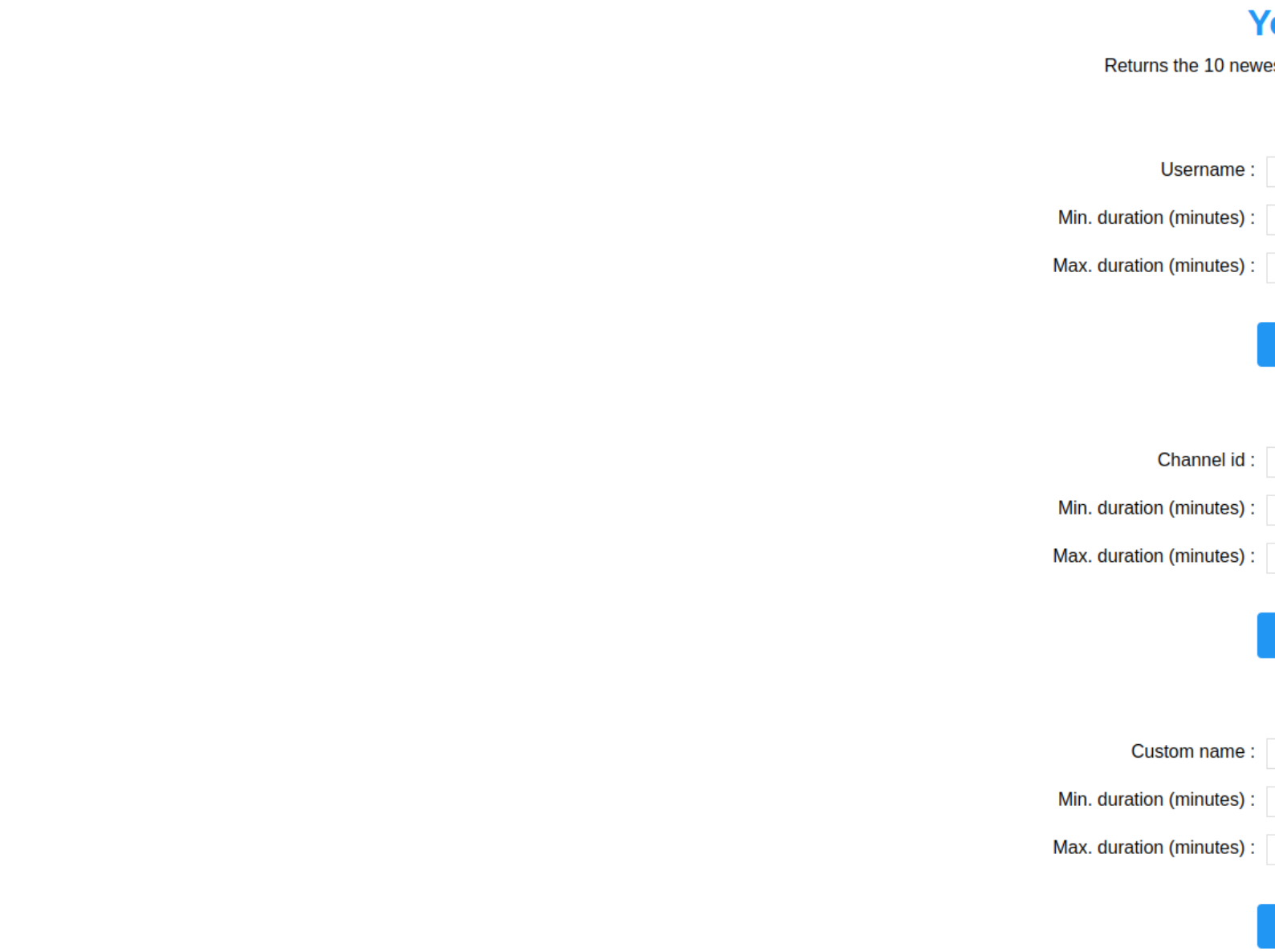
1. Visit the YouTube channel and copy its name to the clipboard



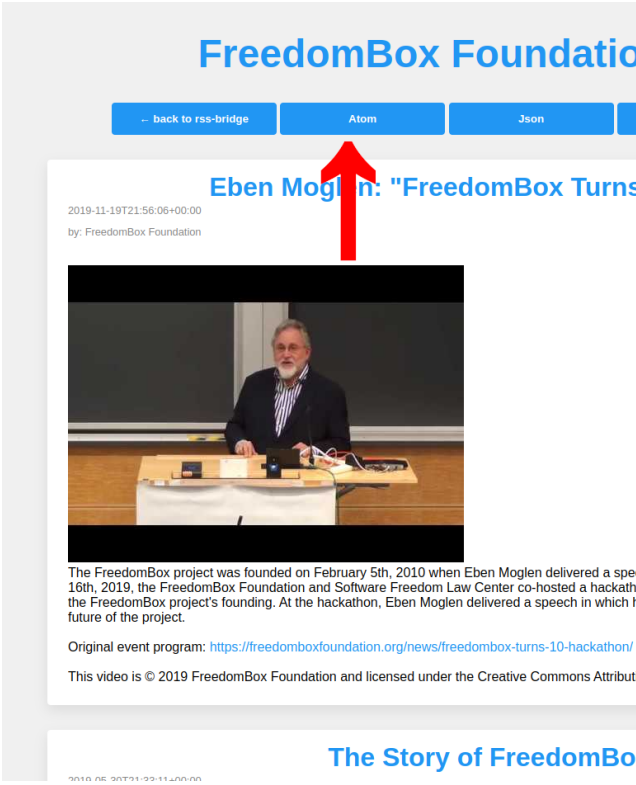
2.#2 Find "YouTube Bridge"



and click on **show more** 3.#3



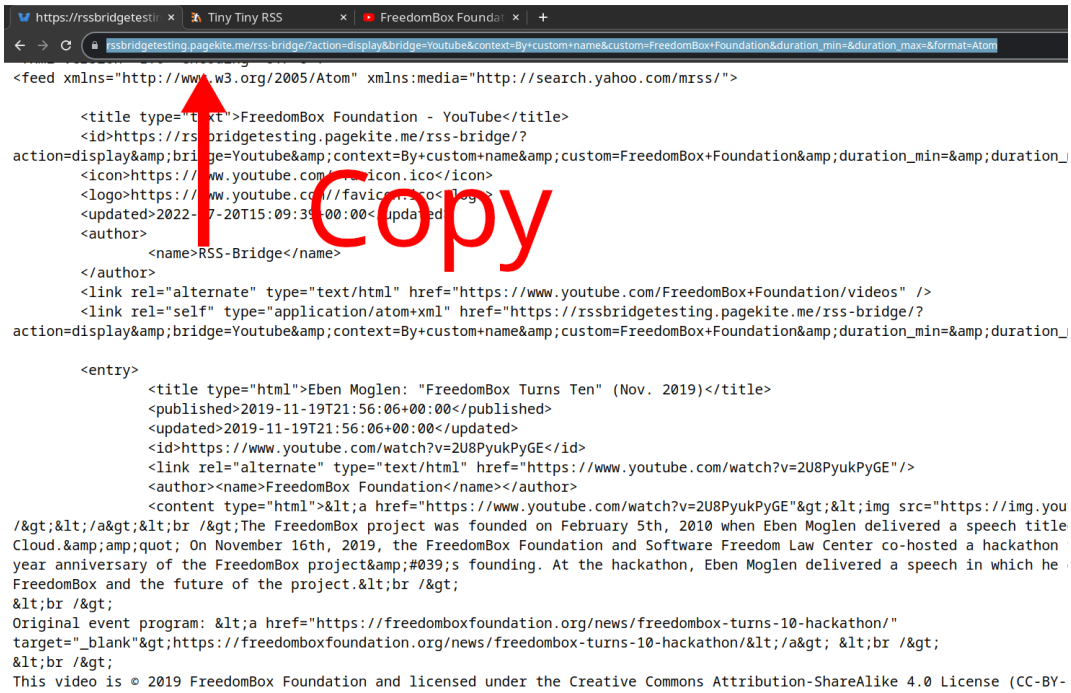
Paste the previously copied channel name in the **Custom name** section and click on **Generate Feed**
4.#4 From the available feed types select **Atom**. If you're using a Chromium based browser, this will open the Atom feed in a new



tab, which you can easily copy into your feed Reader, such as Tiny Tiny RSS

5.27.2.2 Subscribing to feed with Tiny Tiny RSS

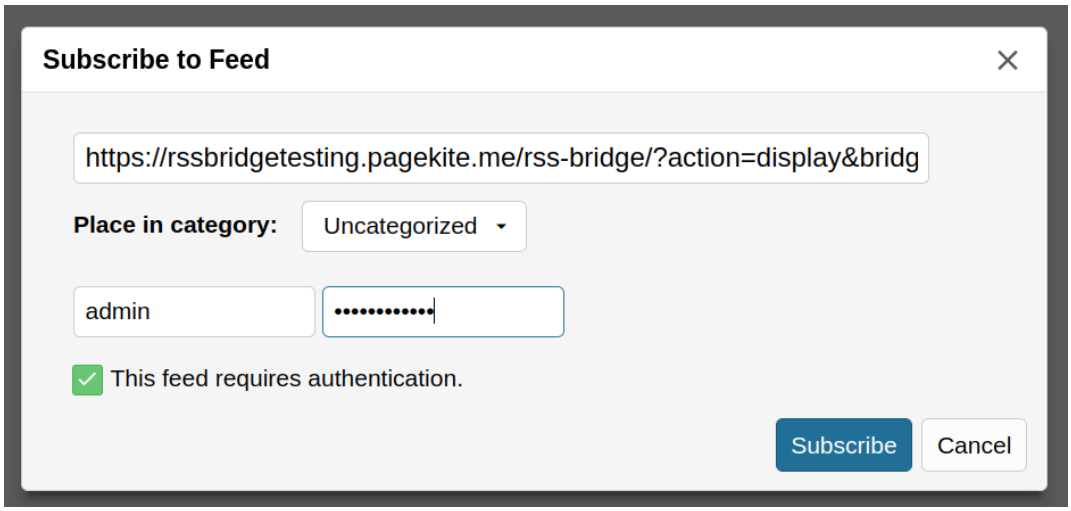
1. Copy the URL that RSS Bridge generated



2.#2 In Tiny Tiny RSS select **Subscribe to feed** from the drop-down menu on the right side.

1. Paste the generated link from step one into the textbox and select **This feed requires authentication.**

2. Submit your FreedomBox username and password and click on **Subscribe**

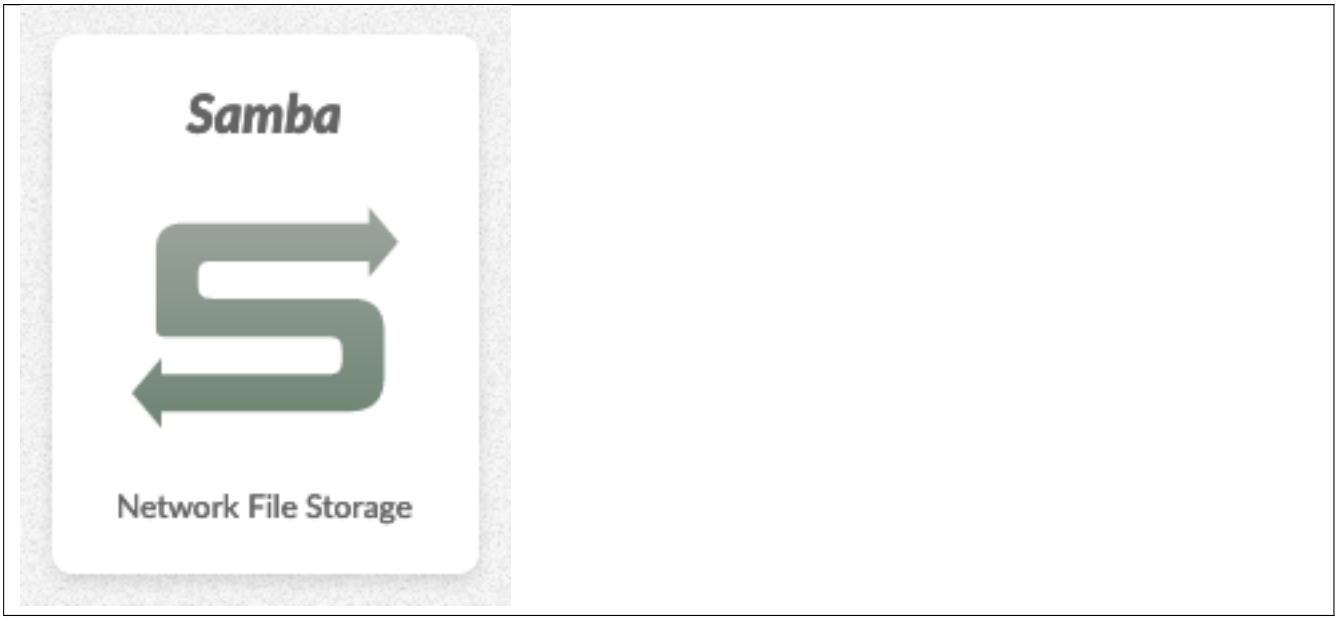


For a more detailed description of Tiny Tiny RSS, see its manual page

5.27.3 External links

- Upstream Project: https://rss-bridge.github.io/rss-bridge/
- User documentation: https://rss-bridge.github.io/rss-bridge/General/Project_goals.html

5.28 Samba (Network File Storage)



Available since: version 19.22

Samba lets you have shared folders over the local network that can be used from multiple computers running different operating systems. We refer to these shared folders as "shares".

You can have a personal folder shared between your own devices (Home share), a folder shared with a trusted group (Group share) or one that is shared with every device on the network (Open share).

Samba lets you to treat a share as if it's a local folder on your computer. However, shares are available only on the local network.

To learn more about Samba, please refer to the user documentation on their wiki.

5.28.1 Using Samba

After installation, you can choose which disks to use for sharing. Enabled shares are accessible in the file manager on your computer at location \\freedombox (on Windows) or smb://freedombox.local (on Linux and Mac). There are three types of shares you can choose from:

Open share - accessible to everyone in your local network.

Group share - accessible only to FreedomBox users who are in the *freedombox-share* group.

Home share - every user in the *freedombox-share group* can have their own private space.

5.28.1.1 Connecting from an Android device

To access Samba shares on an Android device, install "Android Samba Client" from F-Droid or Google Play. Enter *smb://freedombox.loc* as the share path in the app. Your shared folders should then be visible in the file manager app. Samba shares can also be used by VLC for Android which automatically discovers them.

5.28.1.2 Connecting from a macOS device

- *Open a Finder window on your Mac. *Use Go -> Connect to Server... from the file menu or press the shortcut Cmd+K to open the Connect To Server dialog. *Enter the address of your Samba share, e.g. *smb://192.168.0.105/disk* and click Connect.

5.28.2 Integration with other apps

Transmission app on FreedomBox provides a setting to allow downloads to be saved directly to a Samba share.

If you want to make available files synchronized with Syncthing through Samba you need to make sure you synchronize in a Samba share folder. Additionally in order to make Syncthing shares available in Samba Open share or Group share you will need to ensure you click "Permissions > Ignore" button under the "Advanced" tab in folder you wish in the Syncthing web UI. This will ensure that the files will be writable through Samba.

5.28.3 Comparison with other apps

5.28.3.1 Syncthing

Syncthing maintains a copy of the shared folder on each device that it is shared with. Samba maintains only one copy on your FreedomBox device.

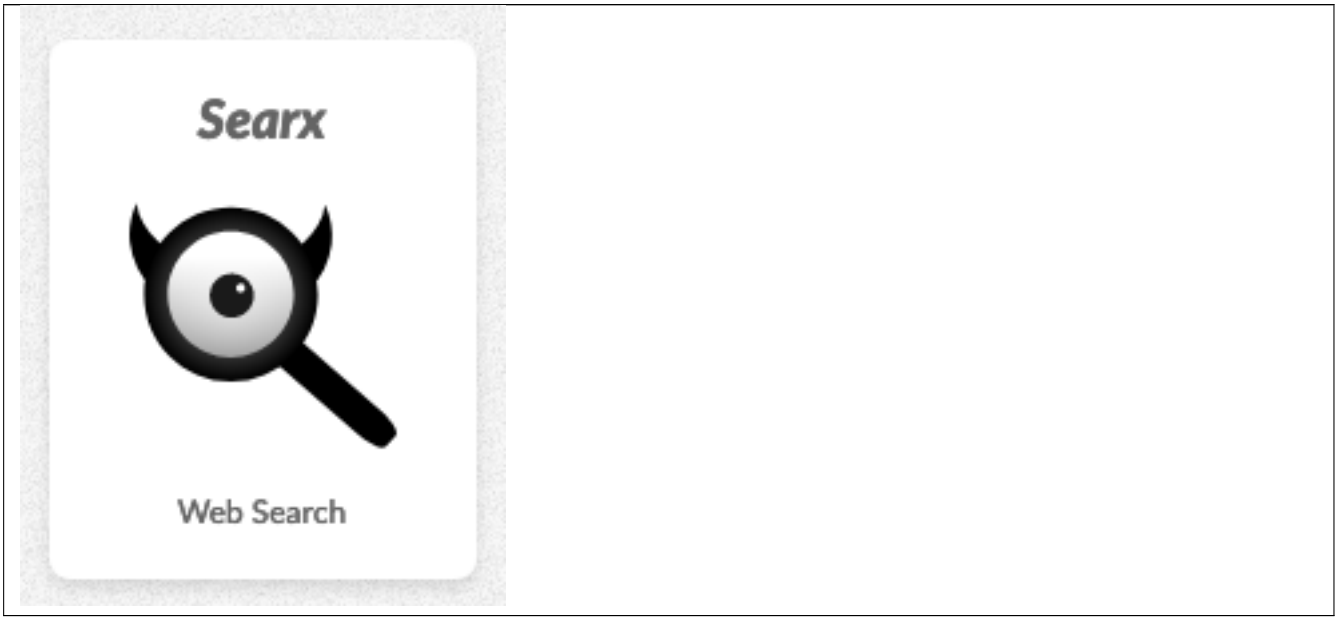
Syncthing can synchronize your shared folders between devices over the Internet. Samba shares are only available on the local network.

Since Syncthing is primarily a synchronization solution, it has features like conflict resolution and versioning. Samba has only copy of the file, so it doesn't need such features. For example, if two people are editing a spreadsheet stored on a Samba share, the last one to save the file wins.

5.28.4 External links

- Upstream Project: `https://www.samba.org`
- Upstream documentation: `https://www.samba.org/samba/docs`
- Debian Samba wiki: `https://wiki.debian.org/Samba`

5.29 Searx (Web Search)



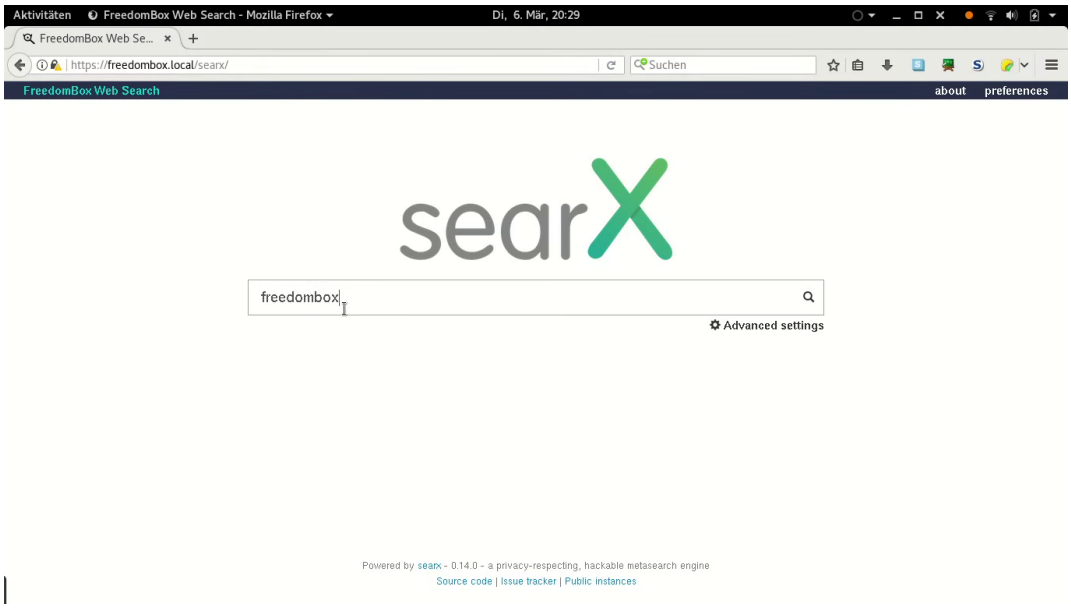
Available since: version 0.24.0

5.29.1 About Searx

Searx is a metasearch engine. A metasearch engine aggregates the results from various search engines and presents them in a unified interface.

Read more about Searx on their official website.

5.29.2 Screenshot



5.29.3 Screencast

Searx installation and first steps (14 MB)

5.29.4 Why use Searx?

5.29.4.1 Personalization and Filter Bubbles

Search engines have the ability to profile users and serve results most relevant to them, putting people into filter bubbles, thus distorting people's view of the world. Search engines have a financial incentive to serve interesting advertisements to their users, increasing their chances of clicking on the advertisements.

A metasearch engine is a possible solution to this problem, as it aggregates results from multiple search engines thus bypassing personalization attempts by search engines.

Searx avoids storing cookies from search engines as a means of preventing tracking and profiling by search engines.

5.29.4.2 Advertisement filtering

Searx filters out advertisements from the search results before serving the results, thus increasing relevance the of your search results and saving you from distractions.

5.29.4.3 Privacy

Searx uses HTTP POST instead of GET by default to send your search queries to the search engines, so that anyone snooping your traffic wouldn't be able to read your queries. The search queries wouldn't stored in browser history either.

Note: Searx used from Chrome browser's omnibar would make GET requests instead of POST.

5.29.5 Searx on FreedomBox

- Searx on FreedomBox uses Single Sign On. This means that you should be logged in into your FreedomBox in the browser that you're using Searx.
- SearX is easily accessible via Tor.
- Searx can be added as a search engine to the Firefox browser's search bar. See Firefox Help on this topic. Once Searx is added, you can also set it as your default search engine.
- Searx also offers search results in csv, json and rss formats, which can be used with scripts to automate some tasks.

5.29.6 External links

- Upstream Project: `https://searx.github.io/searx/`
- Upstream documentation: `https://searx.github.io/searx/user/index.html`
- Searx wiki: `https://github.com/searx/searx/wiki`

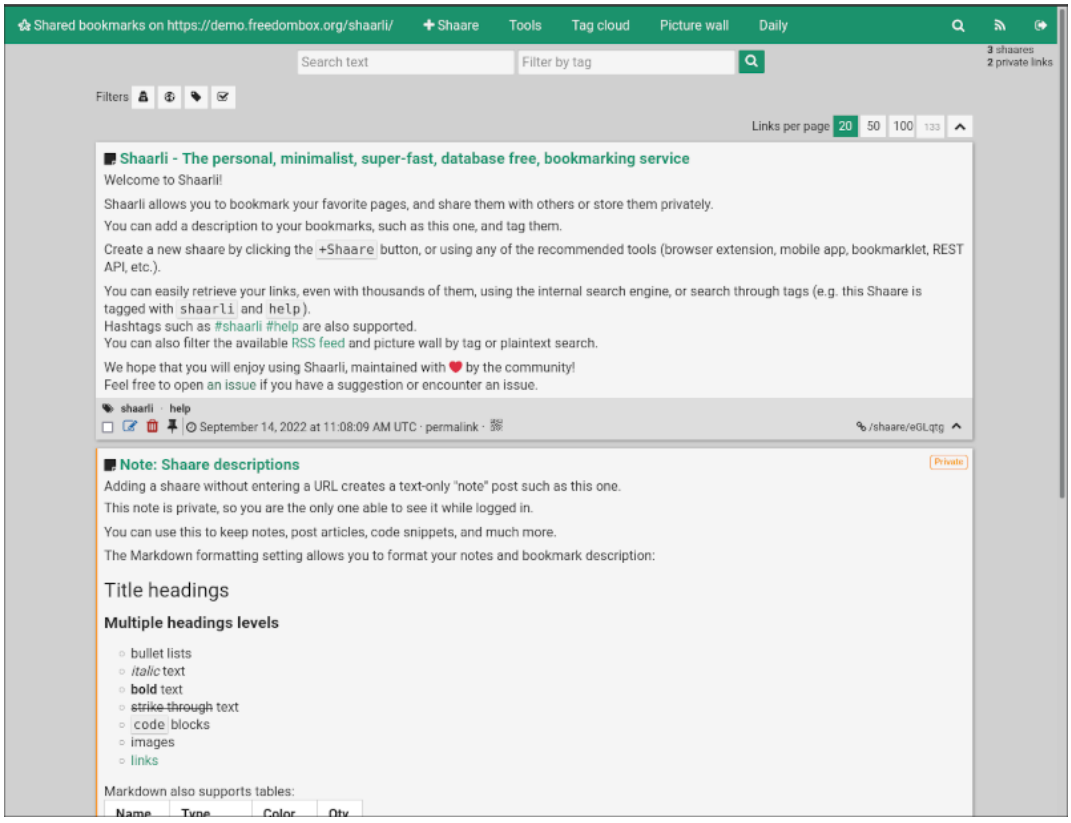
5.30 Shaarli (Bookmarks)



Available since: version 21.15

5.30.1 What is Shaarli?

Shaarli is personal (single-user) bookmarking application to install on your FreedomBox. It can also be used for micro-blogging, pastebin, online notepad and snippet archive. Shaarli is designed as a no-database delicious clone. As such, it provides very fast services, easy backup and import/export links as desktop or mobile browser bookmarks. Links stored can be public or private. Shaarli delivers ATOM and RSS feeds from its minimalist interface.



5.30.2 External links

- Upstream Project: https://shaarli.readthedocs.io/en/master/index.html
- Upstream documentation: https://shaarli.readthedocs.io/en/master/Usage.html

5.31 Shadowsocks (Bypass Censorship)



Available since: version 0.18.0

5.31.1 What is Shadowsocks?

Shadowsocks is a tool for securely forwarding network requests to a remote server. It consists of two parts: (1) a Shadowsocks server, and (2) a Shadowsocks client with a SOCKS5 proxy.

Shadowsocks can be used to bypass Internet filtering and censorship. This requires that the Shadowsocks server is in a location where it can freely access the Internet, without filtering.

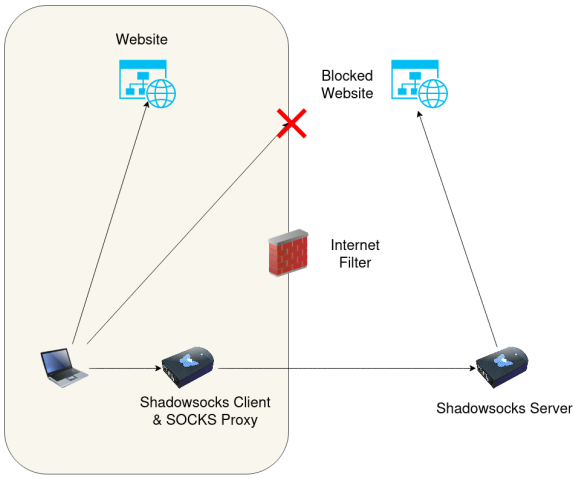
Your FreedomBox can run a Shadowsocks client which can connect to a Shadowsocks server. It will also run a SOCKS5 proxy. Local devices can connect to this proxy, and their data will be encrypted and proxied through the Shadowsocks server.

Alternatively, your FreedomBox can run a Shadowsocks server, that allows Shadowsocks clients to connect to it. Clients' data will be encrypted and proxied through this server.

5.31.2 Using Shadowsocks?

Shadowsocks can be used as follows:

- Shadowsocks Client (a FreedomBox) is in a region where some parts of the Internet are blocked or censored.
- Shadowsocks Server (a different FreedomBox, or another server) is in a different region, which doesn't have these blocks.
- The FreedomBox running Shadowsocks Client provides SOCKS proxy service on the local network for other devices to make use of its Shadowsocks connection to the server.



5.31.3 Configuring your FreedomBox for Shadowsocks Client

To enable Shadowsocks Client, first navigate to the Shadowsocks Client (Bypass Censorship) page, and install it.

Server: the Shadowsocks server is not this FreedomBox's IP or URL; rather, it will be another server or VPS that has been configured as a Shadowsocks server. There are also some public Shadowsocks servers listed on the web, but be aware that whoever operates the server can see where requests are going, and any non-encrypted data will be visible to them.

To use Shadowsocks Client after setup, set the SOCKS5 proxy URL in your device, browser or application to `http://freedombox_address:1080/`

5.31.4 Configuring your FreedomBox for Shadowsocks Server

To enable Shadowsocks Server, first navigate to the Shadowsocks Server (Help Others Bypass Censorship) page, and install it.

Note: In general, a FreedomBox should be set up as either a Shadowsocks Server, or a Shadowsocks Client, but not both!

For Shadowsocks Clients to connect to your server, they will need to know your domain name, the password, and the encryption method.

5.31.4.1 Port Forwarding

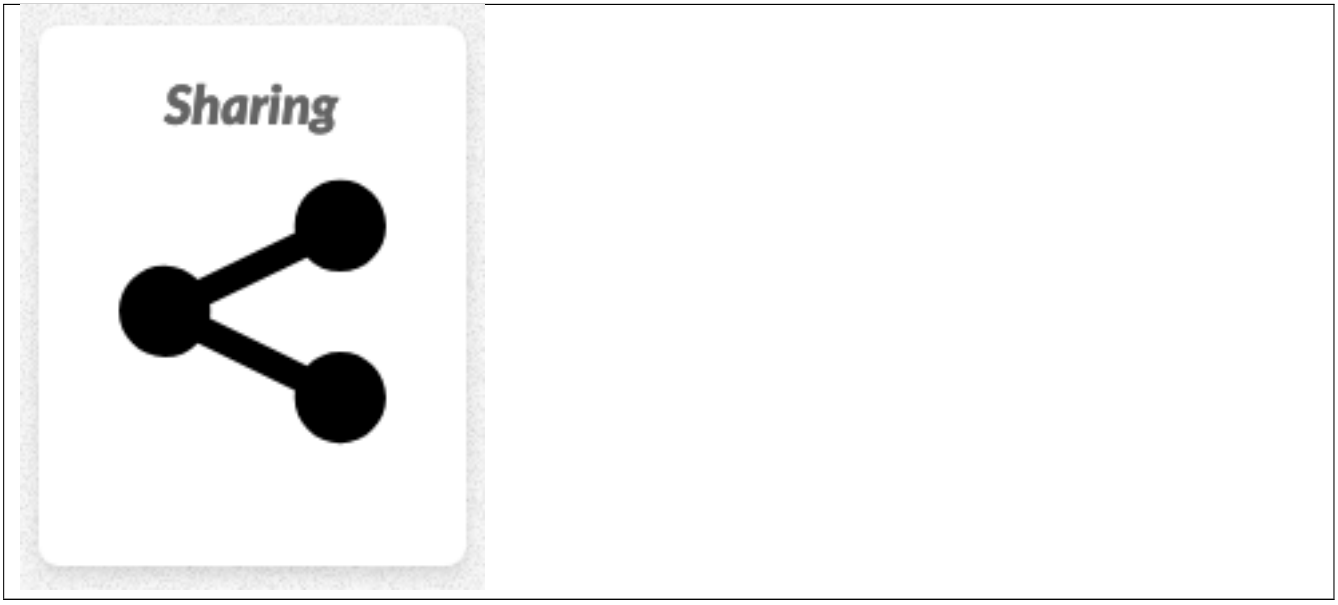
If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports for Shadowsocks Server:

- TCP 8388
- UDP 8388

5.31.5 External links

- Upstream Project: `https://shadowsocks.org/`
- Upstream Documentation: `https://shadowsocks.org/doc/what-is-shadowsocks.html`

5.32 Sharing (File Publishing)



Available since: version 0.25

5.32.1 What Is Sharing App?

Sharing app allows you to share content over the web. Shared content can be individual files or whole directories.

The content can be shared publicly or restricted to the users of listed allowed groups. Allowed users will be able to access the shared content from their web browser at `https://your_freedombox/share/content_name`. Users not belonging to any of the allowed groups won't see or access the content through this mechanism.

5.32.2 Setting Up Shares

- In FreedomBox web interface, enable the Sharing App. Only admins can create, edit or remove shares. They'll find the Sharing app in the Apps section of the FreedomBox web interface. Many shares can coexist in the same server.
- Add a new share
  - Give it a name (an thereby the URL) with which the users will ask for the content. In the example above it would be called *content_name*.
  - The Disk Path of the content to be served. This path is relative to *root* on your FreedomBox. For instance */var/lib/freedombox/sharing* might be a choice.
  - Sharing mode. On restricted mode, it also has the list of allowed groups. Only groups recognized by FreedomBox service can be combined in the list of allowed groups. Groups created in the CLI won't be offered by the Sharing app.
- Create the directory specified under *Disk Path* on your FreedomBox through *Cockpit*, *Nautilus* or remote login.
- Make sure the user, who will provide the content, has write access to that directory for instance by making him the owner of that directory.

5.32.3 Providing/Updating Content

The content can be created before or after the share is created and they can be updated independently.

The content doesn't need to be provided by an admin either. Any user with write access to the share's disk path can create or update it.

Multiple shares might point to the same content.

If you are user of FreedomBox and your admin refuses to create shares for you, and you don't need to restrict the access to your content, you still can fall back to the User Websites mechanism or the P2P networks (Deluge or Transmission for Torrent) to publish your files.

5.32.4 Technicalities

Sharing will share the content using the built-in Apache web server.

5.32.5 External Links

- Upstream Project: https://projects.apache.org/project.html?httpd-http_server
- Upstream Documentation: https://httpd.apache.org/docs/2.4/en/mod/mod_alias.html
- Debian Apache Wiki: https://wiki.debian.org/Apache

5.33 Syncthing (File Synchronization)



Available since: version 0.14

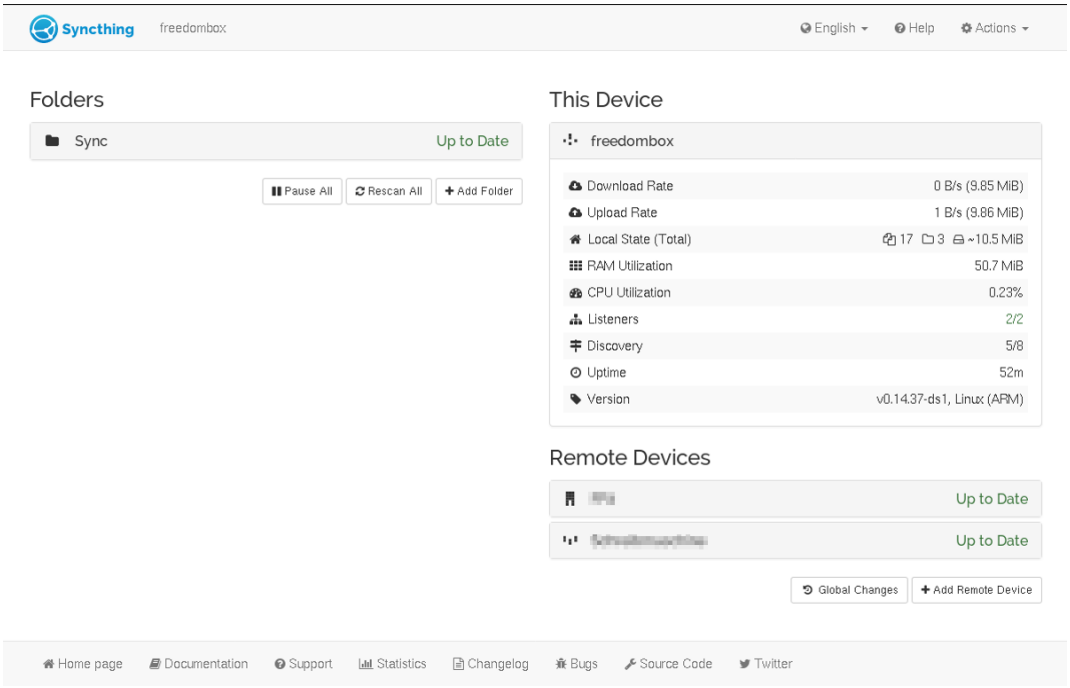
With *Syncthing* installed on your FreedomBox, you can synchronize content from other devices to your FreedomBox and vice-versa. For example, you can keep the photos taken on your mobile phone synchronized to your FreedomBox.

Users should keep in mind that Syncthing is a peer-to-peer synchronization solution, not a client-server one. This means that the FreedomBox isn't really the server and your other devices clients. They're all devices from Syncthing's perspective. You can use Syncthing to synchronize your files between any of your devices. The advantage that FreedomBox provides is that it is a server that's always running. Suppose you want your photos on your phone to be synchronized to your laptop, if you simply sync the photos to the FreedomBox, the laptop can get them from the FreedomBox whenever it comes online the next time. You don't have to be worried about your other devices being online for synchronization. If your FreedomBox is one of the devices set up with your Syncthing shared folder, you can rest assured that your other devices will eventually get the latest files once they come online.

After installation follow the instructions in the getting started of the Syncthing project. Syncthing allows individual folders to be selectively shared with other devices. Devices must be paired up before sharing by scanning QR codes or entering the device

ids manually. Syncthing has a discovery service for easily identifying the other devices on the same network having Syncthing installed.

In order to access to the web client of the Syncthing instance running on your FreedomBox, use the path */syncthing*. This web client is currently only accessible to the users of the FreedomBox that have administrator privileges, though it might be accessible to all FreedomBox users in a future release.



Syncthing has android apps available on the F-Droid and Google Play app stores. Cross-platform desktop apps are also available.

To learn more about Syncthing, please visit their official website and documentation.

5.33.1 Synchronizing over Tor

Syncthing should automatically sync with your FreedomBox even if it is only accessible as a Tor Onion Service.

If you would like to proxy your Syncthing client over Tor, set the `all_proxy` environment variable:

```
$ all_proxy=socks5://localhost:9050 syncthing
```

For more information, see the Syncthing documentation on using proxies.

5.33.2 Avoiding Syncthing Relays

Syncthing uses dynamic connections by default to connect with other peers. This means that if you are synchronizing over the Internet, the data might have to go through public Syncthing relays to reach your devices. This doesn't take advantage of the fact that your FreedomBox has a public IP address.

When adding your FreedomBox as a device in other Syncthing clients, set the address like "tcp://<my.freedombox.domain>" instead of "dynamic". This allows your Syncthing peers to directly connect to your FreedomBox avoiding the need for relays. It also allows for fast on-demand syncing if you don't want to keep Syncthing running all the time on your mobile devices.

5.33.3 Using Syncthing with other applications

5.33.3.1 Password Manager

Password managers that store their databases in files are suitable for synchronization using Syncthing. The following example describes using a free password manager called KeePassXC in combination with Syncthing to serve as a replacement for proprietary password managers that store your passwords in the cloud.

KeePassXC stores usernames, passwords etc. in files have the .kdbx extension. These kdbx files can be stored in a Syncthing shared folder to keep them synchronized on multiple machines. Free software applications which can read this file format are available for both desktop and mobile. You typically have to just point the application at the .kdbx file and enter the master password to access your stored credentials. For example, the same kdbx file can be accessed by using KeePassXC on desktop and KeePassDX on Android. KeePassXC can also be used to fill credentials into login fields in the browser by installing a browser extension.

5.33.3.2 Note-taking and Journaling

Several note-taking applications allow the notes to be stored in text files in plaintext or Markdown format. logseq is one such application that is licensed AGPLv3. It can be used to write a daily journal or discrete pages about various topics which can be linked in the form of a knowledge graph. Please note that the text is not encrypted at rest. logseq can also be used as a note-taking software by teams, instead of hosting a wiki.

A wiki software available on Debian desktops is Zim. You can set the storage for your personal zim wiki to a Syncthing folder.

If your notes are only ever going to be personal, a simpler solution exists. It is called a "quine". A wiki implemented as a quine is a single HTML file with enough embedded CSS and JavaScript in it to make it useful for note-taking, journaling and even documentation. After editing the quine's HTML file in a web browser you can use the "Save as" feature of the browser to overwrite the existing HTML file in your Syncthing folder, each time you want to save it. Quines are the simplest note-taking/personal wiki solution that work on all operating systems with no additional software required, other than a web browser. TiddlyWiki and FeatherWiki are popular software in this category.

5.33.4 External links

- Upstream project: `https://syncthing.net`
- Upstream documentation: `https://docs.syncthing.net`

5.34 TiddlyWiki (Non-linear Notebooks)



Available since: version 24.17

TiddlyWiki is an interactive, nonlinear web notebook for capturing and organizing complex information. It can serve as a knowledge management system for one person. It is portable (single HTML file), highly customizable and easy to use. Despite

being a single HTML file, TiddlyWiki can embed multiple types of media - images, audio, video and PDF files (or refer to them by file paths, not as portable). It can also store data in JSON format.

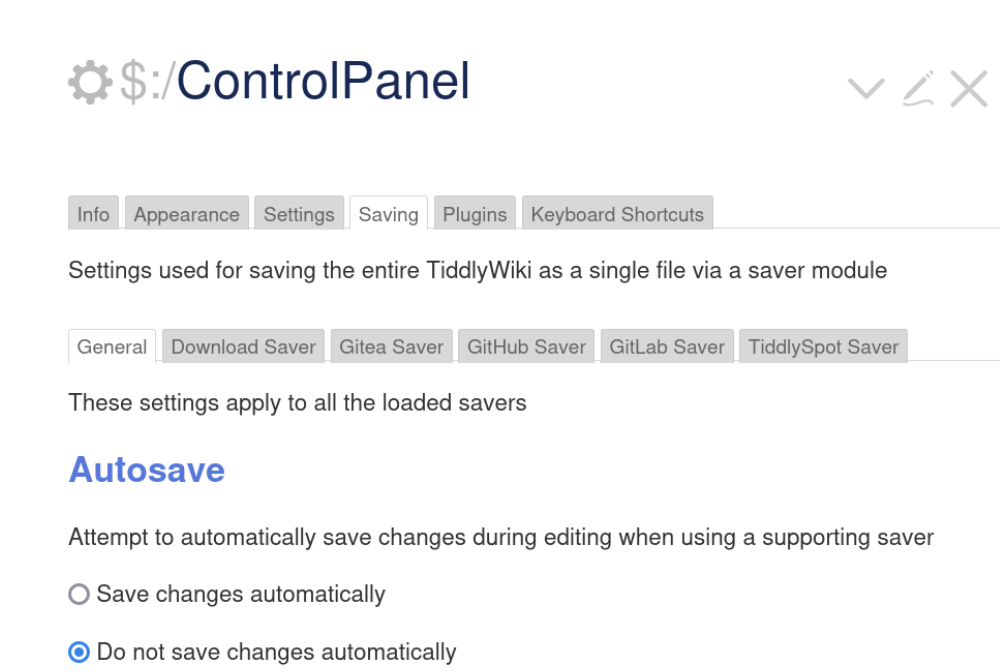
5.34.1 TiddlyWiki on FreedomBox

FreedomBox includes a native implementation of a Feather Wiki Nest (a server to save your wiki files). The same is used to host TiddlyWiki files. This implementation is not as efficient as the official TiddlyWiki server which can save one Tiddler at a time, instead of saving the entire HTML file.

5.34.1.1 Bandwidth considerations with the Autosave setting

Each empty TiddlyWiki file (2.4 MB) is 40x the size of an empty Feather Wiki file (60 KB). TiddlyWiki has autosave enabled by default. This causes the entire wiki file to be transmitted to the server, each time an individual Tiddler is saved. You might write long wiki pages in Feather Wiki, but TiddlyWiki encourages writing lots of short Tiddlers which results in more saves for the same amount of text. TiddlyWiki files tend to grow large as you add media content, with people growing their wikis to 100 MB in size over the years.

If you are editing while using mobile data or you are connected via a dial-up connection, hundreds of (2.4 MB + content) saves can add up quickly. In such cases, it is recommended to disable autosave from the settings and save manually.



5.34.2 Use cases

- **Personal Wiki**: TiddlyWiki can be used as a personal wiki to store all kinds of information in short notes called Tiddlers. It can hold a small database of your personal knowledge that is linked in a graph and easy to search and organize.
- **Research Notes**: TiddlyWiki can be used as a zettelkasten or a second brain. This can lead to new creative insights across different concepts.
- **Encrypted Journal/Diary**: TiddlyWiki supports password-protection and journal entries (a Tiddler with today's date as the title). Your personal diary/journal can be one TiddlyWiki file. Engineering Daybooks is a similar use case, but for work.
- TiddlyWiki can be used as a project management tool, e.g. for wedding planning.

- TiddlyWiki's WikiText though declarative can be used to create interactive content such as flash cards and automatically-updating lists. There is support for templates and JavaScript macros.
- **Linkblog**: A TiddlyWiki file can work as a replacement for Shaarli if you want to keep all of your bookmarks private.
- TiddlyWiki can be used as a bug tracker for your projects on GitWeb.

5.34.3 Upgrades

Upgrading TiddlyWiki is a manual process.

1. Save a local copy of the file from your browser. Right click -> Save page as...
2. Use the web-based Upgrade Wizard from the official website.
3. After downloading the upgraded file, load it into a browser to verify your content.
4. Replace the older file on your FreedomBox with the upgraded file.

If you ever lose a TiddlyWiki file, you can always retrieve a slightly outdated copy from the Backups app. It is better to keep your local copy after the upgrade, in case you want to revert.

5.34.4 Tips

5.34.4.1 Setting a favicon

Unlike Feather Wiki, TiddlyWiki does not automatically use your FreedomBox's favicon. To set it, follow these steps:

1. Create a new tiddler with the title `$:/favicon.ico`. Leave the content section empty
2. In the `Type` dropdown, select `ICO icon (image/x-icon)`
3. In the `Add a new field:` section, set the first field to `_canonical_uri` and the second field to `https://<your-freed`
4. Save the new tiddler and the wiki

Reference: Setting a favicon: TiddlyWiki

You can also have a custom image as the favicon for each TiddlyWiki. Using a distinct favicon makes it easier to identify the tab in your browser. Let's say your TiddlyWiki file is your personal journal and you want to set the favicon to the image "notebook.png" which looks like a diary

1. Open the folder the image is in, using your desktop file explorer
2. Drag and drop the file into your TiddlyWiki's browser tab
3. A special tiddler called `$:/import` will be opened
4. In the special tiddler, you will have an option to rename the file. Set the file name to `$:/favicon.ico`
5. Click the `Import` button and save the wiki

5.34.5 External links

- Website: `https://tiddlywiki.com`
- Grok TiddlyWiki (online e-book): `https://groktiddlywiki.com/read/`
- Source code for TiddlyWiki: `https://github.com/TiddlyWiki/TiddlyWiki5`

5.35 Tiny Tiny RSS (News Feed Reader)



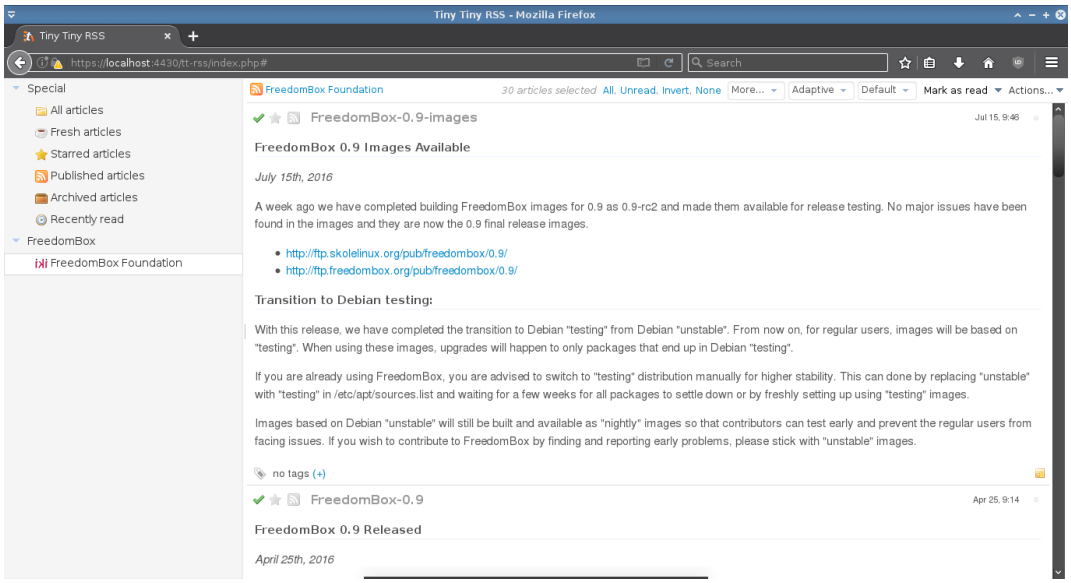
Available since: This app is no longer available since Debian Trixie. Please migrate to Miniflux or Nextcloud News.

Tiny Tiny RSS is a news feed (RSS/Atom) reader and aggregator, designed to allow reading news from any location, while feeling as close to a real desktop application as possible.

Any user created through FreedomBox web interface will be able to login and use this app. Each user has their own feeds, state and preferences.

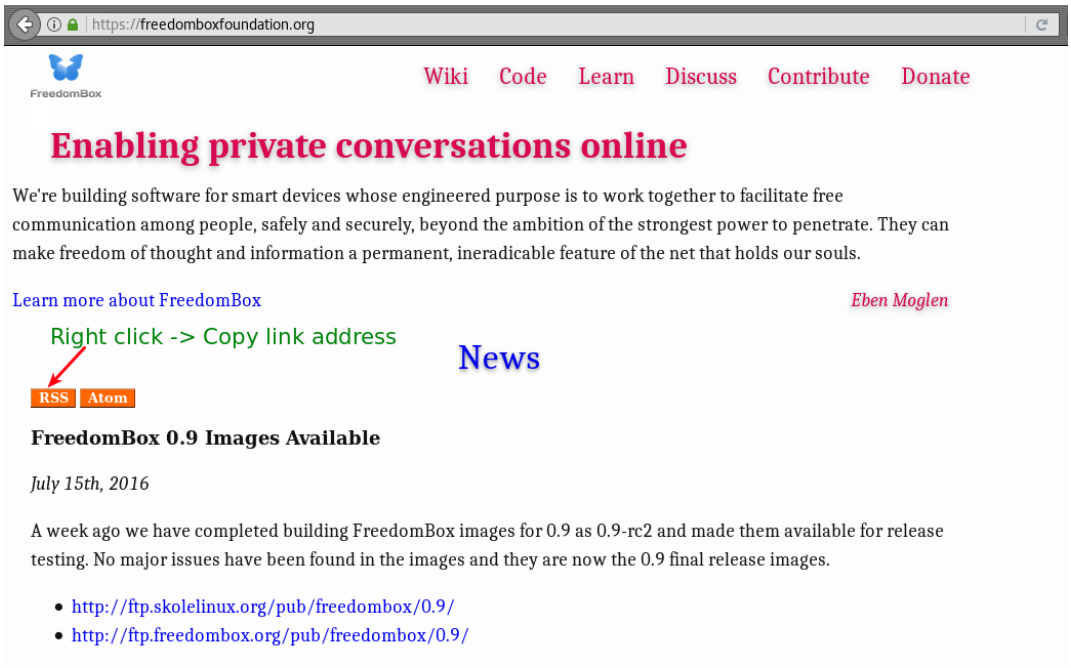
5.35.1 Using the Web Interface

When enabled, Tiny Tiny RSS will be available from */tt-rss* path on the web server. Any user created through FreedomBox will be able to login and use this app.

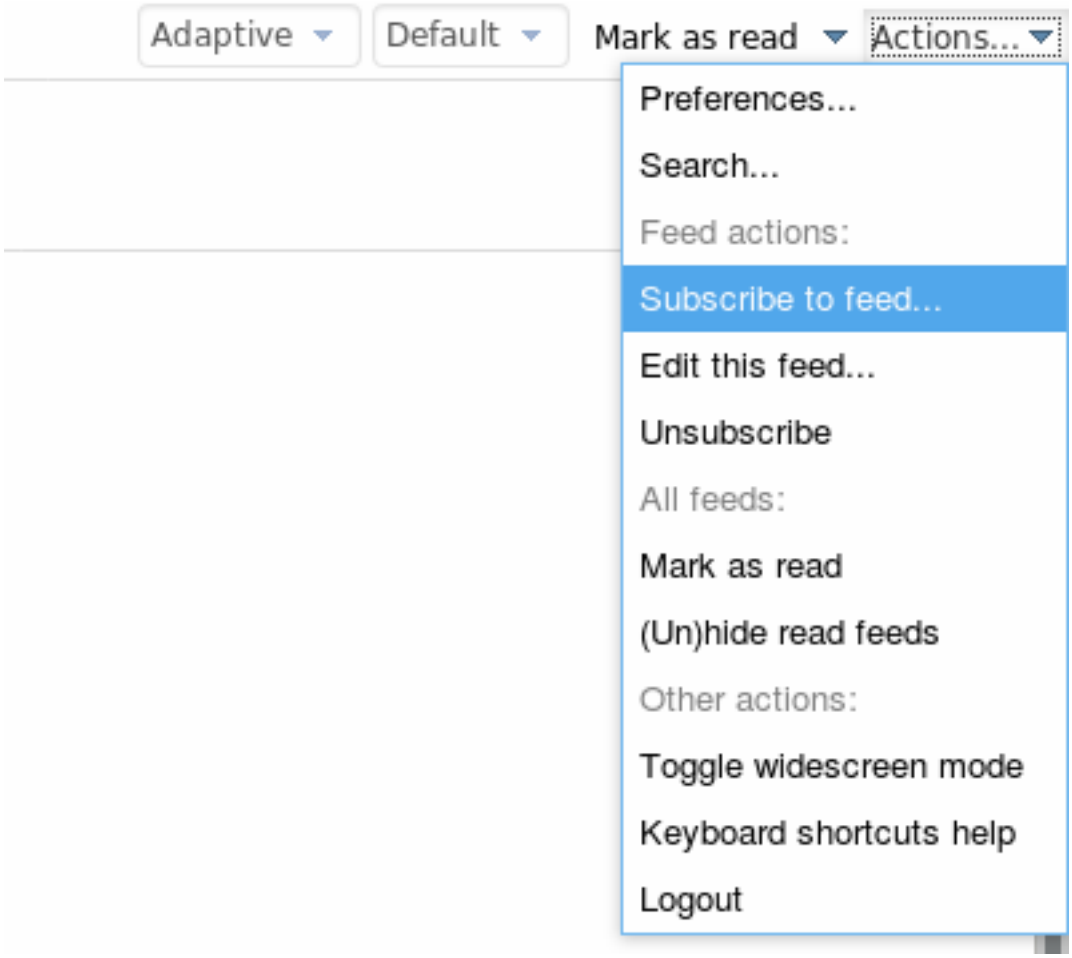


5.35.1.1 Adding a new feed

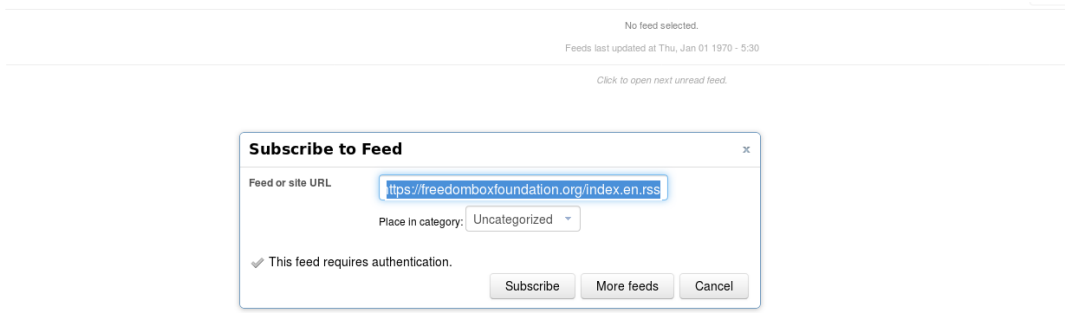
1. Go to the website you want the RSS feed for and copy the RSS/Atom feed link from it.



2. Select "Subscribe to feed.." from the Actions dropdown.



3. In the dialog box that appears, paste the URL for copied in step 1 and click the **Subscribe** button.

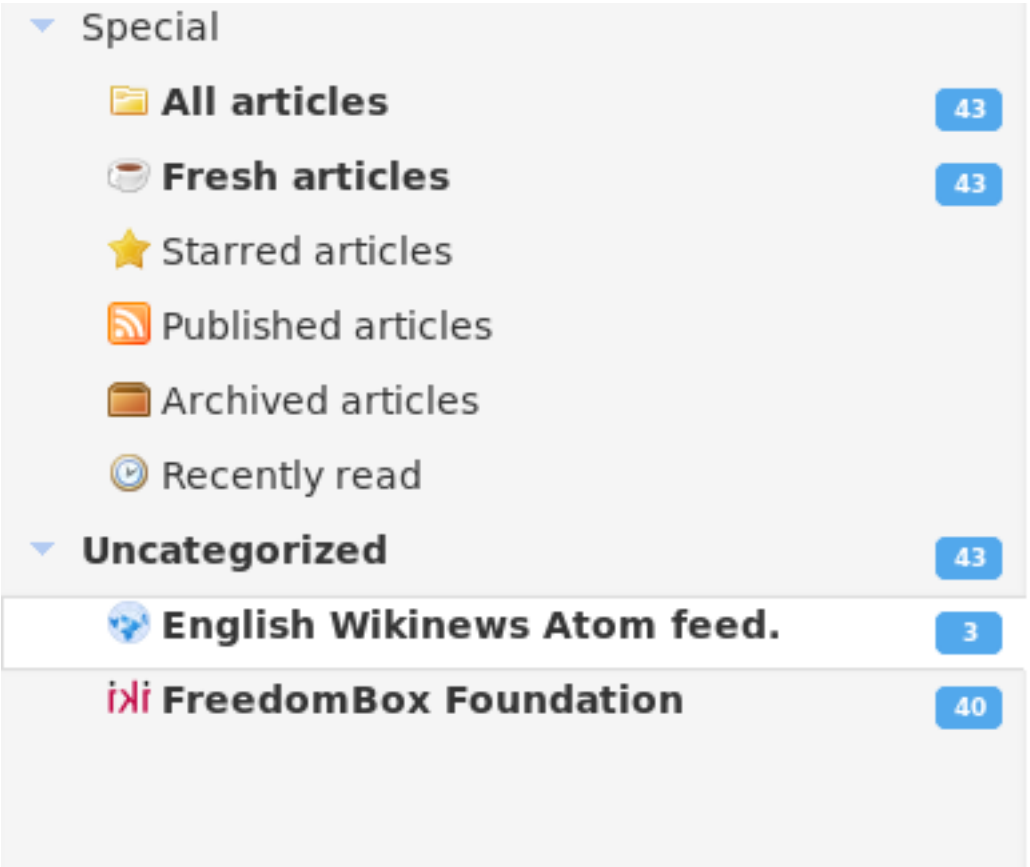


Give the application a minute to fetch the feeds after clicking Subscribe.

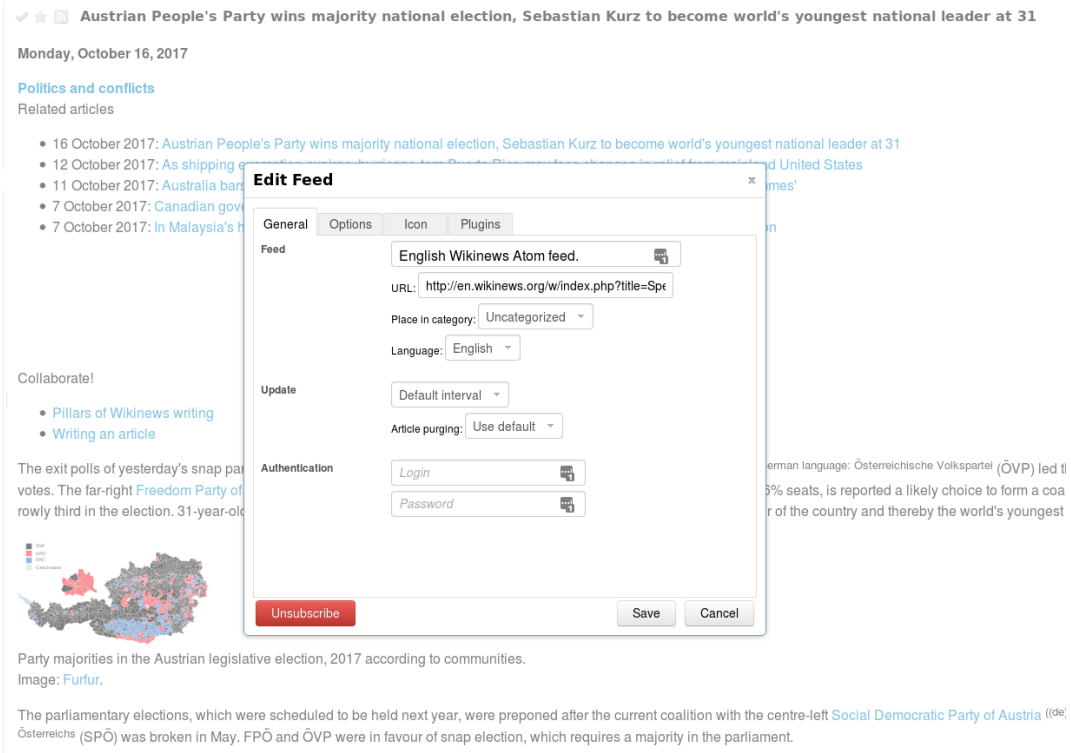
In some websites, the RSS feeds button isn't clearly visible. In that case, you can simply paste the website URL into the Subscribe dialog (step 3) and let TT-RSS automatically detect the RSS feeds on the page.

You can try this now with the homepage of WikiNews

As you can see in the image below, TT-RSS detected and added the Atom feed of WikiNews to our list of feeds.



If you don't want to keep this feed, right click on the feed shown in the above image, select **Edit feed** and click **Unsubscribe** in the dialog box that appears.

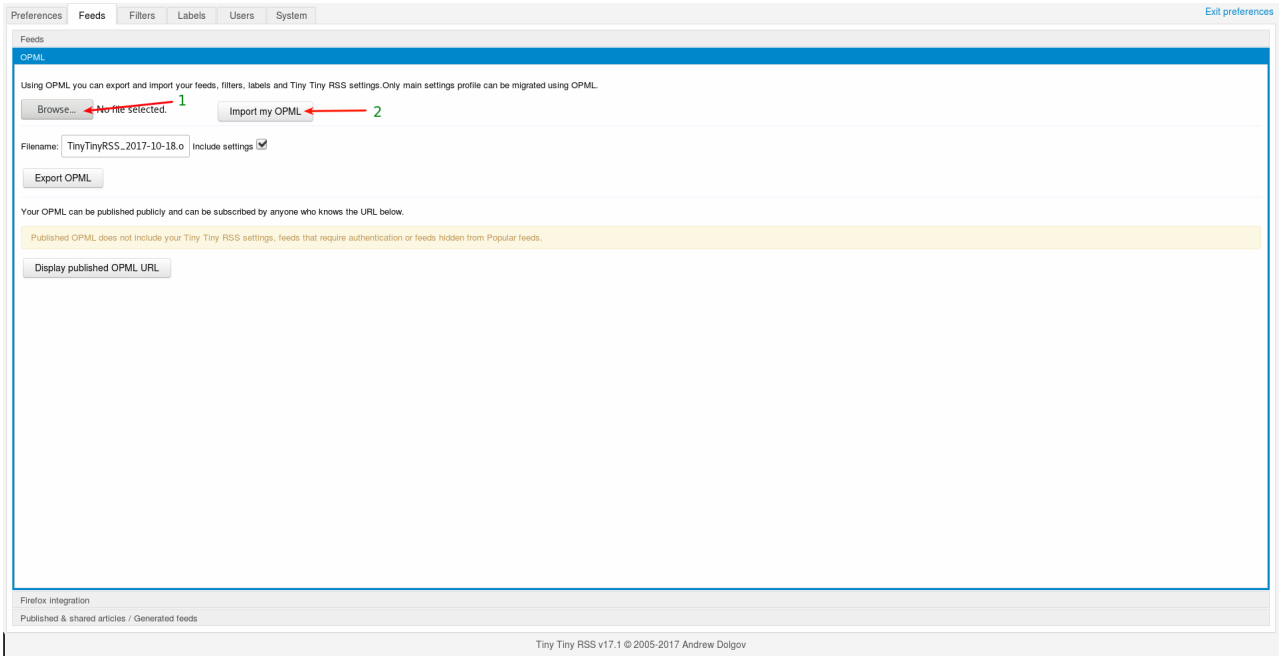


5.35.1.2 Importing your feeds from another feed reader

In your existing feed reader, find an option to *Export* your feeds to a file. Prefer the OPML file format if you have to choose between multiple formats. Let's say your exported feeds file is called Subscriptions.opml

Click on the *Actions* menu at the top left corner and select *Preferences*. You will be taken to another page.

Select the second tab called *Feeds* in the top header. Feeds has several sections. The second one is called *OPML*. Select it.



To import your Subscriptions.opml file into TT-RSS,

1. Click *Browse* and select the file from your file system

2. Click *Import my OPML*

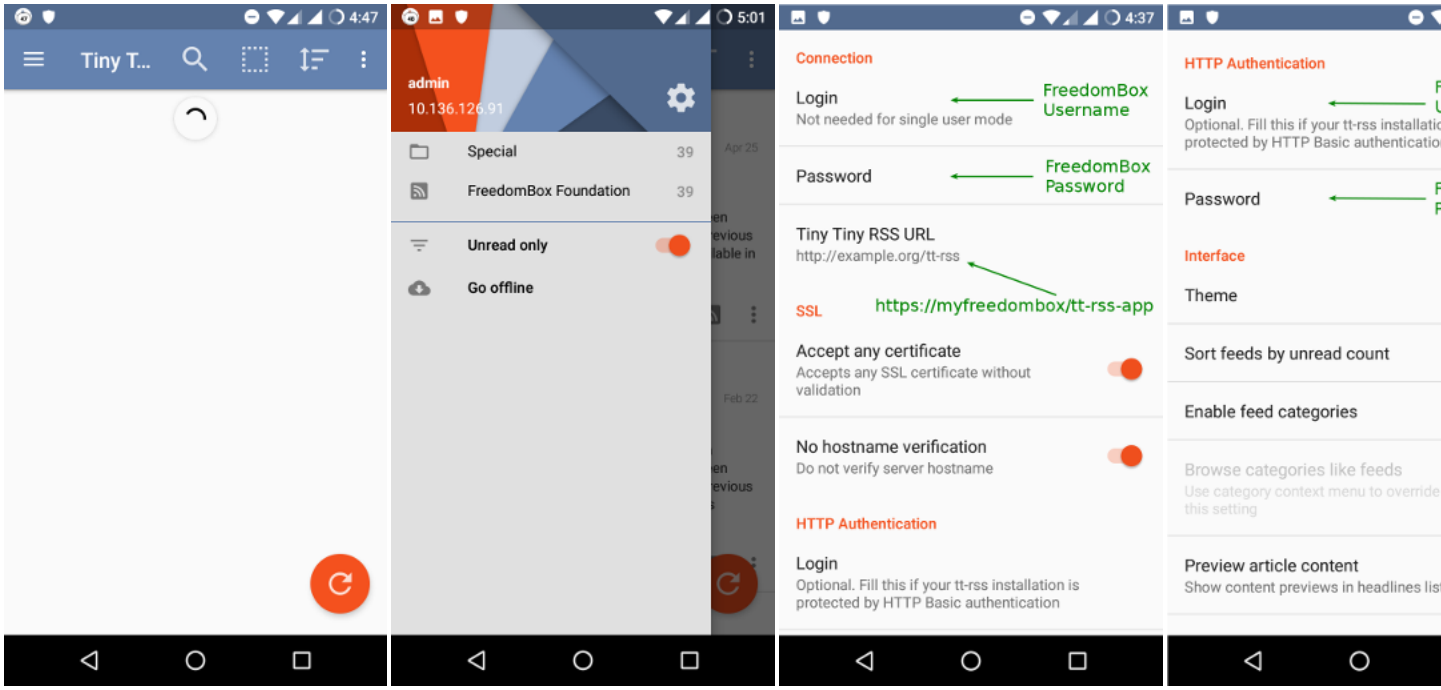
After importing, you'll be taken to the **Feeds** section that's above the OPML section in the page. You can see that the feeds from your earlier feed reader are now imported into Tiny Tiny RSS. You can now start using Tiny Tiny RSS as your primary feed reader.

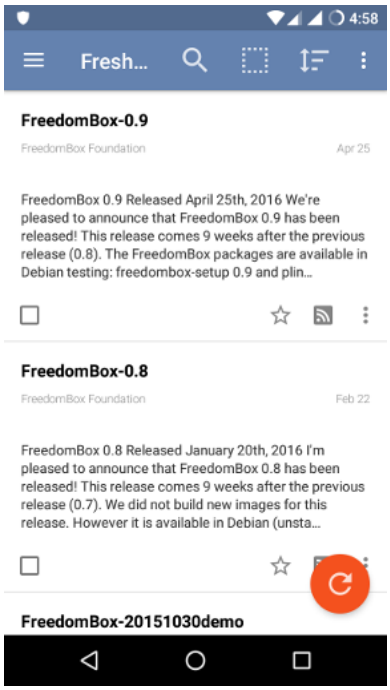
In the next section, we will discuss setting up the mobile app, which can let you read your feeds on the go.

5.35.2 Using the Mobile App

The official Android app from the Tiny Tiny RSS project works with FreedomBox's Tiny Tiny RSS Server. The older TTRSS-Reader application is known **not** to work.

To configure, first install the application, then in the setting page, set URL as *https://<your.freedombox.address>/tt-rss-app/*. Set your user name and password in the Login details as well as HTTP Authentication details. If your FreedomBox does not have a valid HTTPS certificate, then in settings request allowing any SSL certificate and any host.





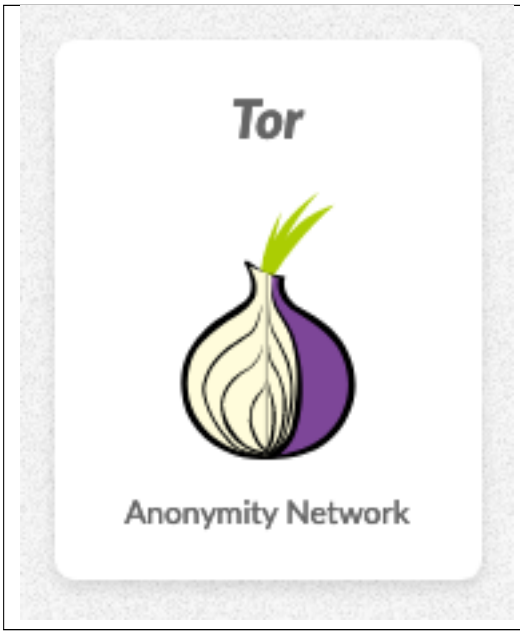
5.35.3 RSS Bridge

RSS Bridge can be used with Tiny Tiny RSS to generate Atom/RSS links for websites that don't provide one.

5.35.4 External links

- Upstream project: `https://tt-rss.org`
- Upstream Documentation: `https://tt-rss.org/wiki.php`

5.36 Tor (Anonymity Network)



Available since: version 0.3

5.36.1 What is Tor?

Tor is a network of servers operated by volunteers. It allows users of these servers to improve their privacy and security while surfing on the Internet. You and your friends are able to access to your FreedomBox via Tor network without revealing its IP address. Activating Tor application on your FreedomBox, you will be able to offer remote services (chat, wiki, file sharing, etc...) without showing your location. This application will give you a better protection than a public web server because you will be less exposed to intrusive people on the web.

To download software packages over Tor, or to run a SOCKS proxy, see the Tor Proxy app.

5.36.2 Using Tor to browse anonymously

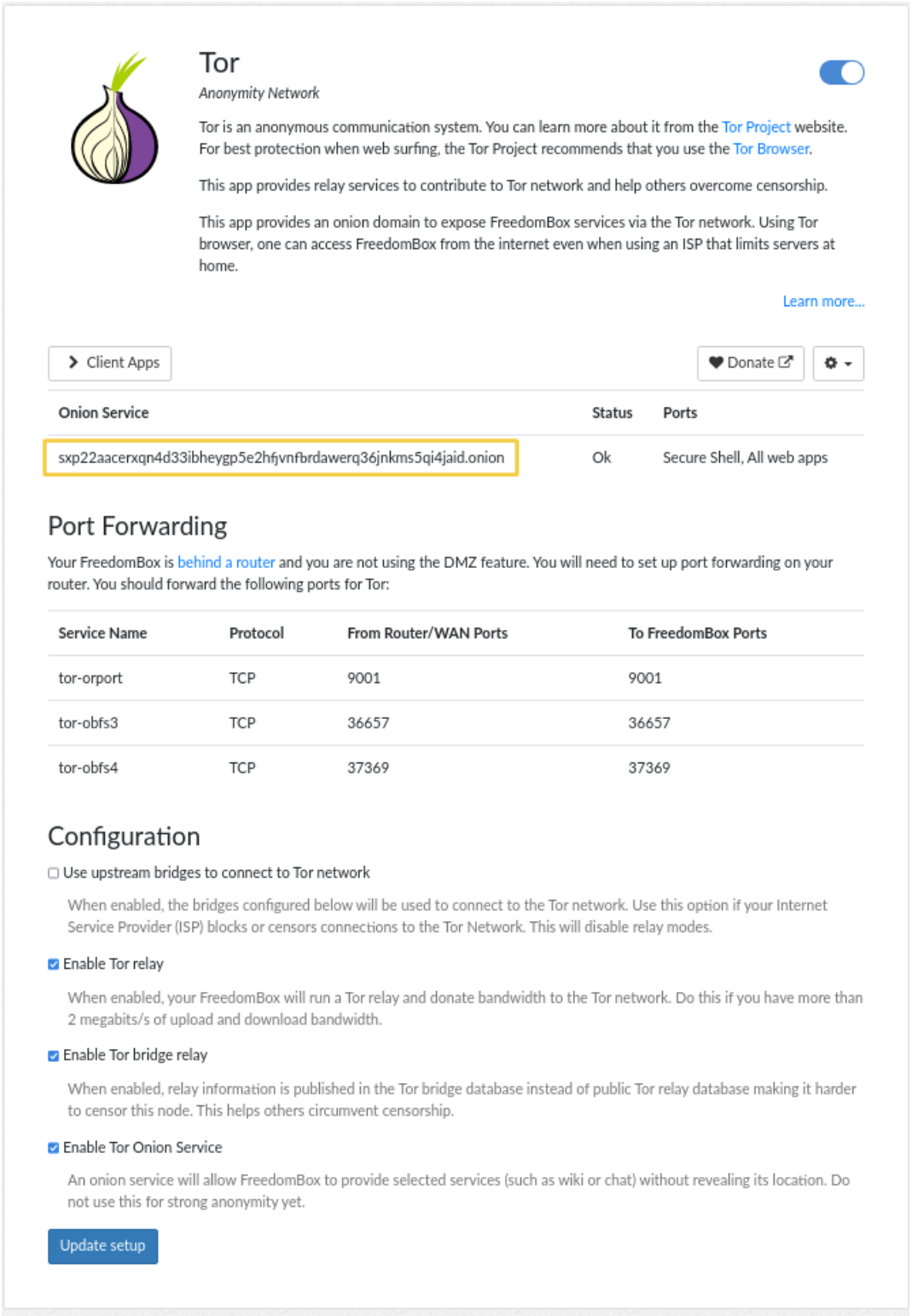
Tor Browser is the recommended way to browse the web using Tor. You can download the Tor Browser from `https://www.torproject.org/projects/torbrowser.html` and follow the instructions on that site to install and run it.

5.36.3 Using Tor Onion Service to access your FreedomBox

Tor Onion Service provides a way to access your FreedomBox, even if it's behind a router, firewall, or carrier-grade NAT (i.e., your Internet Service Provider does not provide a public IPv4 address for your router).

To enable Tor Onion Service, first navigate to the Anonymity Network (Tor) page. (If you don't see it, click on the FreedomBox logo at the top-left of the page, to go to the main Apps page.) On the Anonymity Network (Tor) page, under Configuration, check "Enable Tor Onion Service", then press the Update setup button. Tor will be reconfigured and restarted.

After a while, the page will refresh and under Status, you will see a table listing the Onion Service .onion address. Copy the entire address (ending in .onion) and paste it into the Tor Browser's address field, and you should be able to access your FreedomBox. (You may see a certificate warning because FreedomBox has a self-signed certificate.)



Currently only HTTP (port 80), HTTPS (port 443), and SSH (port 22) are accessible through the Tor Onion Service configured on the FreedomBox.

5.36.4 Apps accessible via Tor

The following apps can be accessed over Tor. Note that this list is not exhaustive.

- Calendar and Addressbook (Radicale)

- File Synchronization (Syncthing)
- Feed reader (TinyTinyRSS)
- Web Search (Searx)
- Wiki (MediaWiki)
- Wiki and Blog (Ikiwiki)

5.36.5 Running a Tor relay

When Tor is installed, it is configured by default to run as a bridge relay. The relay or bridge option can be disabled through the Tor configuration page in FreedomBox.

At the bottom of the Tor page in FreedomBox, there is a list of ports used by the Tor relay. If your FreedomBox is behind a router, you will need to configure port forwarding on your router so that these ports can be reached from the public Internet.

The requirements to run a relay are listed in the Tor Relay Guide. In short, it is

- recommended that a relay has at least 16 Mbit/s (Mbps) upload and download bandwidth available for Tor. More is better.
- required that a Tor relay be allowed to use a minimum of 100 GByte of outbound and of incoming traffic per month.
- recommended that a <40 Mbit/s non-exit relay should have at least 512 MB of RAM available; A relay faster than 40 Mbit/s should have at least 1 GB of RAM.

5.36.6 External links

- Upstream project: `https://www.torproject.org`
- Upstream documentation: `https://support.torproject.org/`

5.37 Tor Proxy (Anonymity Network)



Available since: version 23.14

5.37.1 What is Tor Proxy?

Tor is a network of servers operated by volunteers. It allows users of these servers to improve their privacy and security while surfing on the Internet.

The Tor Proxy app provides an option to download software packages over Tor for added anonymity. It also provides a web proxy on your FreedomBox for internal networks, which can be used by various apps to access the internet via the Tor network. In case of ISP censorship, upstream bridges can be configured to circumvented the censorship.

For using Tor Onion Service, or running a Tor relay, see the Tor app instead.

5.37.2 Using Tor Browser to browse anonymously

Tor Browser is the recommended way to browse the web using Tor. You can download the Tor Browser from `https://www.torproject.org/projects/torbrowser.html` and follow the instructions on that site to install and run it.

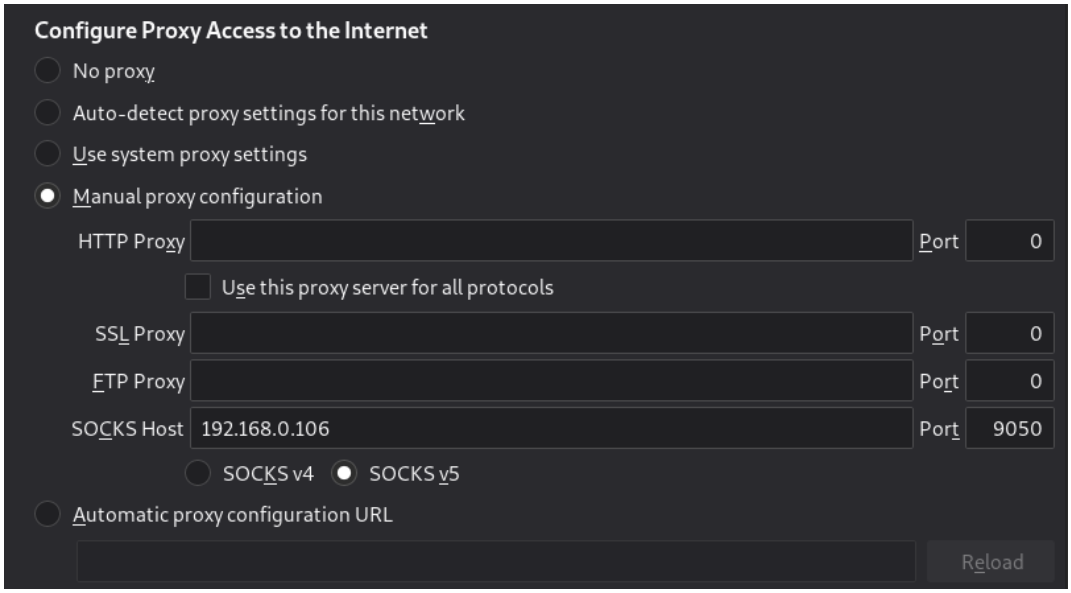
5.37.3 (Advanced) Using Tor Proxy as a SOCKS proxy

Tor Proxy provides a SOCKS port that other applications can connect to, in order to route their traffic over the Tor network. This port is accessible on any interfaces configured in the internal firewall zone. To configure the application, set SOCKS Host to the internal network connection's IP address, and set the SOCKS Port to 9050.

5.37.3.1 Example with Firefox

Your web browser can be configured to use the Tor network for all of your browsing activity. This allows for censorship circumvention and also hides your IP address from websites during regular browsing. For anonymity, using Tor Browser is recommended.

Configure your local FreedomBox IP address and port 9050 as a SOCKS v5 proxy in Firefox. There are extensions to allow for easily turning the proxy on and off.

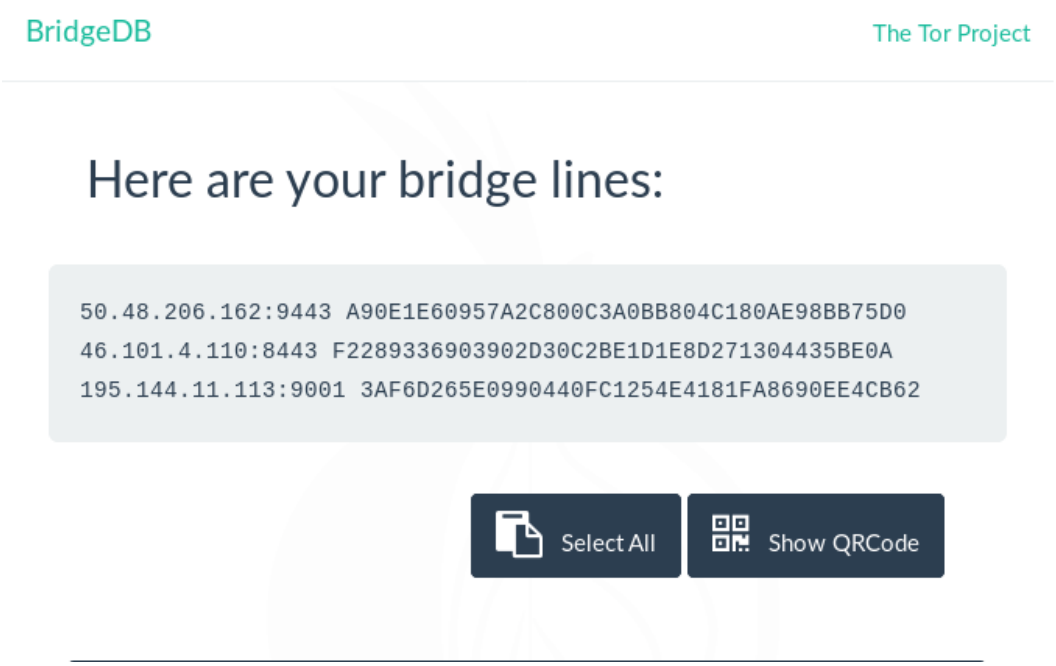


With the SOCKS proxy configured, you can now access any onion URL directly from Firefox. If you have also enabled the Tor app, then FreedomBox itself has an onion v3 address that you can connect to over the Tor network (bookmark this for use in emergency situations).

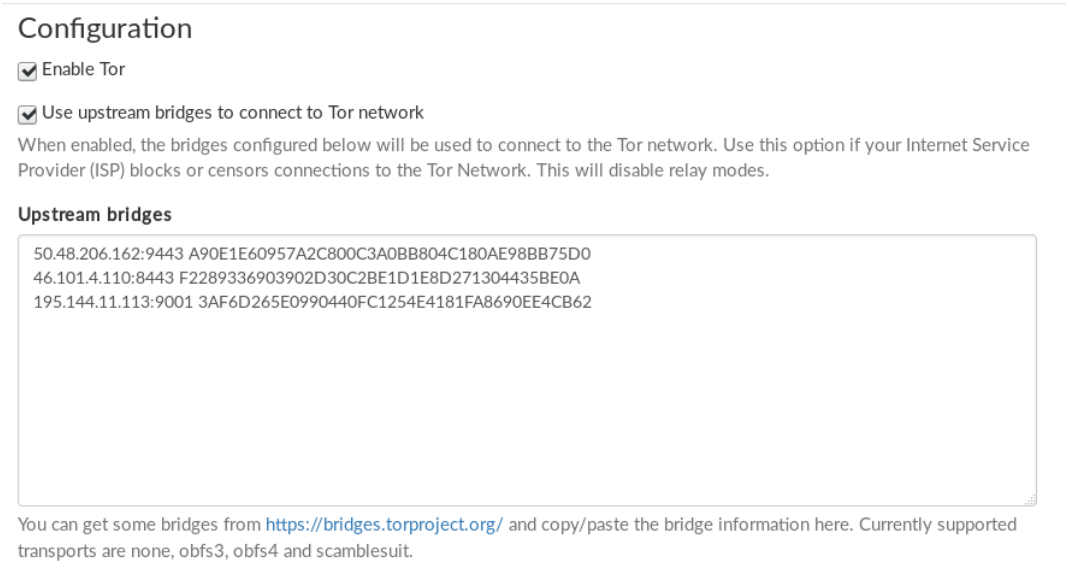
5.37.4 Circumventing Tor censorship

If your ISP is trying to block Tor traffic, you can use tor bridge relays to connect to the Tor network.

1. Get the bridge configuration from the Tor BridgeDB



2. Add the lines to your FreedomBox Tor Proxy configuration as show below.



5.37.5 External links

- Upstream project: https://www.torproject.org/
- Upstream end user documentation: https://support.torproject.org/

5.37.5.1 Client Apps

Links to recommended client apps, where applicable, can be found in FreedomBox by select the Apps page, selecting the application tile, and clicking the **> Client Apps** button.

Tor Proxy may be used by configuring a proxy in your web browser. Alternately, you can use the Tor Browser.

- Tor Browser download: https://www.torproject.org/download/

5.38 Transmission (Distributed File Sharing via BitTorrent)



Available since: version 0.5

5.38.1 What is Transmission ?

Transmission is a BitTorrent node (both, client and server at the same time).

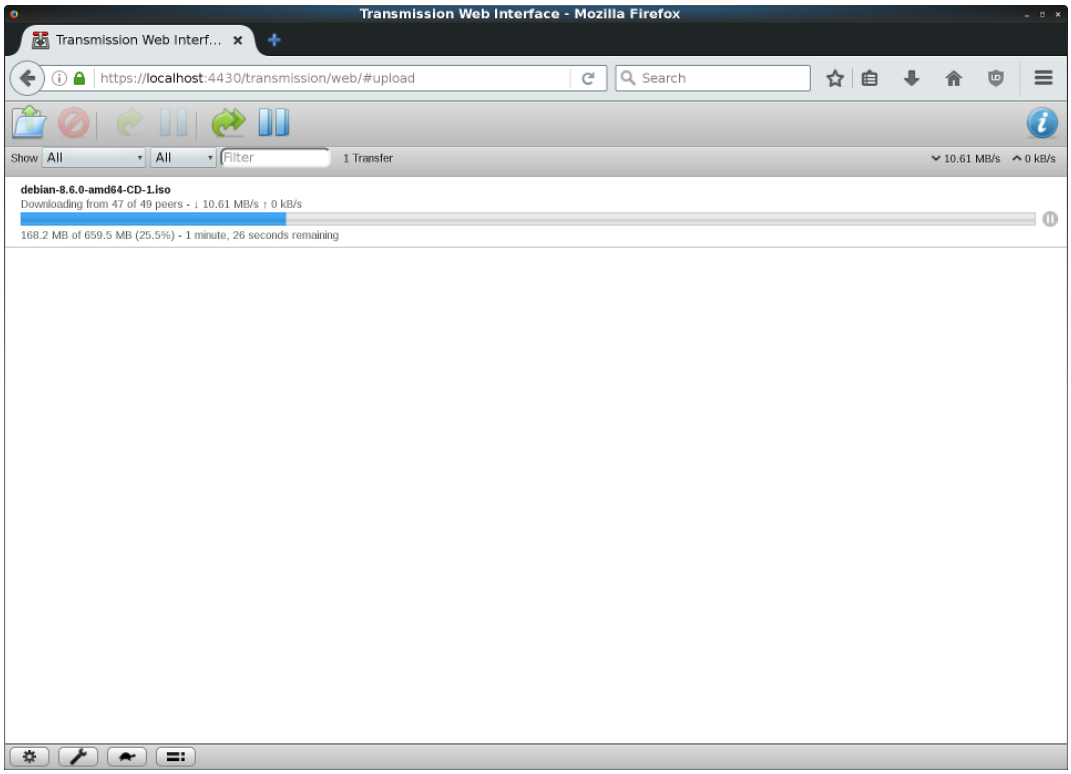
BitTorrent is a communications protocol for peer-to-peer (P2P) file sharing.

- It is **not anonymous**; you should assume that others can see what files you are sharing.
- This technology works best for big, popular files.

There are two BitTorrent web nodes available in FreedomBox: Transmission and Deluge. They have similar features, but you may prefer one over the other.

Transmission is a lightweight BitTorrent client that is well known for its simplicity and a default configuration that "Just Works".

5.38.2 Screenshot



5.38.3 Using Transmission

After installing Transmission, it can be accessed at `https://<your freedombox>/transmission`. Transmission uses single sign-on from FreedomBox, which means that if you are logged in on your FreedomBox, you can directly access Transmission without having to enter the credentials again. Otherwise, you will be prompted to login first and then redirected to the Transmission app.

5.38.4 Tips

5.38.4.1 Transferring Downloads from the FreedomBox

1. Transmission's downloads directory can be added as a shared folder in the Sharing app. You can then access your downloads from this shared folder using a web browser.
2. (Advanced) If you have the ssh access to your FreedomBox, you can use sftp or scp to browse the downloads directory using a suitable file manager or web browser:
   - Dolphin or Konqueror, if you access from a KDE desktop
   - The *Other locations* option in the default file manager, if you're on Gnome
   - WinSCP or FileZilla, if you're on Windows
   - Ghost Commander or Spider, if you're on Android.

5.38.5 Port Forwarding

If your FreedomBox is behind a router you optionally might want to set up port forwarding on your router in order to improve communication with other peers. You should forward the following ports for Transmission:

- TCP 51413 (or your configured *peer listening* port)

5.38.6 Using Remote Apps

In addition to using the web interface to control Transmission on FreedomBox, desktop and mobile apps may also be used. List of tested clients and their platforms are listed on the app page in FreedomBox web interface. When configuring these clients the URL to connect must be /transmission-remote/rpc and the port must be 443.

5.38.7 External Links

- Upstream Project: https://transmissionbt.com/
- Upstream Support Forum: https://forum.transmissionbt.com/
- Debian Transmission Wiki: https://wiki.debian.org/Transmission
- BitTorrent Introduction: https://www.bittorrent.org/introduction.html
- Wikipedia on BitTorrent: https://en.wikipedia.org/wiki/BitTorrent

5.39 User Websites

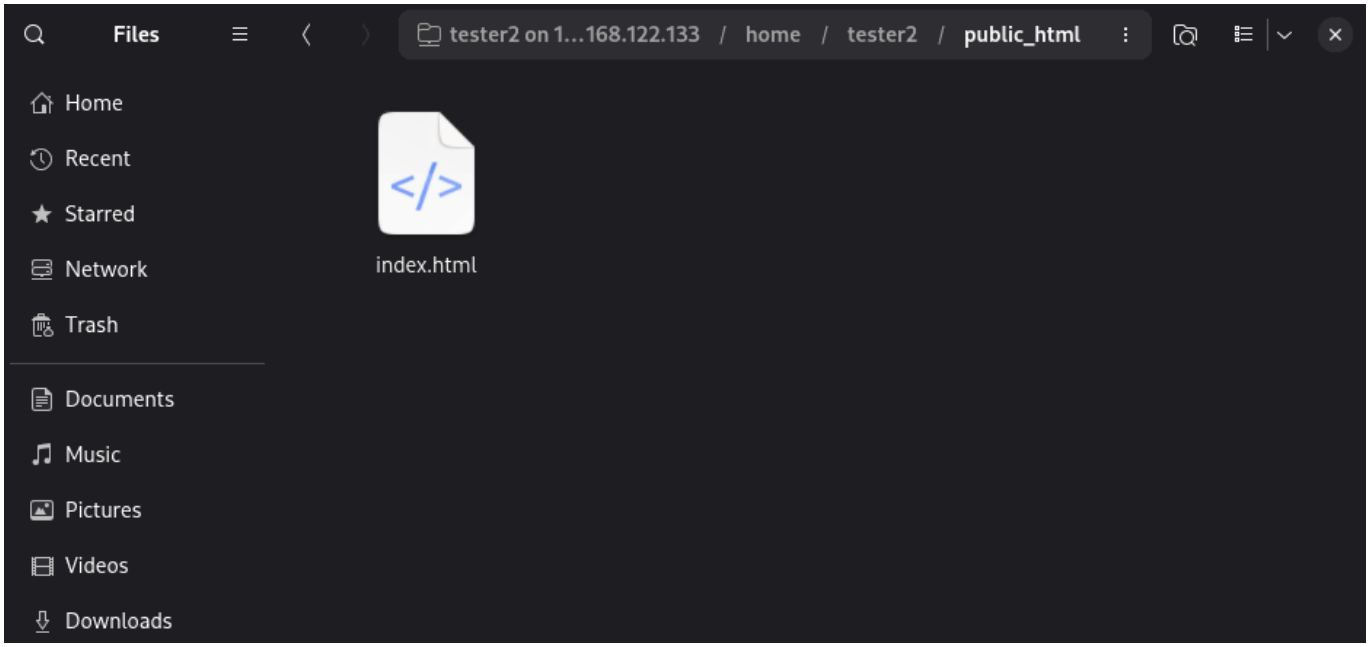
Available since: version 0.9.4

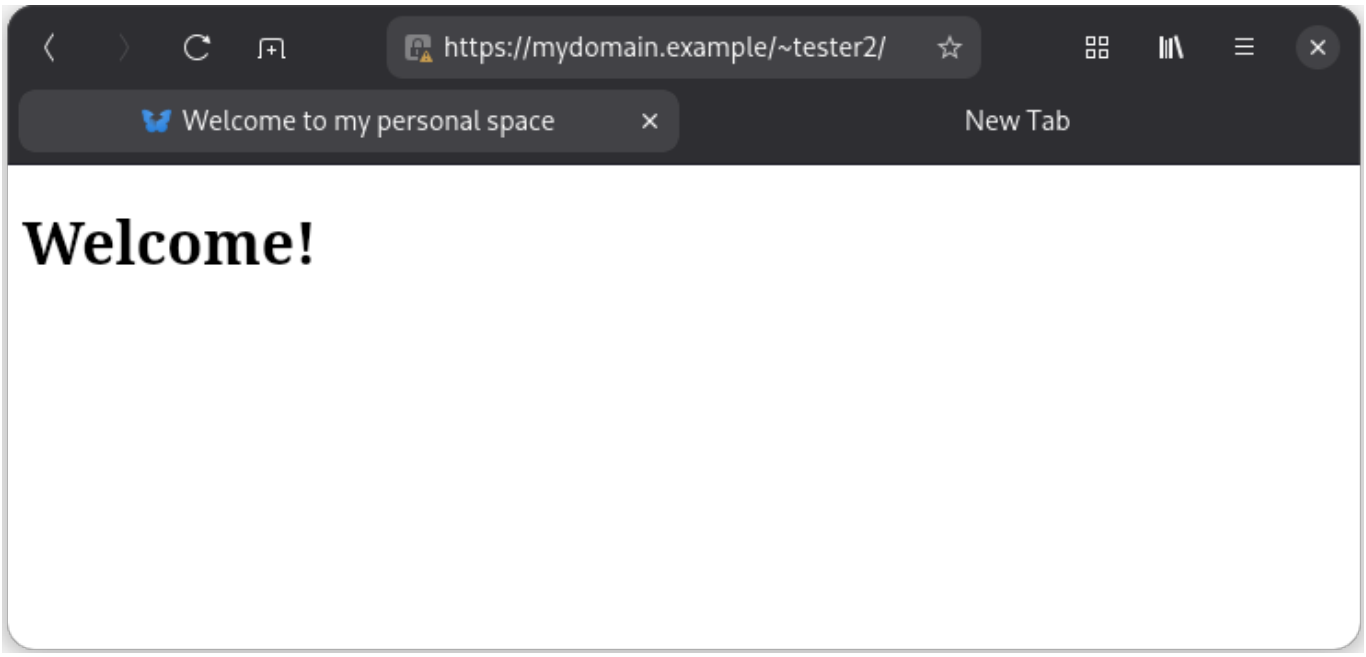
5.39.1 What is User Websites?

User websites is a feature that allows any (even non-admin) user on a FreedomBox to host their own website simply by copying files to well known location in their home directory on the FreedomBox server. The URL for the website will look like `https://mydomain.example/~myusername/`. The website will be available on the local network and/or the internet according to the network and firewall setup. If the copied files are HTML pages, they will show up as a website. If they are other types of files such as photos or documents. A list of those files is shown and a visitor will be able to view or download them.

Apache is the web server used in FreedomBox and this feature is implemented using an Apache module.

5.39.2 Screenshot





5.39.3 Using User Websites

The module is always enabled and offers no configuration from the FreedomBox web interface. There is no configuration or status page shown for this module in the FreedomBox web interface.

To serve documents, place the files in the designated directory in a FreedomBox user's home directory in the filesystem.

This directory is: **public_html**

Thus, the absolute path for the directory of a user named *fbx* with home directory in */home/fbx* will be **/home/fbx/public_html**. The User Websites feature will serve documents placed in this directory when requests for documents with the URI path *~fbx* are received. For example, if `mydomain.example` is your domain then a request for the URL `https://mydomain.example/~fbx` will display the file in `/home/fbx/public_html/photo.jpg`. If a file named *index.html* is placed in the directory, it will shown when no file name is provided in the URL. So, the URL `https://mydomain.example/~fbx/` will show the HTML page `/home/fbx/public_html/index.html`

5.39.4 Creating public_html folder and uploading documents

5.39.4.1 Visually from Linux

Linux standard desktop file managers use to support remote filesystem access through SFTP out of the box. Among others, Gnome's Nautilus, KDE/Plasma's Dolphin and XFCE's Thunar do so. This standarization allows for very easy, similar and straightforward procedures:

1. Connect with the file manager to your FreedomBox:
   - Gnome's Nautilus:
     1. To lauch Nautilus you can seek its archive icon, or search ether its name or the word "file".
     2. At the bottom of the left pane you'll find an option "+ Other locations".
     3. It leads you to a list of locations. Find "`freedombox SFTP server`" (english literal for all desktop languages). Click on it. If you don't find this, you can instead type `sftp://username@freedombox.local` in the address bar.
     4. The first time you'll be asked for your user and password. Enter your FreedomBox user and its password. The dialog will also offer you some options to remember it for some time.
   - Plasma file manager AKA Dolphin:

1. Click on the location bar at the top of the window.
2. Input `ftp://freedombox.local`
3. The first time you'll be asked for your user and password. Enter your FreedomBox user and its password. The dialog will also offer you some option to remember it.

- XFCE's Thunar:
  1. Type this into the browser bar: `sftp://username@freedombox.local`, replacing the 'username' placeholder with **your** actual FreedomBox username.
  2. I guess the first time you'll be asked for your password. Enter your FreedomBox user's password.

2. You should be shown FreedomBox filesystem. Enter the `home` folder and then enter you user's subfolder.
3. If there's no `public_html` folder, create it: right mouse button click, etc.
4. Drag your file(s) and drop it/'em into the `public_html` folder.
5. You should now be able to navigate your browser to the corresponding url and see the files.

5.39.4.2 Visually from Other Plattforms

If you want to use graphical free software clients, install:

- FileZilla or WinSCP for Windows.
- FileZilla for Mac.
- Spider or Ghost Commander, available in F-Droid application repository for Android.



Their usage will be similar to that described for Linux desktops.

5.39.4.3 With a Command Line Interface (CLI)

Usually any Unix system, including Linux in all (most) of its flavours and Mac, provide the standard utilities ssh, scp and sftp. FreeDOS provides SSH2DOS. No need to install anything. It's already there!

Examples:

Connect to FreedomBox via SSH:

1. (replacing `username` with a valid FreedomBox user name and `freedombox.local` with your FreedomBox's domain name or IP):

```
$ ssh username@freedombox.local
```

2. If your data is ok and your FreedomBox reachable, the first time you'll be asked to confirm its signature.
3. Then you'll be asked for the password of your FreedomBox user.
4. Then you'll be shown the welcome banner with the FreedomBox's buttefly logo in ASCII art (*painted* with characters).
5. The prompt changes to `username@freedombox:~$`.

Once connected create your website folder with:

```
username@freedombox:~$ mkdir ~/public_html
```

...or one for another user:

1. use the `sudo` prefix like

```
username@freedombox:~$ sudo mkdir /home/<the_other_user>/public_html
```

and introduce your password.

2. When you create a folder, by default it belongs to you no matter where it is created. Thus you'll then need to set its ownership to the other user:

```
username@freedombox:~$ sudo chown <the_other_user>:<the_other_user> /home/< ↩
    the_other_user>/public_html
```

3. Better check it before you disconnect that `public_html` is listed among the contents of the other user's home folder.

```
username@freedombox:~$ ls -l /home/<the_other_user>
...
drwxr-xr-x  2 <the_other_user> <the_other_user>   4096 jan 29 17:39  public_html
...
```

The name of the other user must appear **twice** in the `public_html` line and its permissions should be **`drwxr-xr-x`**.

Then any user can upload their files to their respective folders with any of the graphical clients. Ask them to check it.

It is a good security practice to exit instead of to just wait for the connection to time out:

```
username@freedombox:~$ exit
```

If then you want to also upload the web content through the command line you can

```
$ scp path/to/files username@freedombox.local:public_html/
```

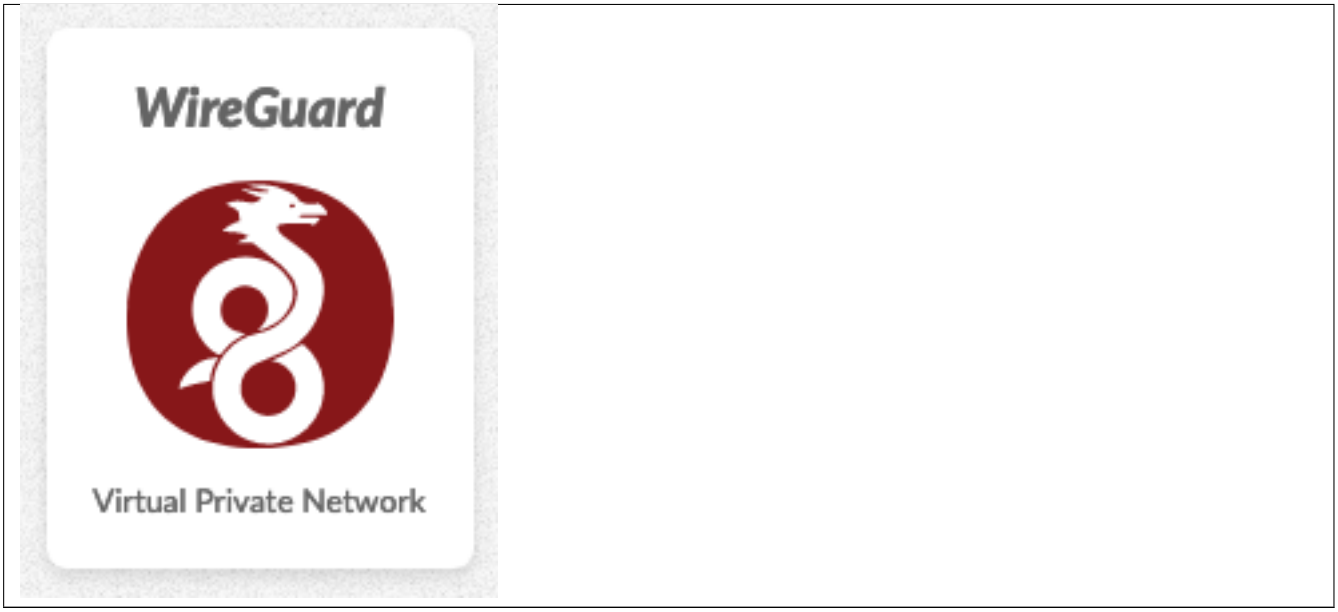
It will ask your password in FreedomBox. You should then be able to navigate your browser to the corresponding url and see the files.

Learn more about ssh, scp and sftp with `$ man ssh`, `$ man scp` and `$ man sftp`.

5.39.5 External Links

- Upstream Project: https://httpd.apache.org/
- Upstream Documentation:
  - https://httpd.apache.org/docs/2.4/mod/mod_userdir.html
  - https://httpd.apache.org/docs/2.4/howto/public_html.html
- Debian Apache Wiki: https://wiki.debian.org/Apache

5.40 WireGuard (Virtual Private Network)



5.40.1 About WireGuard

WireGuard is an extremely simple yet fast and modern VPN that utilizes state-of-the-art cryptography. It can be a useful replacement for IPSec or OpenVPN.

5.40.2 Installation

You can install wireguard from the Apps section of the FreedomBox web interface.

5.40.3 Usage

- Point-to-point tunnel
- VPN client with default route

5.40.4 Configuration - Debian Peers

Note: These steps are handled automatically on FreedomBox. So you only need to follow these steps on any Debian clients that will connect to FreedomBox, or Debian servers that FreedomBox will connect to.

- Step 1 - Generating Keypairs
- Step 2 - Alternative A - Manual Configuration

5.40.5 Configuration - Mobile Clients

WireGuard has a user space implementation for mobile devices available via the WireGuard app - available for Android and iOS (a full list of supported operating systems is available here).

The client can be configured in several ways:

5.40.5.1 Alternative A - Create configuration manually

This is self-explanatory, you actually create the config on the mobile device then transfer the relevant keys to the server's config.

5.40.5.2 Alternative B - Create configuration from archive

Here you have to create a .zip archive of the client configuration file, transfer it to the device then import it into the app.

5.40.5.3 Alternative C - Import by reading a QR code (most secure method)

The mobile client as of version 0.0.20180724 supports QR code based input.

DebianPackage:qrencode can be used to generate qr codes, even in a terminal/console using UTF8 characters.

The syntax is:

```
# qrencode -t ansiutf8 < client.conf
```

This will generate a QR code that is readable by the mobile client.

The advantage of this approach is that there is no need to transfer sensitive information via data channels that can potentially be compromised and there is no need for any additional software.

5.40.6 External Links

- Upstream Project: https://www.wireguard.com
- Upstream Documentation: https://www.wireguard.com/quickstart/
- Debian WireGuard Wiki: https://wiki.debian.org/WireGuard

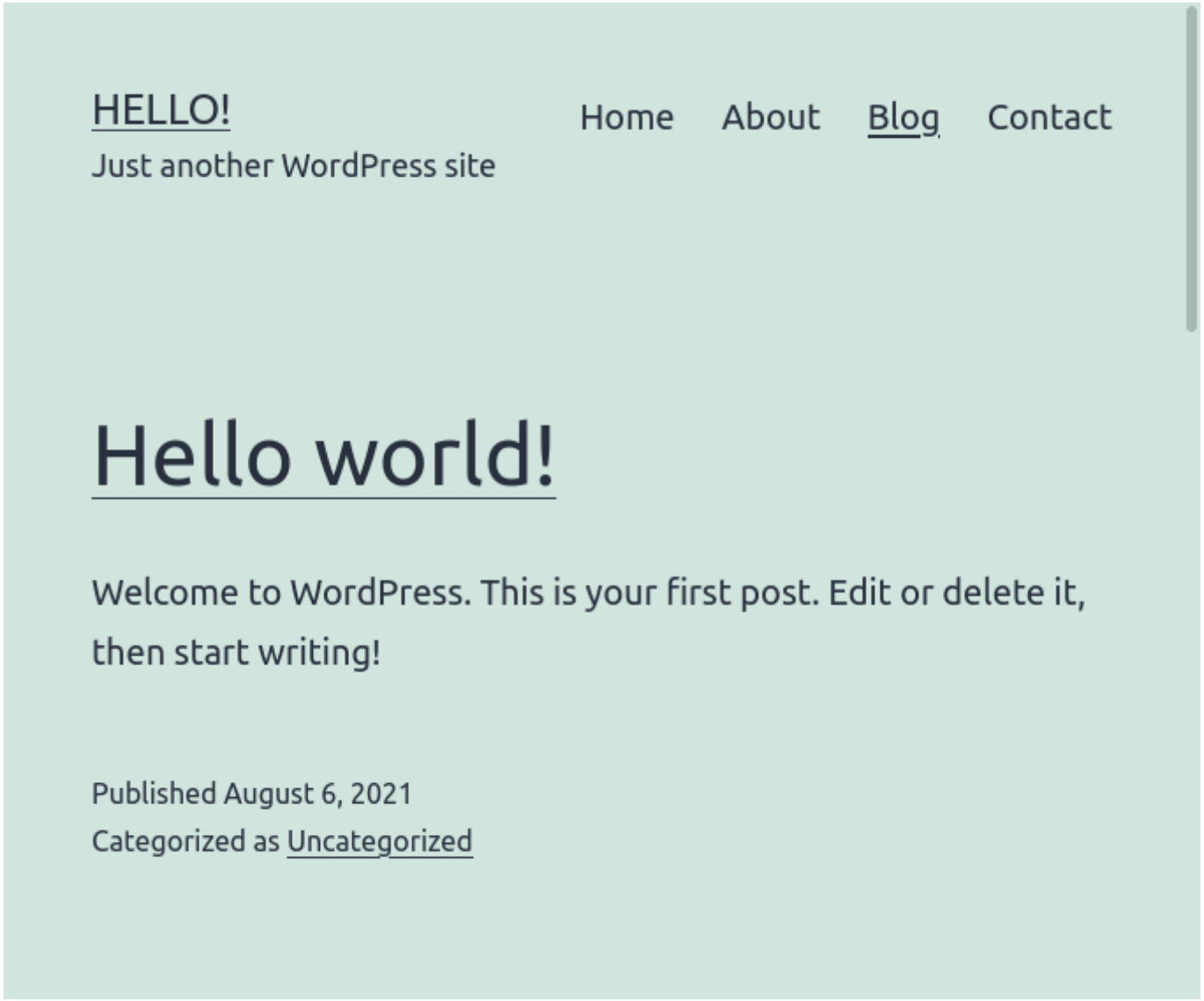
5.41 WordPress (Website and Blog)

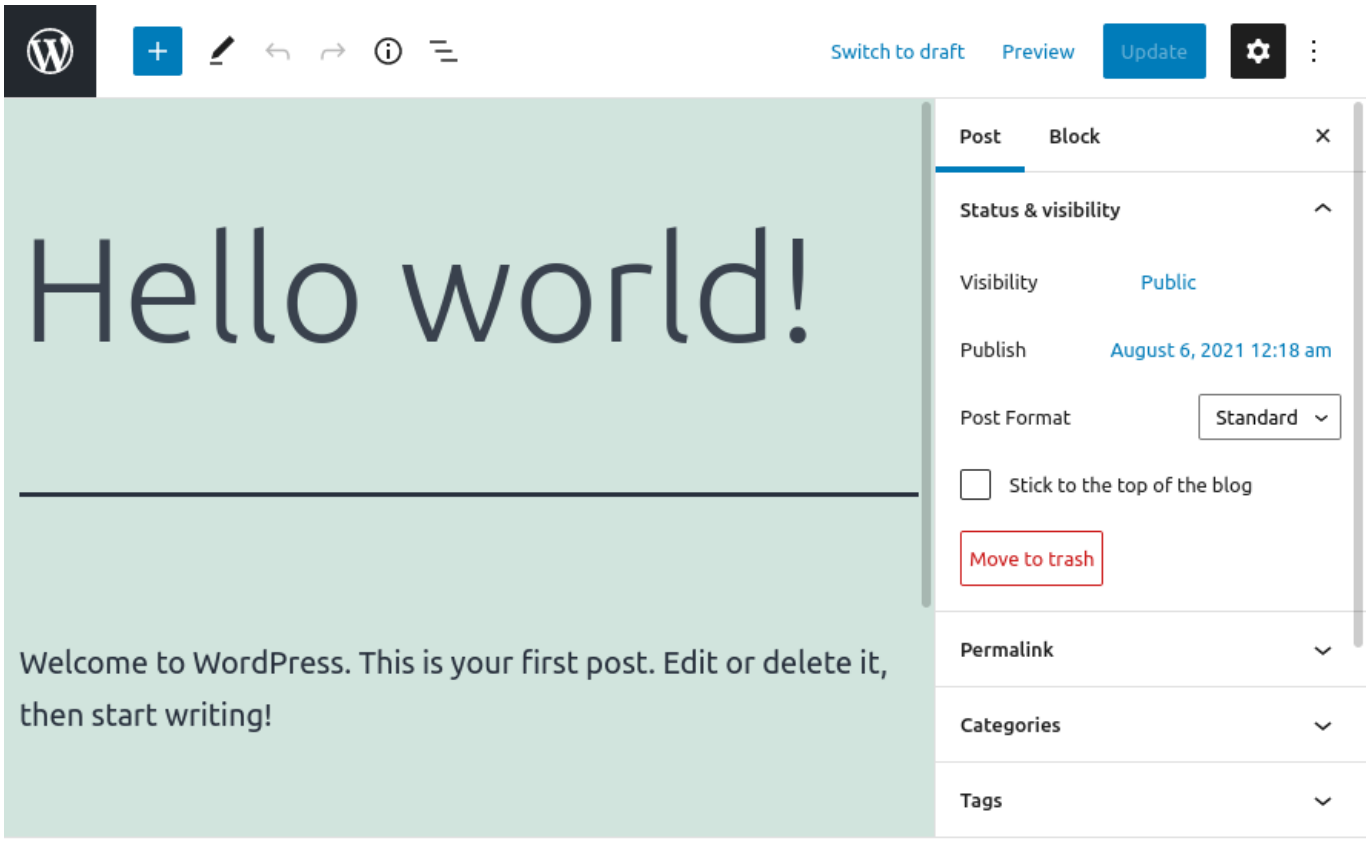


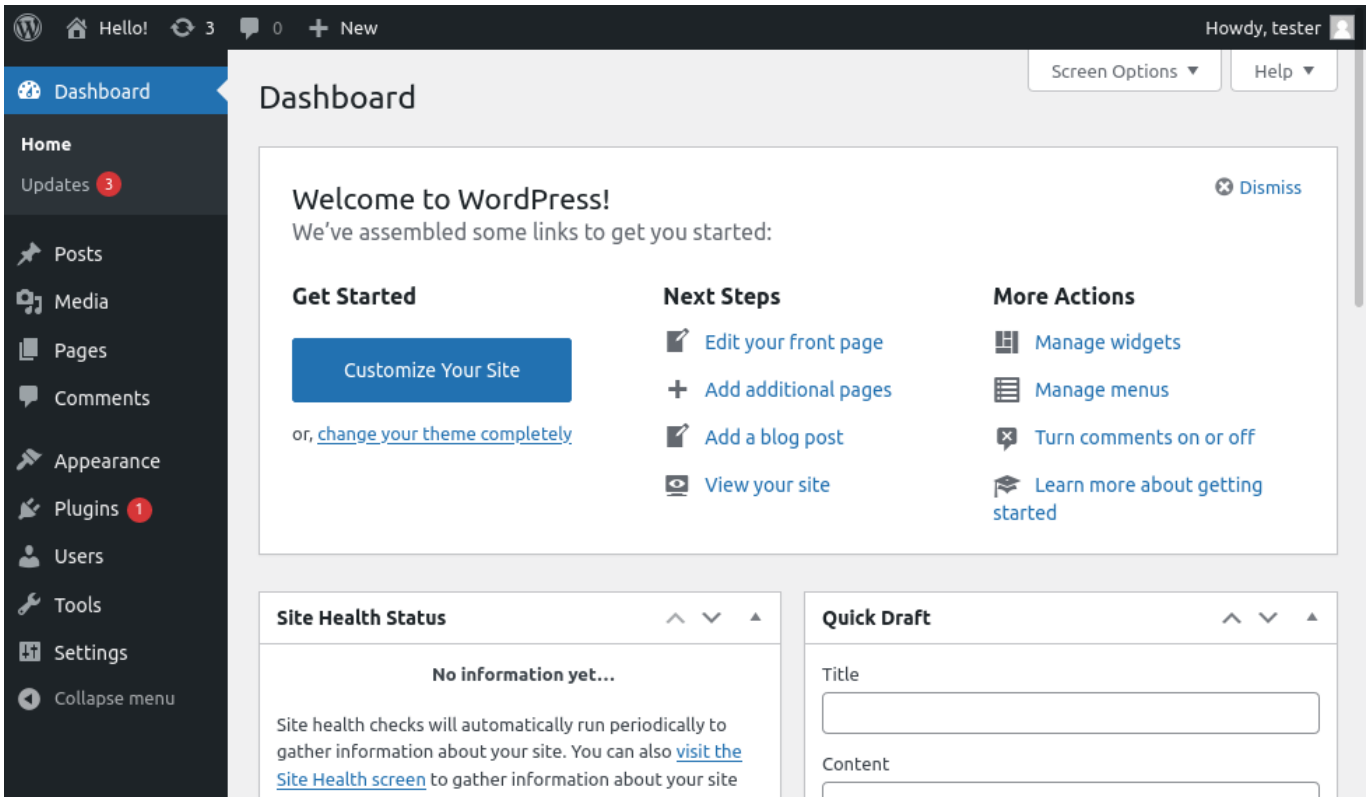
Available since: version 21.7

5.41.1 What is WordPress?

WordPress is a popular way to create and manage websites and blogs. It is a content management system that allows editing content using a visual web-based interface. No knowledge of HTML or other markup is needed to create websites. Complete layout and functionality of the web pages can be customized. Appearance can be chosen using themes. Administration interface and produced web pages are suitable for mobile devices.

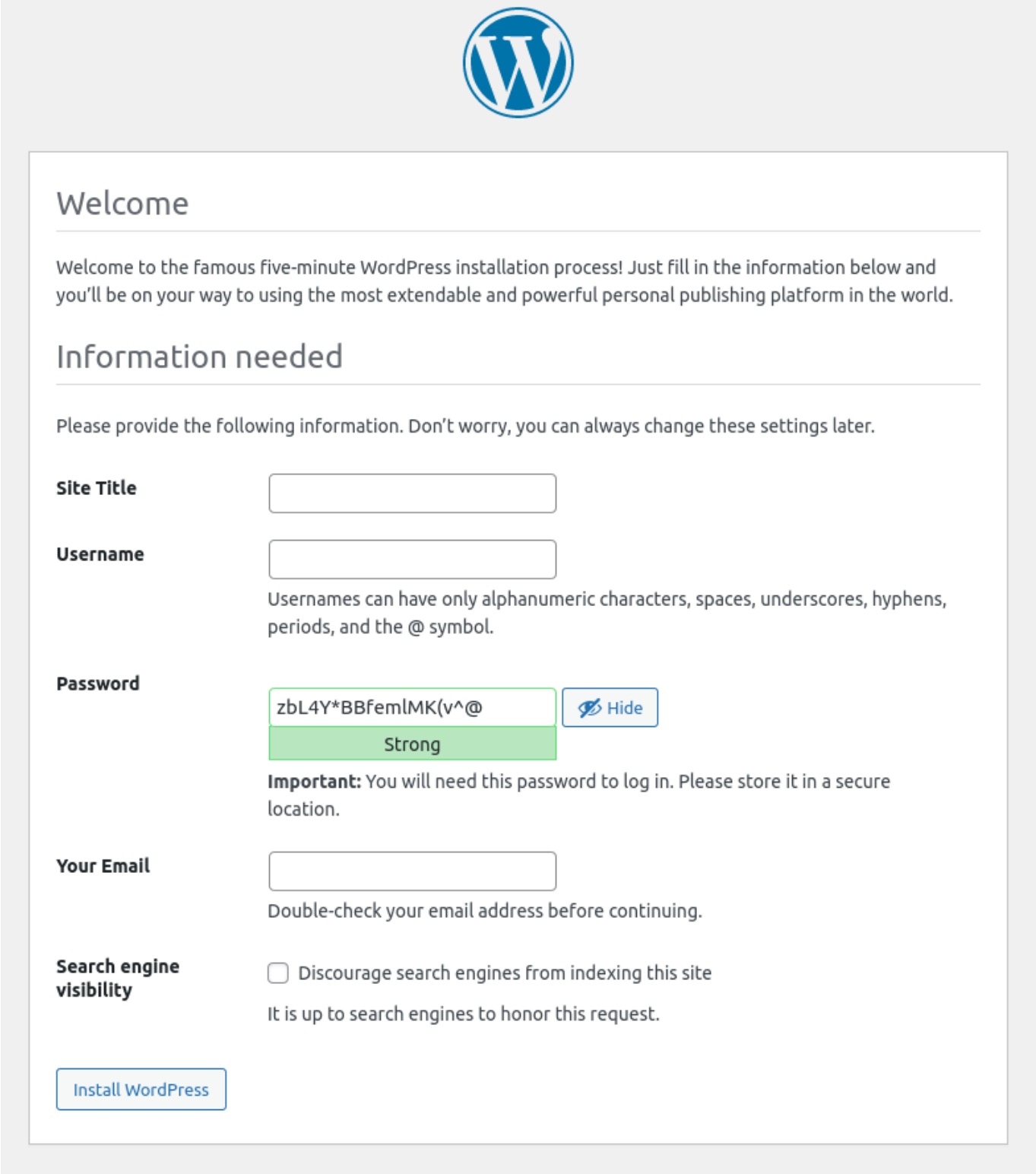



Switch to draft
Preview
Update
Post
Block
Hello world!
Status & visibility
Visibility
Public
Publish
August 6, 2021 12:18 am
Post Format
Standard
Stick to the top of the blog
Move to trash
Welcome to WordPress. This is your first post. Edit or delete it, then start writing!
Permalink
Categories
Tags
Document


Hello!
3
0
New
Howdy, tester
Dashboard
Home
Updates 3
Posts
Media
Pages
Comments
Appearance
Plugins 1
Users
Tools
Settings
Collapse menu
Dashboard
Screen Options
Help
Welcome to WordPress!
We've assembled some links to get you started:
Dismiss
Get Started
Customize Your Site
or, change your theme completely
Next Steps
Edit your front page
Add additional pages
Add a blog post
View your site
More Actions
Manage widgets
Manage menus
Turn comments on or off
Learn more about getting started
Site Health Status
No information yet...
Site health checks will automatically run periodically to gather information about your site. You can also visit the Site Health screen to gather information about your site
Quick Draft
Title
Content

5.41.2 Setting up WordPress on FreedomBox

To setup WordPress, navigate to the WordPress (Website and Blog) page and install it. WordPress needs a valid domain name to be configured. Before proceeding further, setup a proper domain name. Domains are configured using System -> Configure page. Access your FreedomBox web interface using the domain you have configured. After this, visit the WordPress web interface. This will show a setup page asking for the name of the site and details for a new administrator account. After this step, WordPress is fully configured and ready. You can then return to the FreedomBox's WordPress app page and optionally make the WordPress installation available to public.



5.41.3 Public Access

In the WordPress app page in FreedomBox, the app can be made publicly available. It is, by default, not publicly available. It is only available to users who login as administrators in FreedomBox interface. This is to protect the initial setup process from becoming publicly available. If the setup process is publicly available, any visitor will be able to create themselves an administrator account which is undesirable. Administrators must take care not to enable public access until WordPress' setup process has been completed.

5.41.4 Users

At the moment, WordPress has its own user accounts that are unrelated to FreedomBox accounts. The first account created during the setup process is an administrator account. After logging in with the administrator account, separate accounts for viewing, publishing, or administering may be created from within WordPress interface.

Changes to content and configuration of WordPress can only be done after logging into WordPress. However, by default, there is no link from the website or blog to reach the login page. Bookmark or directly type into the browser **https://<mydomainname>/wordpre admin/** to reach administration interface.

5.41.5 WordPress as Home Page

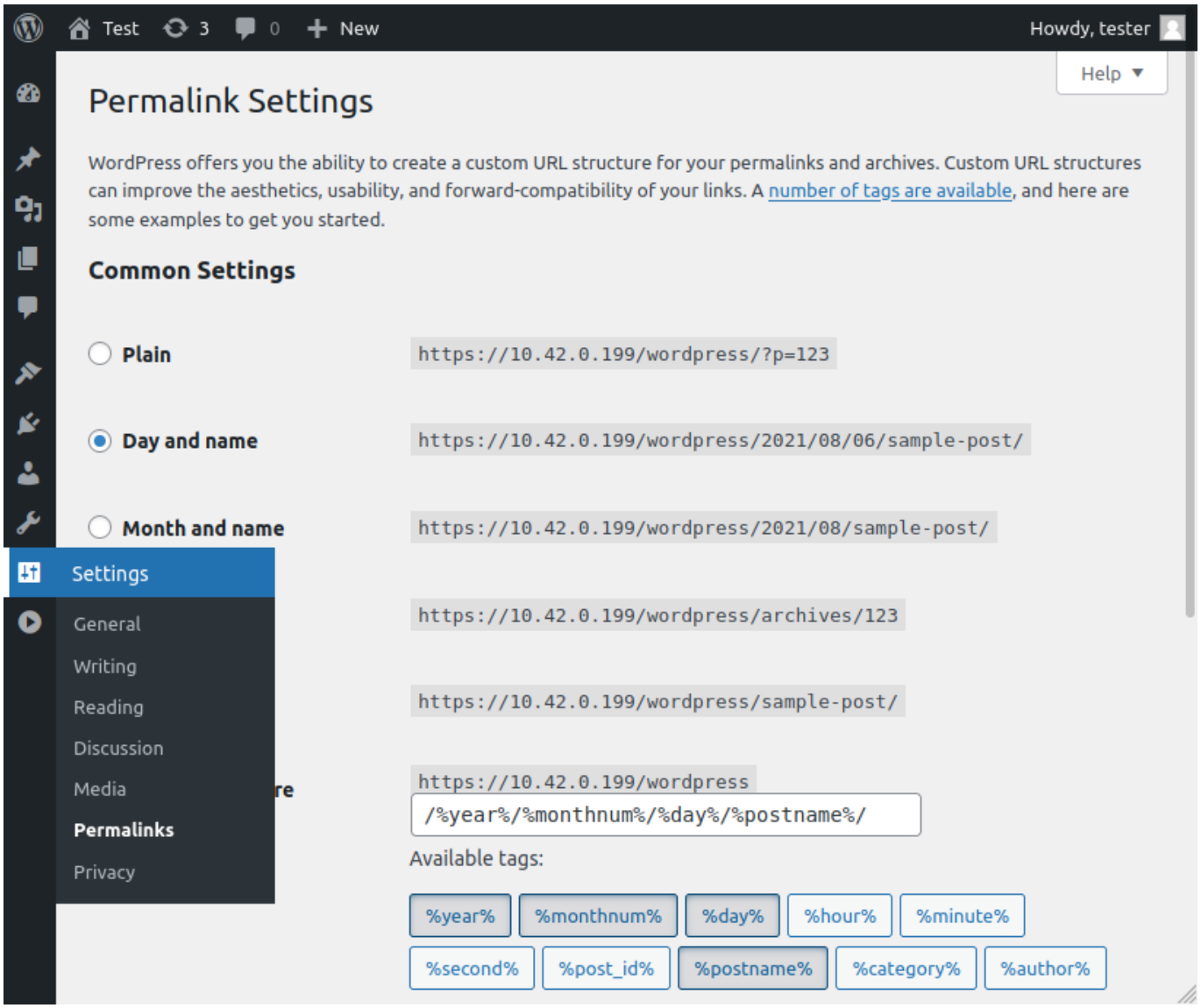
A beautiful, well customized WordPress website can be set as the home page for your FreedomBox. This can be done in System -> Configure page of the FreedomBox web interface. For example, if your FreedomBox's domain name is myfreedombox.rocks and you set WordPress as the home page, visiting `https://myfreedombox.rocks` will take you to `https://myfreedombox.rocks/wordpress/` instead of the FreedomBox interface.

5.41.6 Domain Name

When WordPress is setup for the first time, the domain name through which you access it is noted and WordPress gets configured with that domain name. Be sure to setup your domain name properly and access the WordPress setup process using the domain name and not a local IP address or domain name. Currently, FreedomBox does not provide an easy way to change the domain name once the app is installed.

5.41.7 Permalinks

By default, web addresses for newly created blog posts and pages look like **/wordpress/?p=1**. They can be made to look prettier like **/wordpress/2021/08/06/sample-post/** instead. This can be done from the *Settings* -> *Permalinks* configuration page in WordPress interface. The necessary web server configuration changes are handled by FreedomBox during app installation.



5.41.8 Automatic Upgrades

Similar to all other apps, feature and security upgrades for WordPress are automatically handled by FreedomBox (when not disabled). After a minor version upgrade, changes to the database structure are automatically done by WordPress. However, after a major version upgrade, such as during major distribution upgrade every two years, database changes are not done automatically. For this, you need to login to WordPress and trigger the changes manually.

5.41.9 Plugins and Themes

WordPress in itself is quite powerful and sufficient to create and manage a simple website or blog. It's true power, however, lies in the thousands of plugins and themes. Plugins extend the functionality of WordPress. For example, a contact form can be added to WordPress by installing the appropriate plugin. Themes change the appearance and layout of the site. Installing a new theme will provide an extra administration option for how your site will appear to your visitors. Care must be taken to choose trustworthy plugins and themes that respect software freedom and privacy of users and visitors of the site.

From FreedomBox version **22.13**, you can install plugins and themes directly from the WordPress GUI.

5.41.9.1 Google Fonts Privacy Issue

Please be aware that many of the third-party themes use Google Fonts which will violate your visitors' privacy.

One way to remove Google Fonts from your WordPress site is to remove the respective code lines from your theme's source code under **Appearance >> Theme Editor**. Please note these changes may be overwritten by the theme's next update.

5.41.9.2 Failing Updates

Manual update of the default theme and plugin, namely **Twenty Twenty-One** and **Akismet Anti-Spam** will fail, since these updates are managed separately by Debian. You do not have to worry about their updating.

5.41.9.3 Note for FreedomBox between version 21.7 and 22.12

Since there are few plugins/themes packaged for Debian, FreedomBox does not provide a simple way to install and manage them. You need to install them manually. This can be done as follows:

1. Note the URL of the plugin or theme to download by browsing them from WordPress administration interface or the official website. Be sure to select trustworthy ones with a free software license.
2. Log in via SSH using a FreedomBox administrator account.
3. Download the plugin or theme and unpack into a directory using the command line.
4. Move the directory under **/var/lib/wordpress/wp-content/plugins/** or **/var/lib/wordpress/wp-content/themes/** as appropriate.
5. Watch for upgrades to these plugins from WordPress and repeat the process for installing newer versions.

5.41.10 External links

- Upstream Project: https://wordpress.org
- Upstream Documentation: https://wordpress.org/documentation/
  - WordPress Customization: https://wordpress.org/documentation/customization/
  - Explore WordPress themes: https://wordpress.org/themes/
- Debian WordPress Wiki: https://wiki.debian.org/WordPress

5.42 Zoph (Photo manager)



Available since: version 21.3 (and Debian 11, Bullseye)

5.42.1 What is Zoph?

Zoph is a web based photo manager, allowing uploads of photos to the FreedomBox server, where they can be organised into Albums, and associated with Locations, People and Categories. An individual photo can be in multiple albums, and Albums, Categories and Locations are hierarchical.

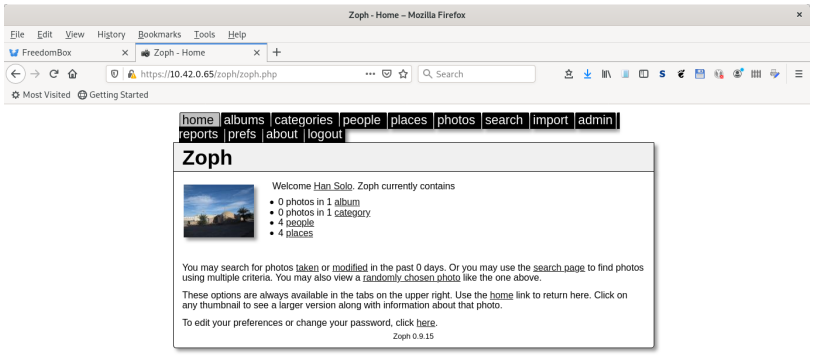
Zoph supports multiple users, and has a permissions system to control which Albums users can see, or create, whether they can see or create People etc.

For FreedomBox the username within Zoph must match the FreedomBox username so Single Sign On will work.

5.42.2 Using Zoph

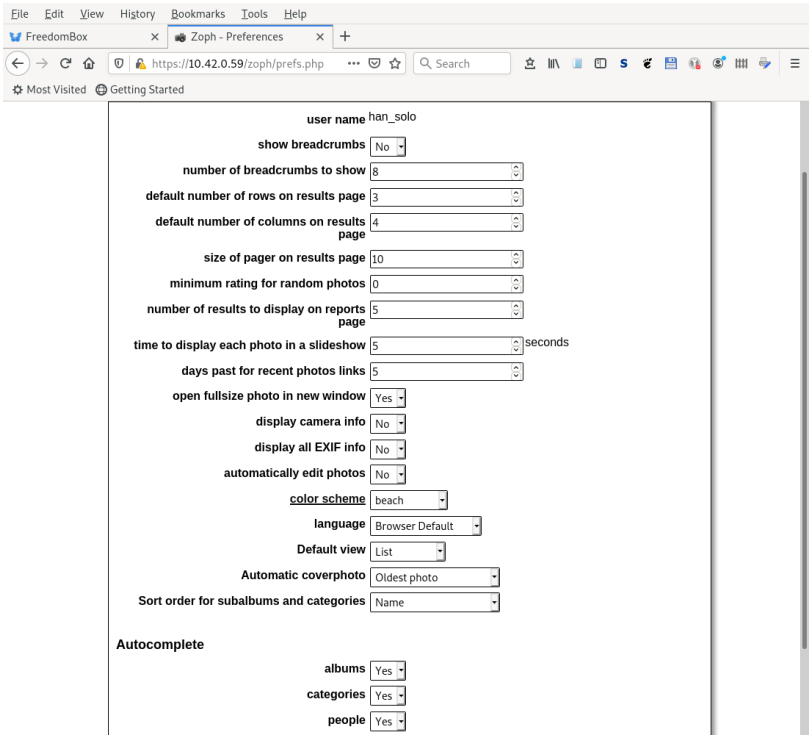
After Zoph is installed, you'll need to click "Setup". Then you can launch the web client. It can also be accessed at `https://<your freedombox>/zoph`.

Only the very first time you'll be asked for user and password. The next times you'll be taken straight to a welcome screen.

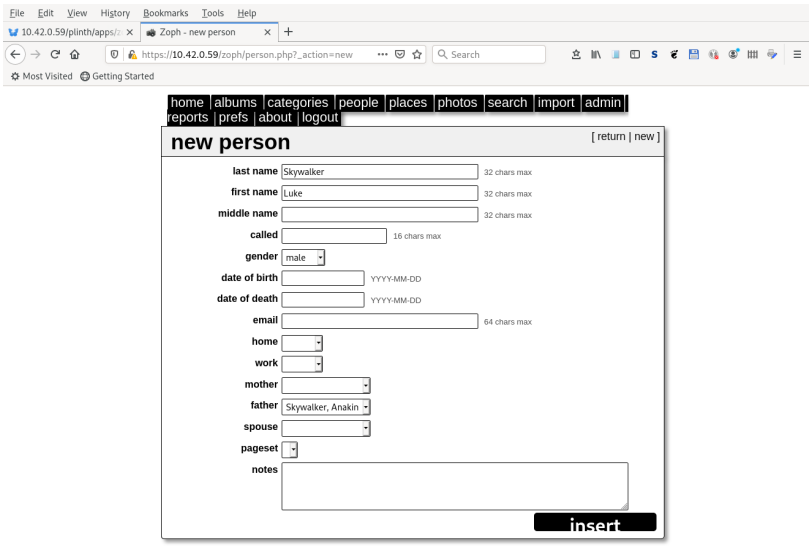


The tab menu will be shown on top of every page. From there you can import photos from any computer, administer Zoph to add other users etc.

You can now go to the 'prefs' tab and set your preferences, for numbers of rows and columns in results displays, how much information you wish displayed about the camera used to take the photo and so on.



You can add information about People who are in your Photos.



5.42.2.1 Choosing a storage location for your photos

Your photos will need a lot of storage space compared to the other uses of your FreedomBox. You may want to put them onto an external disk. You can (not yet) specify in the initial install screen where your photos should be stored. The database which holds information about albums, people etc is held in your normal FreedomBox storage.

5.42.3 External links

- Upstream Project: http://www.zoph.org
- Upstream documentation: https://gitlab.com/zoph-project/zoph/tree/master/docs

6 System

6.1 Backups

FreedomBox includes the ability to backup and restore data, preferences, configuration and secrets from most of the applications. The Backups feature is built using Borg backup software. Borg is a deduplicating and compressing backup program. It is designed for efficient and secure backups. This backups feature can be used to selectively backup and restore data on an app-by-app basis. Backed up data can be stored on the FreedomBox machine itself or on a remote server. Any remote server providing SSH access can be used as a backup storage repository for FreedomBox backups. Data stored remotely may be encrypted and in such cases remote server cannot access your decrypted data.

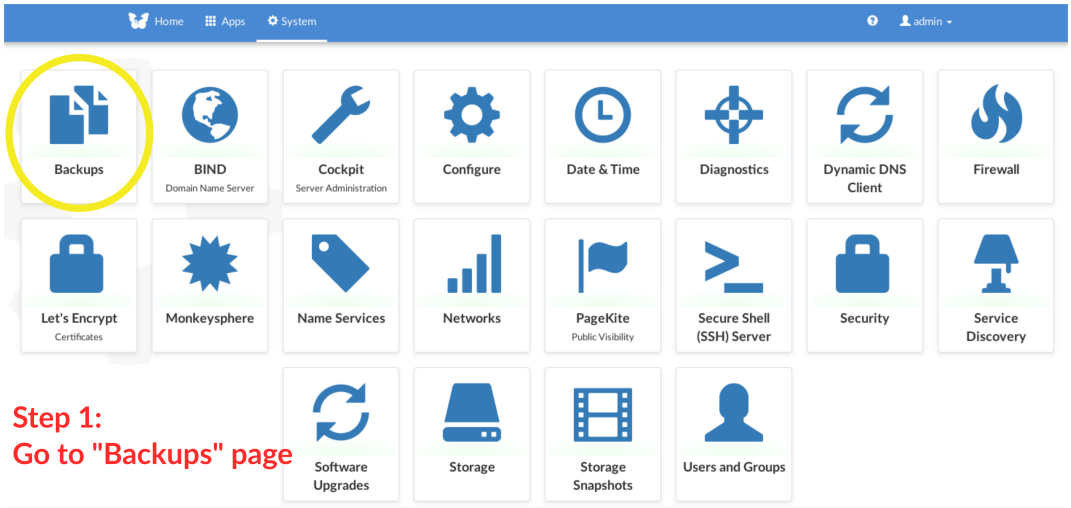
6.1.1 Notes for Specific App Backups

Unless otherwise noted here, backup of an app's data will include its configuration, secrets and other data.

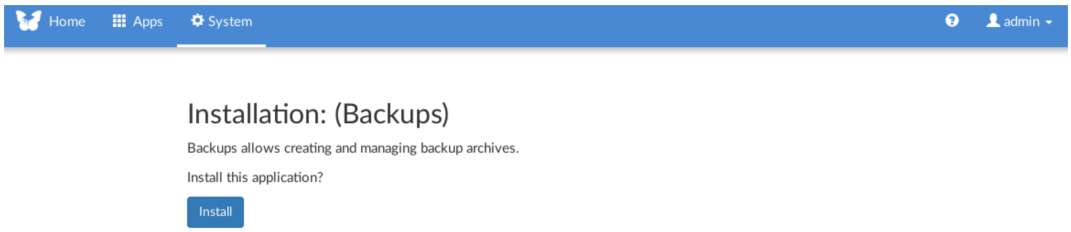
App/Feature	Notes
Deluge	Does not include downloaded/seeding files
MiniDLNA	Does not include the data in the shared folders
Networks	No plans currently to implement backup
Samba	Does not include the data in the shared folders
Sharing	Does not include the data in the shared folders
Snapshot	Only configuration, does not include snapshot data
Syncthing	Does not include data in the shared folders
Transmission	Does not include downloaded/seeding files
Users	Backup of user accounts is planned

6.1.2 How to install and use Backups

Step 1



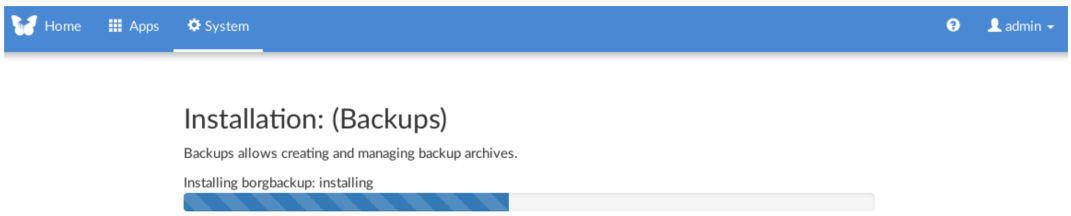
Step 2



Step 2:
Click "Install"

JavaScript license information

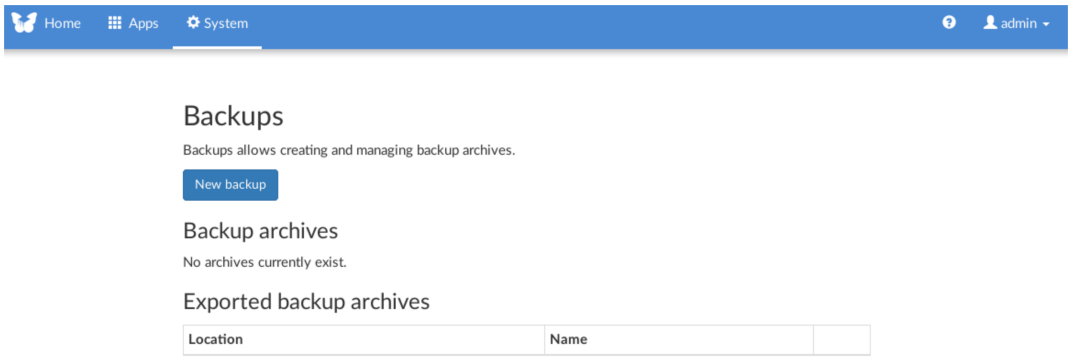
Step 3



Step 3:
Wait for Backups to install

JavaScript license information

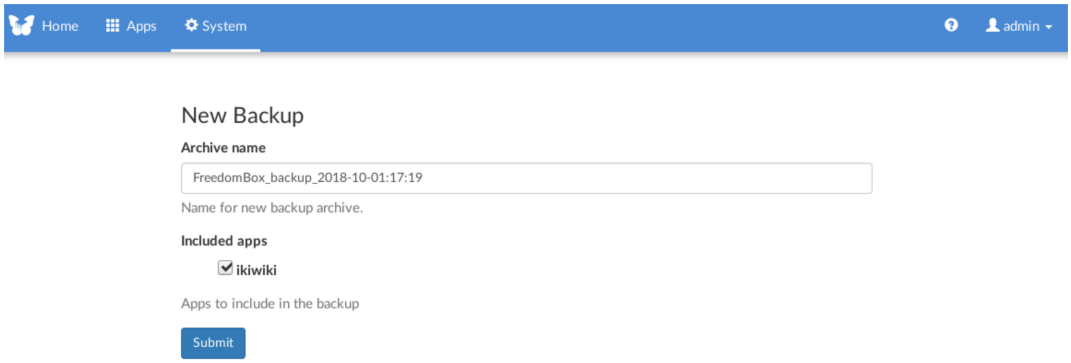
Step 4



Step 4:
Click "New backup" to create the backup archive

JavaScript license information

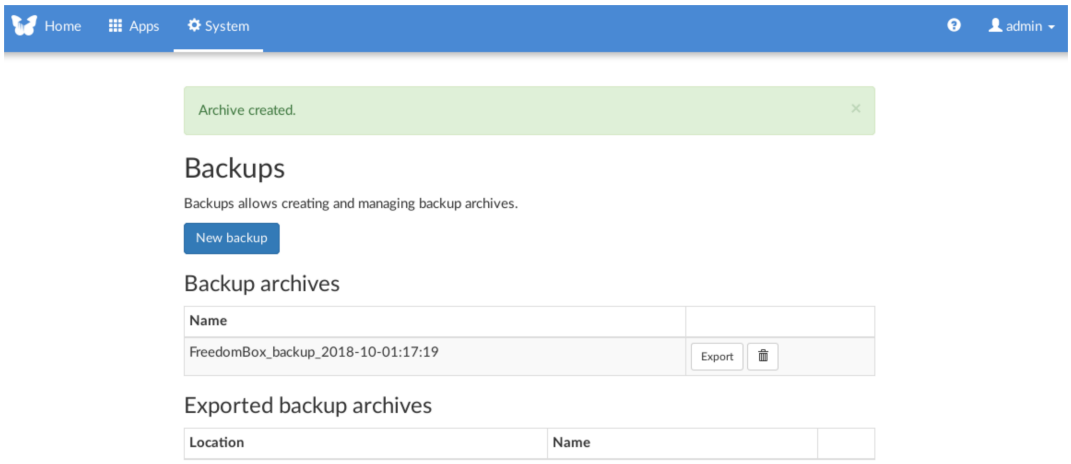
Step 5



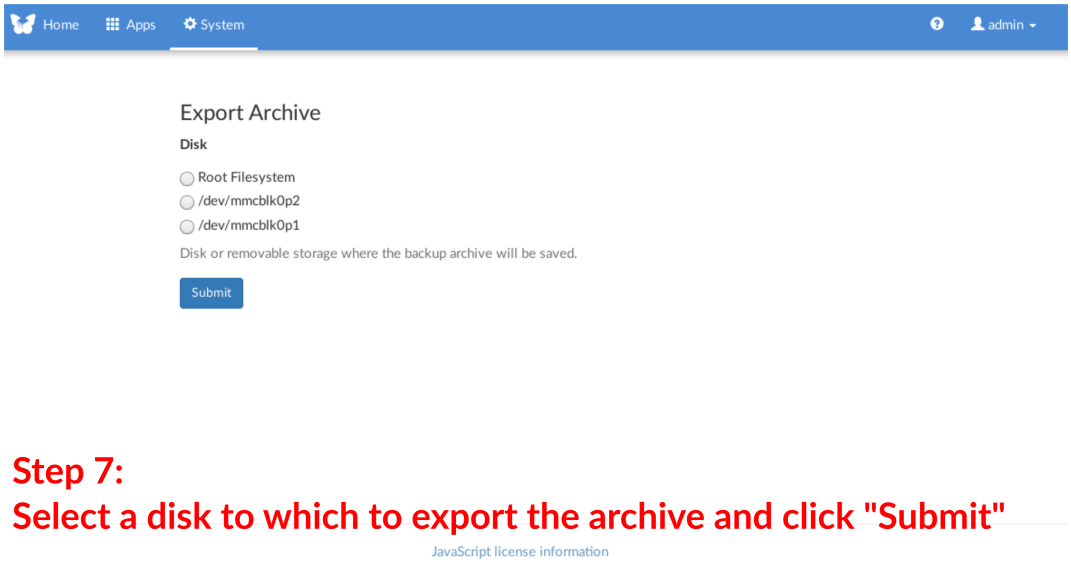
Step 5:
Type a name for the archive, select apps to include, click submit

JavaScript license information

Step 6



Step 7



6.1.3 External links

- Upstream project: https://www.borgbackup.org
- User documentation: https://borgbackup.readthedocs.io/en/stable/

6.2 BIND (Domain Name Server)

BIND enables you to publish your Domain Name System (DNS) information on the Internet, and to resolve DNS queries for your user devices on your network.

Currently, on FreedomBox, BIND is only used to resolve DNS queries for other machines on local network. It is also incompatible with sharing Internet connection from FreedomBox.

Note: This service is available only on networks configured as "internal" zone. It is not available when connected via OpenVPN.

6.2.1 Using BIND

When BIND is enabled, that does not automatically mean that anything is using it. The following can be configured:

- FreedomBox can be configured to use the local BIND service for its own DNS lookups.
- Clients on the Local Area Network can be configured to use the FreedomBox's BIND service for their DNS lookups.

The FreedomBox can be set to use its own BIND service for DNS lookups through Networks:

1. Go to System page, and then select Networks.
2. Select the "FreedomBox WAN" connection and press Edit.
3. Under "IPv4 Addressing Method", there is a field "DNS Server". Set it to `127.0.0.1`. /* TODO: Add IPv6 instructions. */
4. Press "Edit Connection" at the bottom to save the changes.


Restart the FreedomBox from the user drop-down menu.

6.2.2 External links

- Upstream project: https://www.isc.org/bind/

6.3 Cockpit (Server Administration)

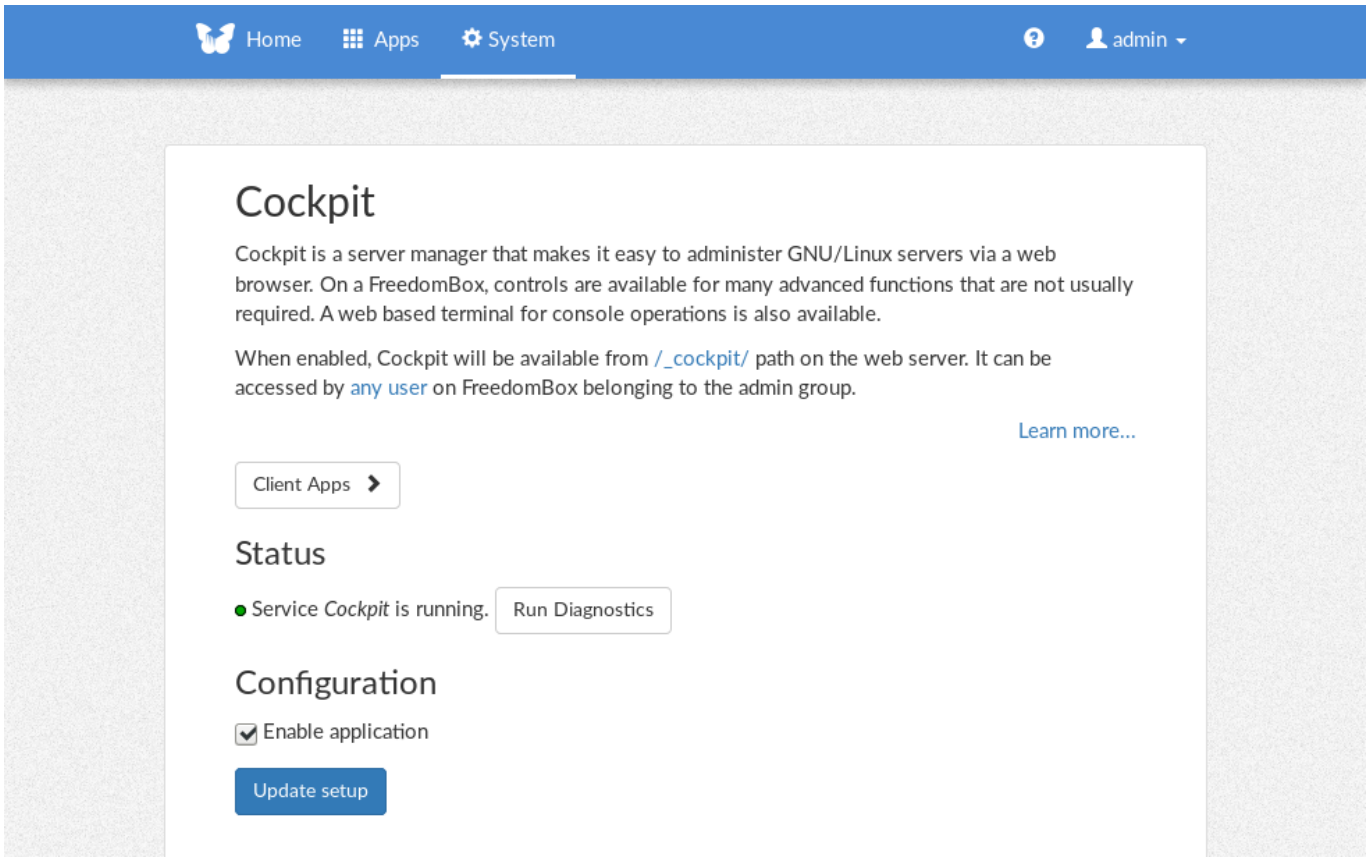
Cockpit is a server manager that makes it easy to administer GNU/Linux servers via a web browser. On a FreedomBox, controls are available for many advanced functions that are not usually required. A web based terminal for console operations is also available.

It can be accessed by any user on your FreedomBox belonging to the admin group. Cockpit is only usable when you have proper domain name setup for your FreedomBox and you use that domain name to access Cockpit. See the Troubleshooting section for more information.

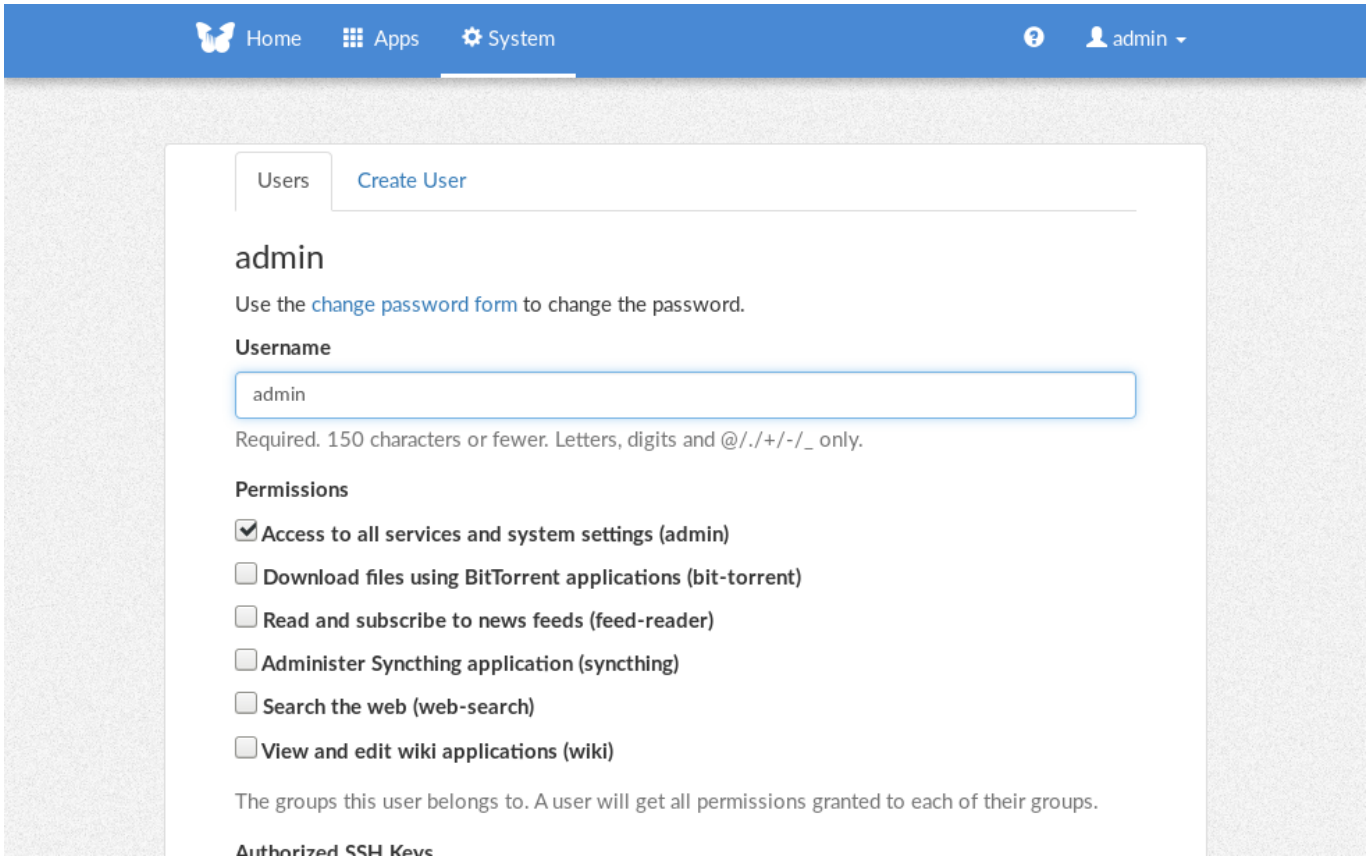

Caution
Use cockpit only if you are an administrator of GNU/Linux systems with advanced skills. FreedomBox tries to coexist with changes to system by system administrators and system administration tools like Cockpit. However, improper changes to the system might causes failures in FreedomBox functions.

6.3.1 Using Cockpit

Install Cockpit like any other application on FreedomBox. Make sure that Cockpit is enabled after that.

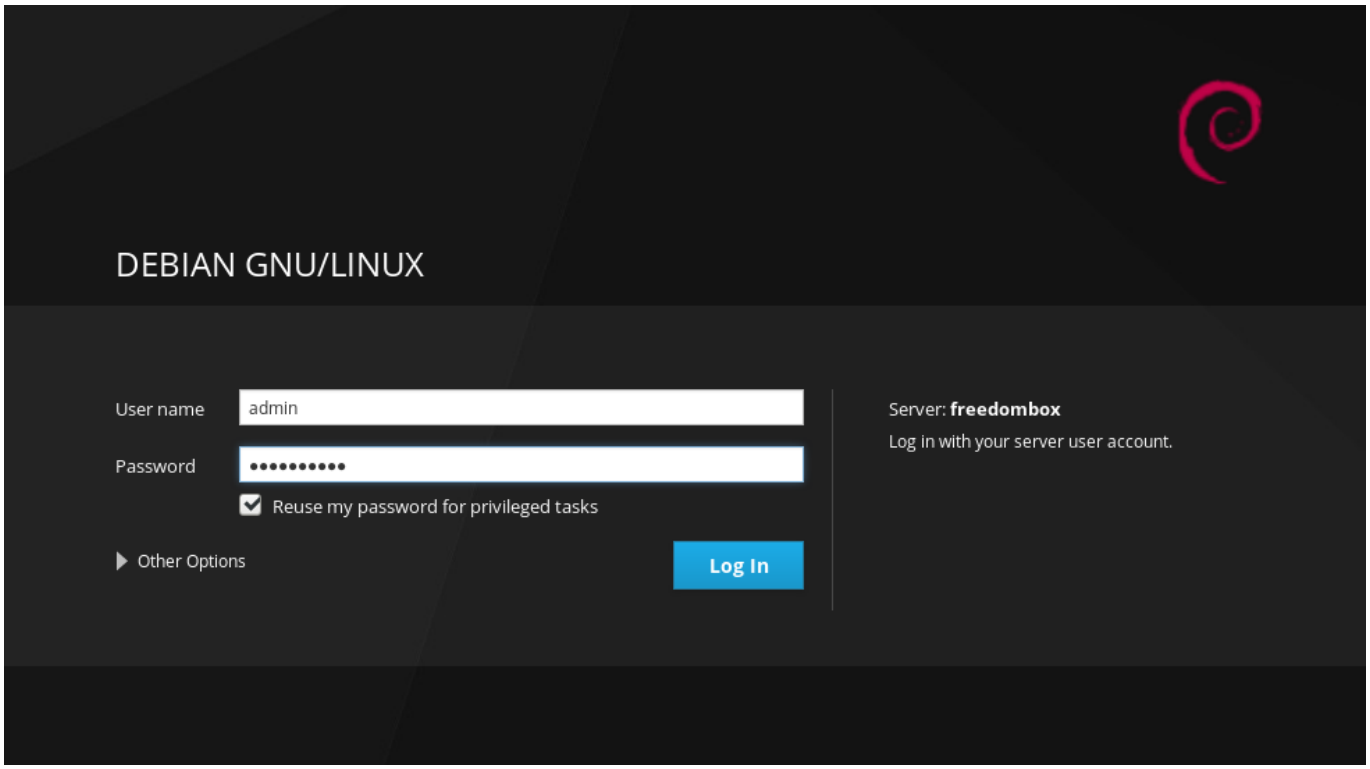


Ensure that the user account on FreedomBox that will used for Cockpit is part of the administrators group.

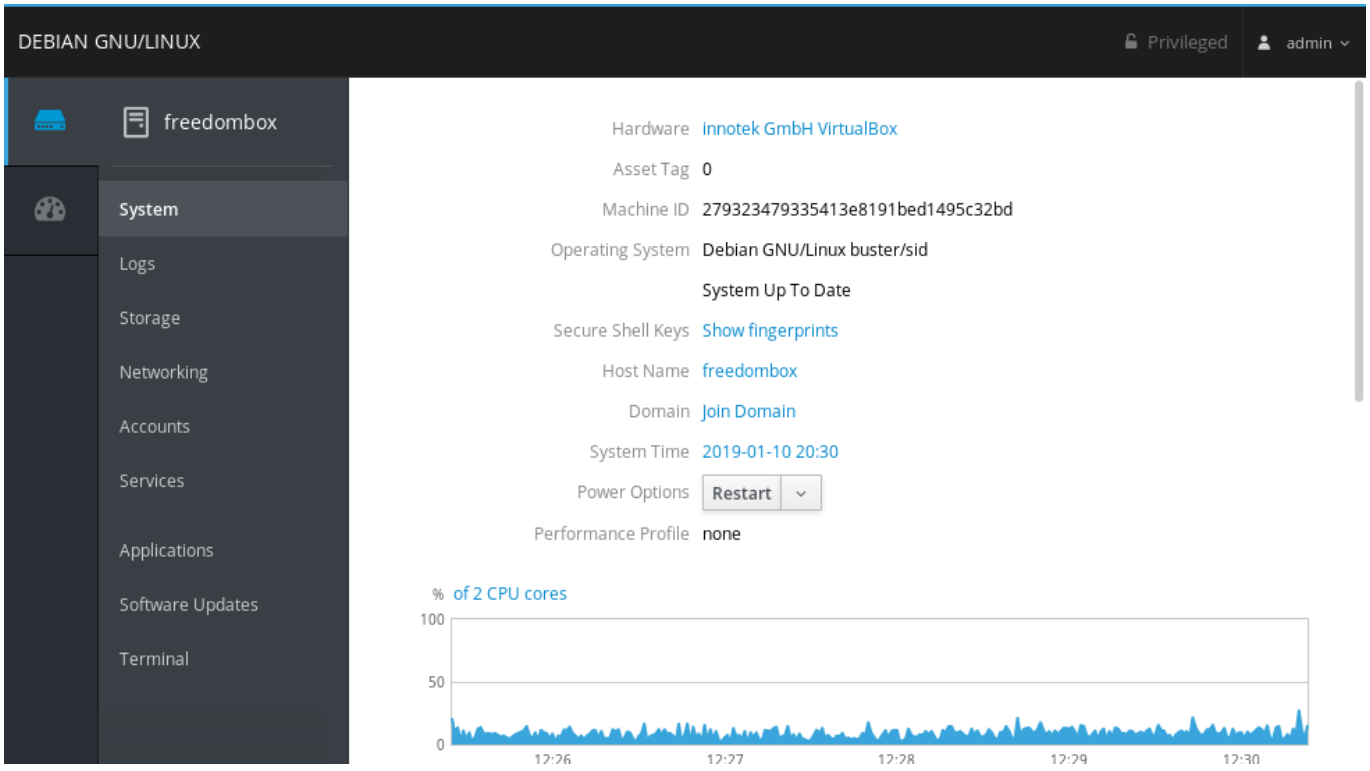


Launch the Cockpit web interface. Login using the configured user account. Be sure to check the box to "reuse my password for

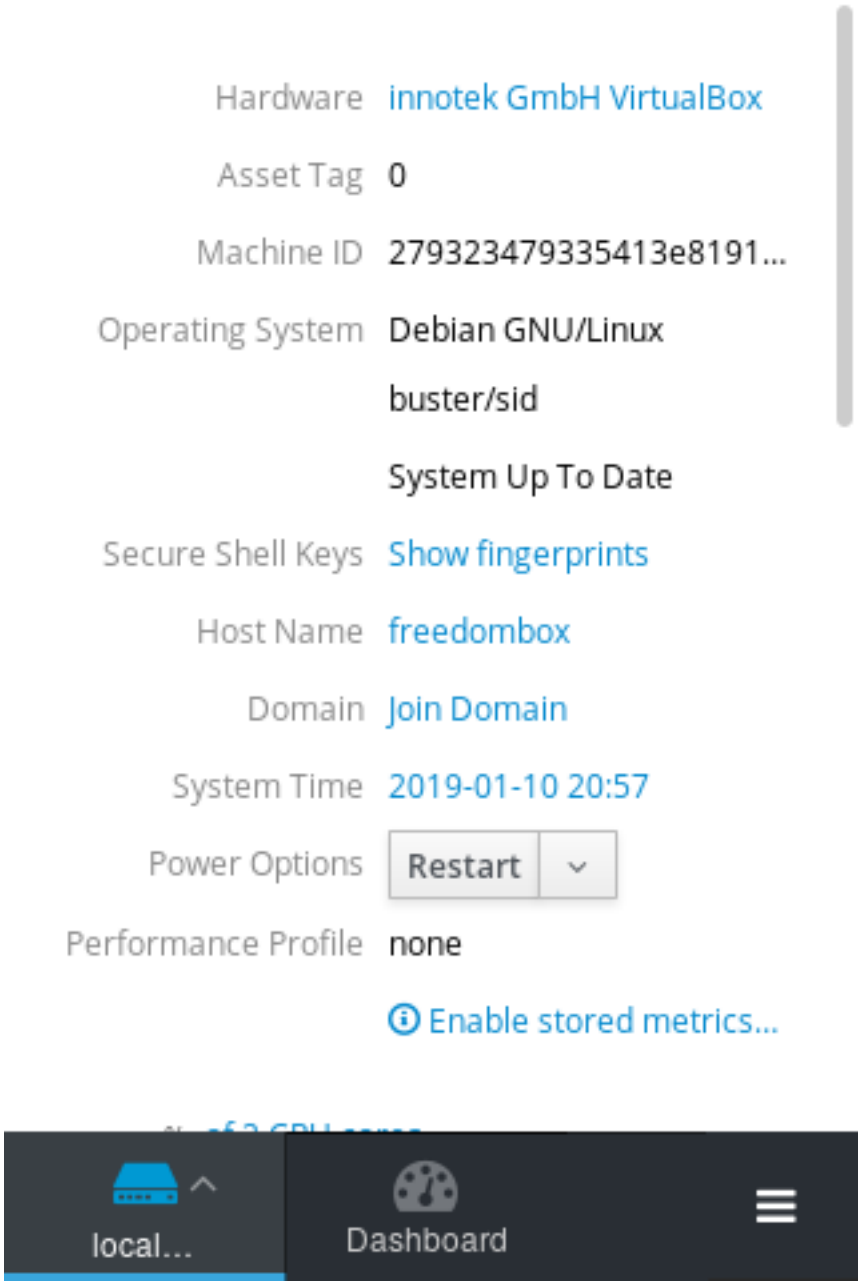
privileged tasks", otherwise you will not be able to perform various tasks such as configuring raid, or editing users, once logged in.



Start using cockpit.



Cockpit is usable on mobile interfaces too.



6.3.2 Features

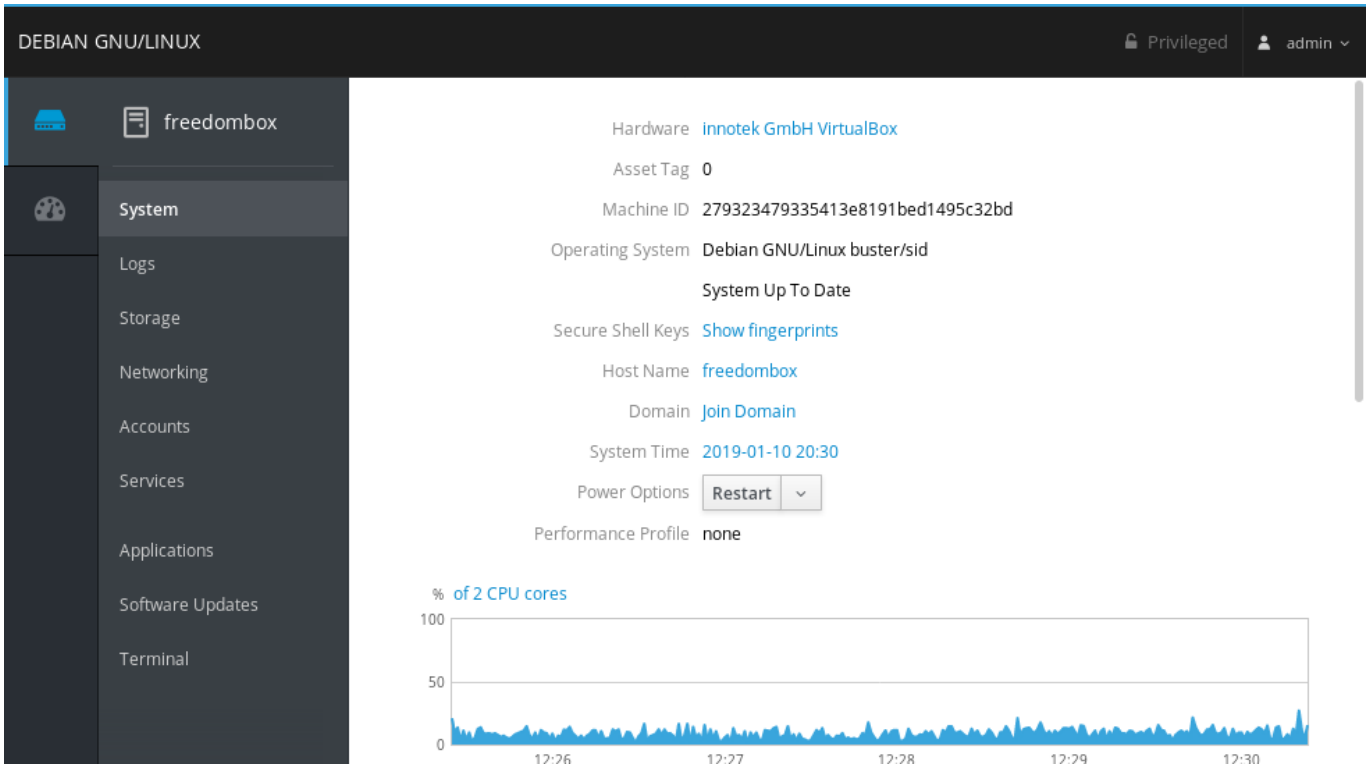
The following features of Cockpit may be useful for advanced FreedomBox users.

6.3.2.1 System Dashboard

Cockpit has a system dashboard that

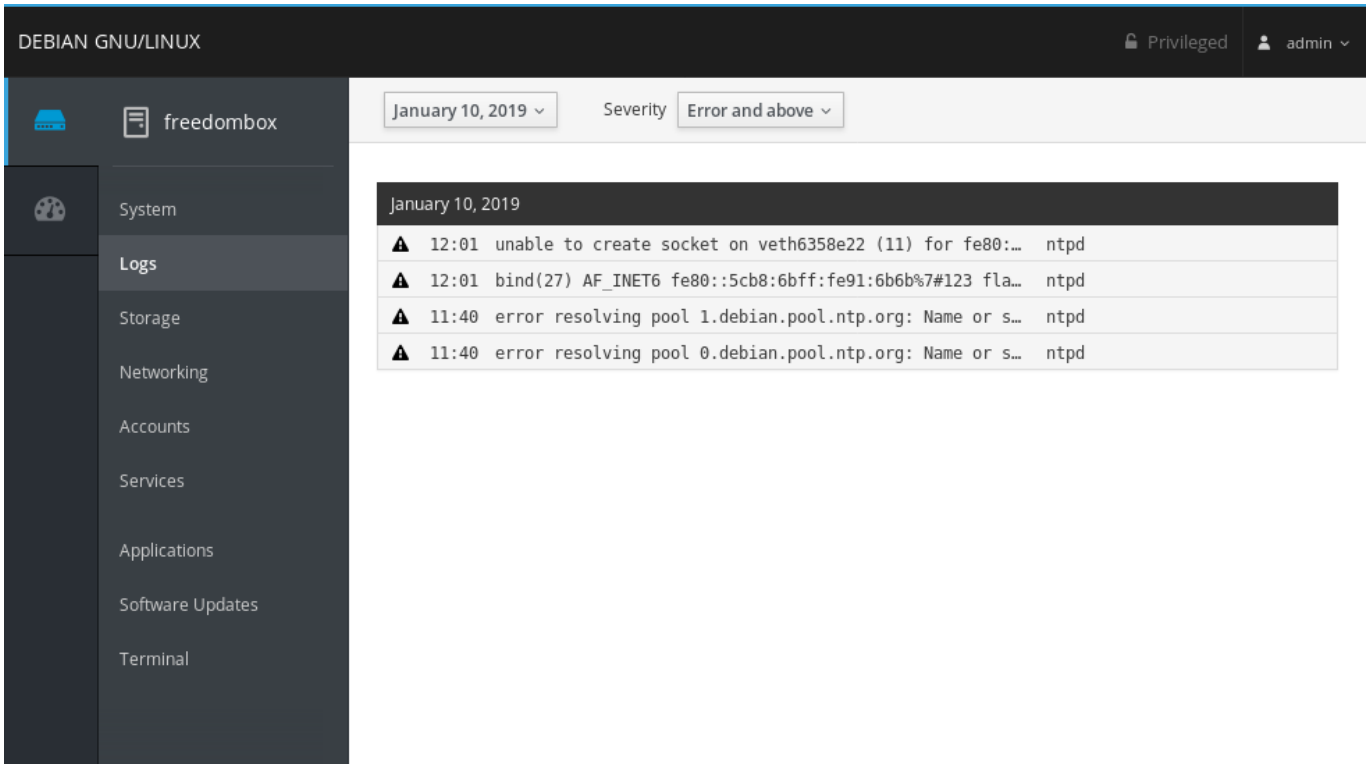
- Shows detailed hardware information
- Shows basic performance metrics of a system
- Allows changing system time and timezone
- Allows changing hostname. Please use FreedomBox UI to do this

- Shows SSH server fingerprints



6.3.2.2 Viewing System Logs

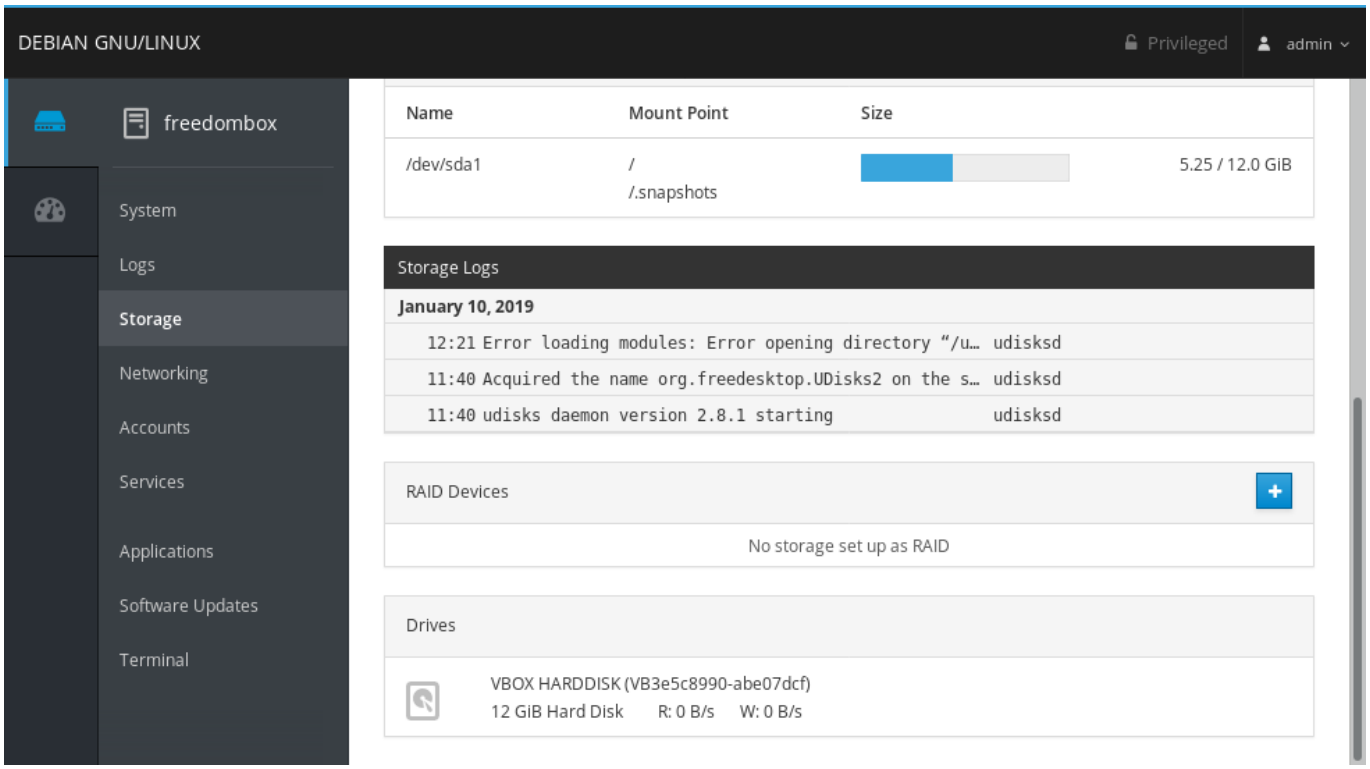
Cockpit allows querying system logs and examining them in full detail.

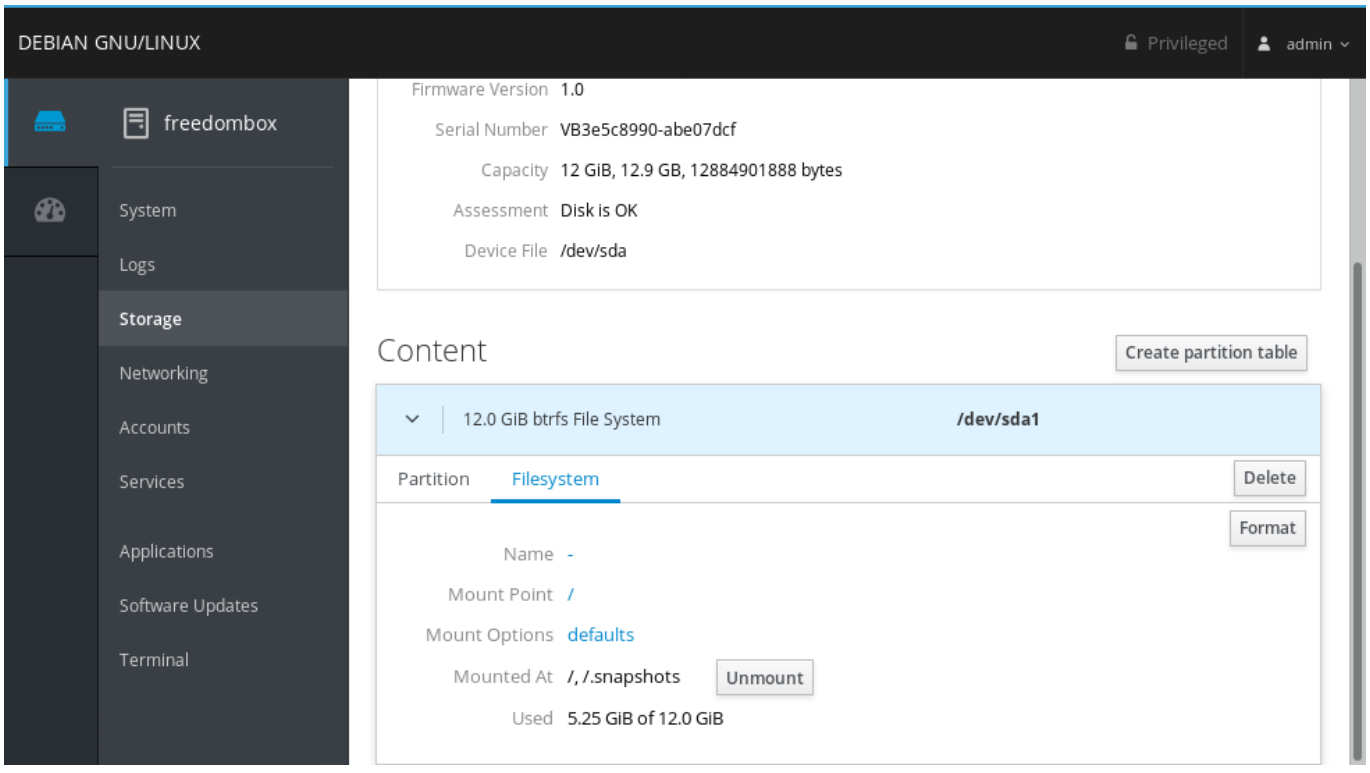


6.3.2.3 Managing Storage

Cockpit allows following advanced storage functions:

- View full disk information
- Editing disk partitions
- RAID management

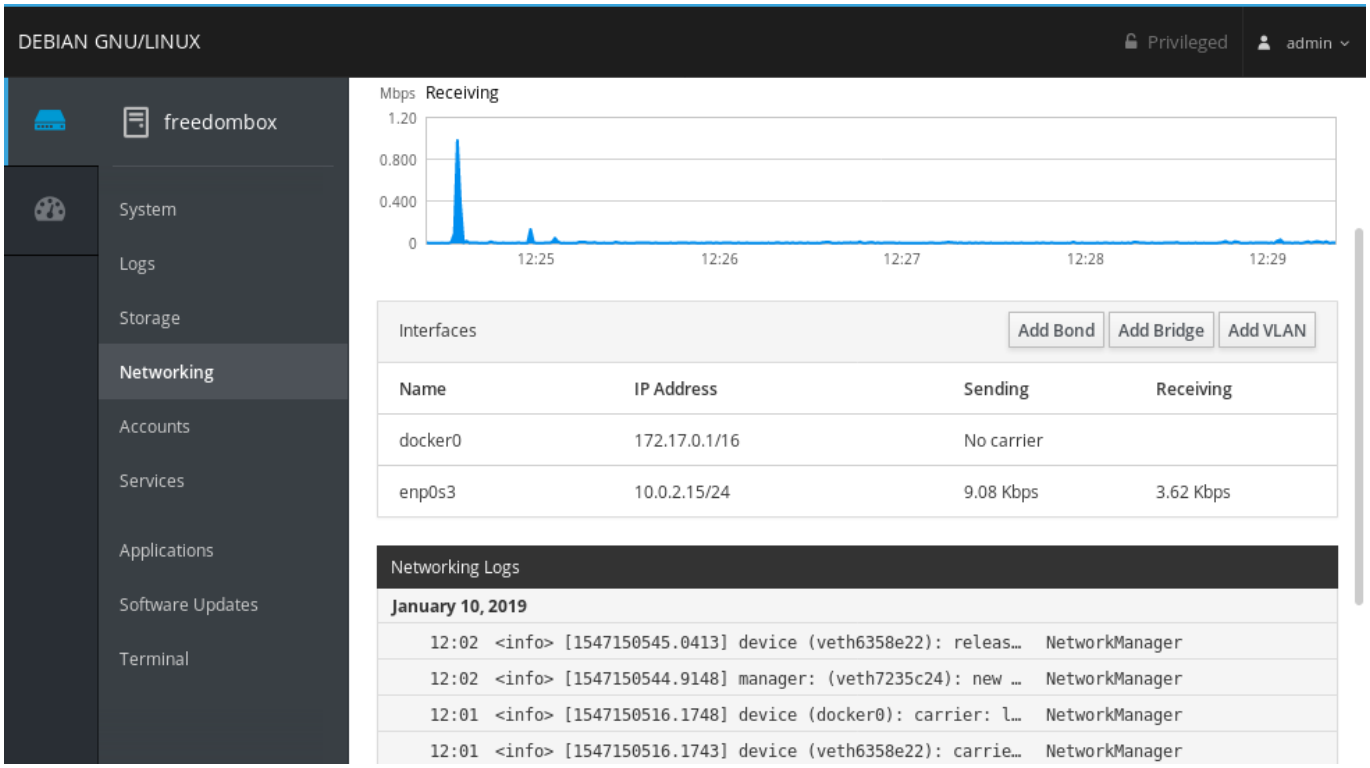


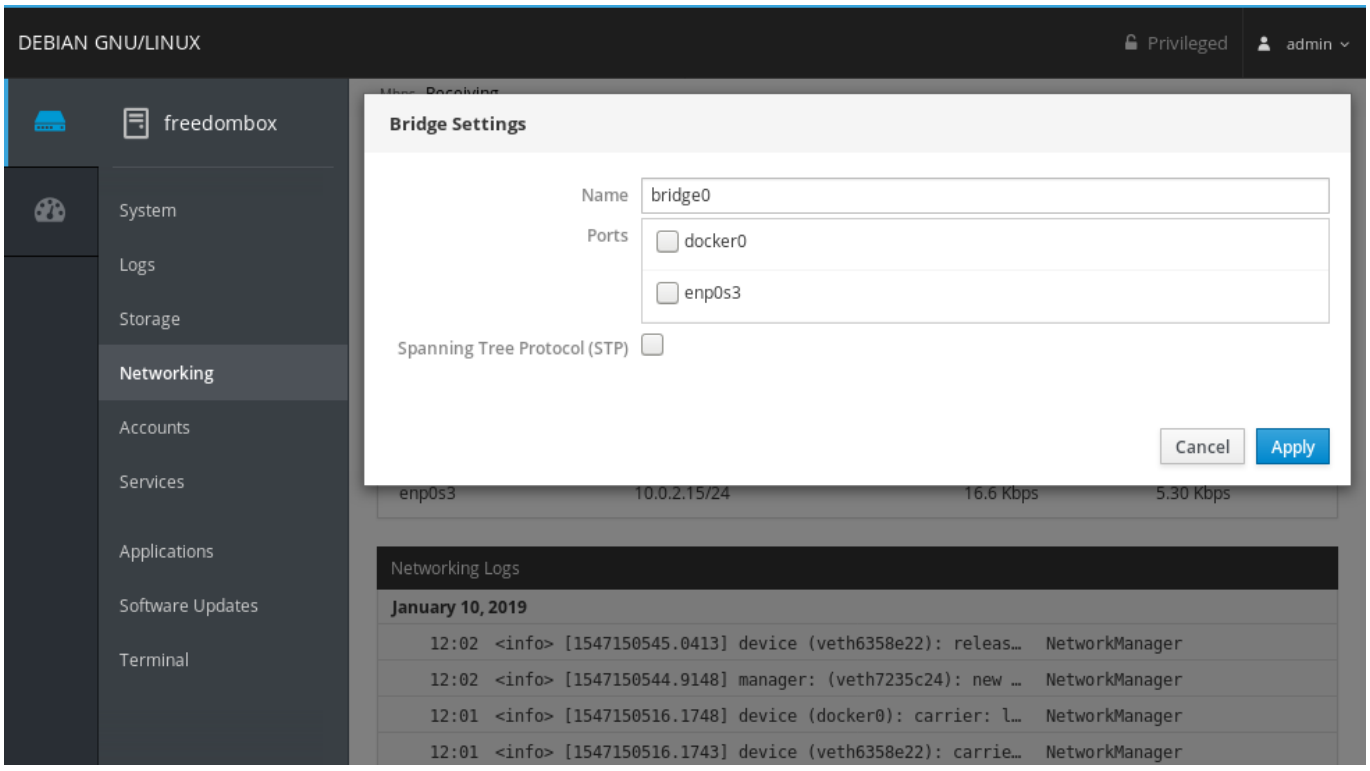


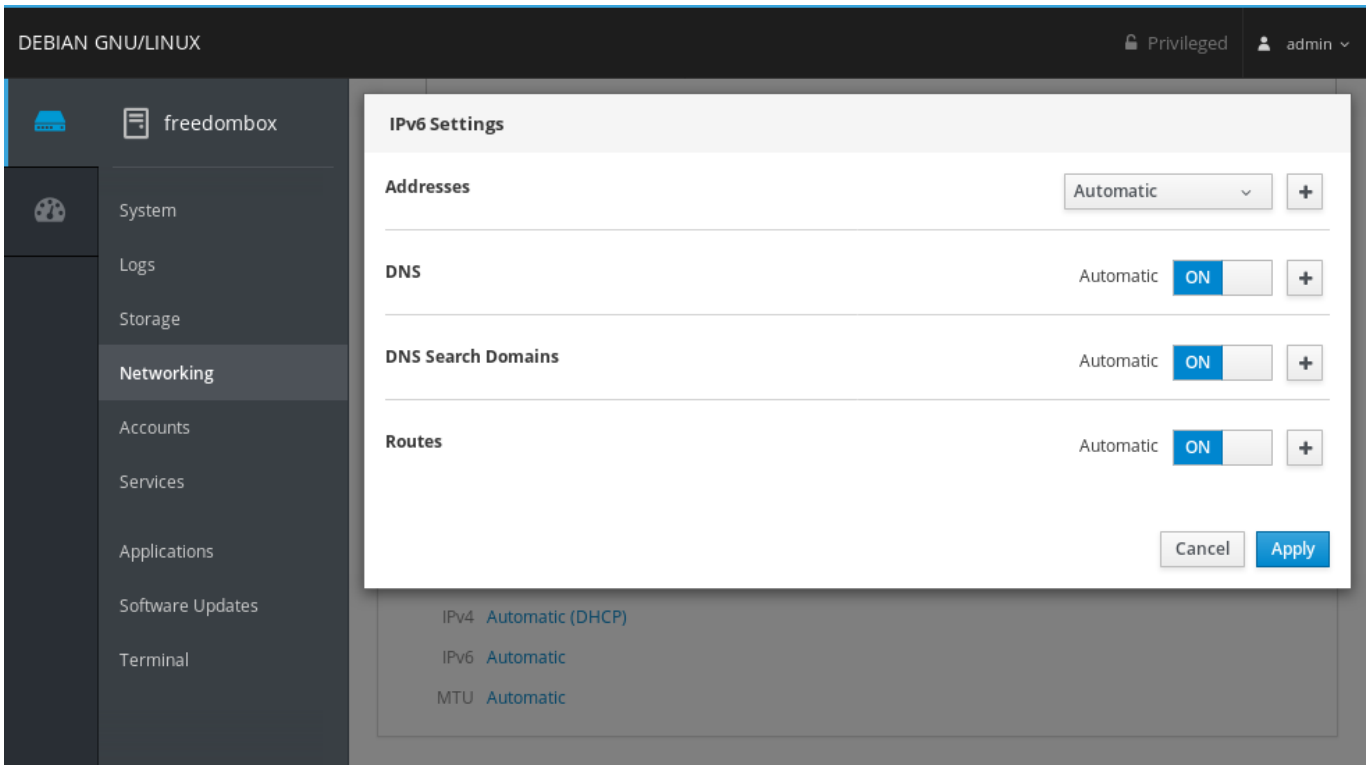
6.3.2.4 Networking

Cockpit and FreedomBox both rely on NetworkManager to configure the network. However, Cockpit offers some advanced configuration not available on FreedomBox:

- Route configuration
- Configure Bonds, Bridges, VLANs

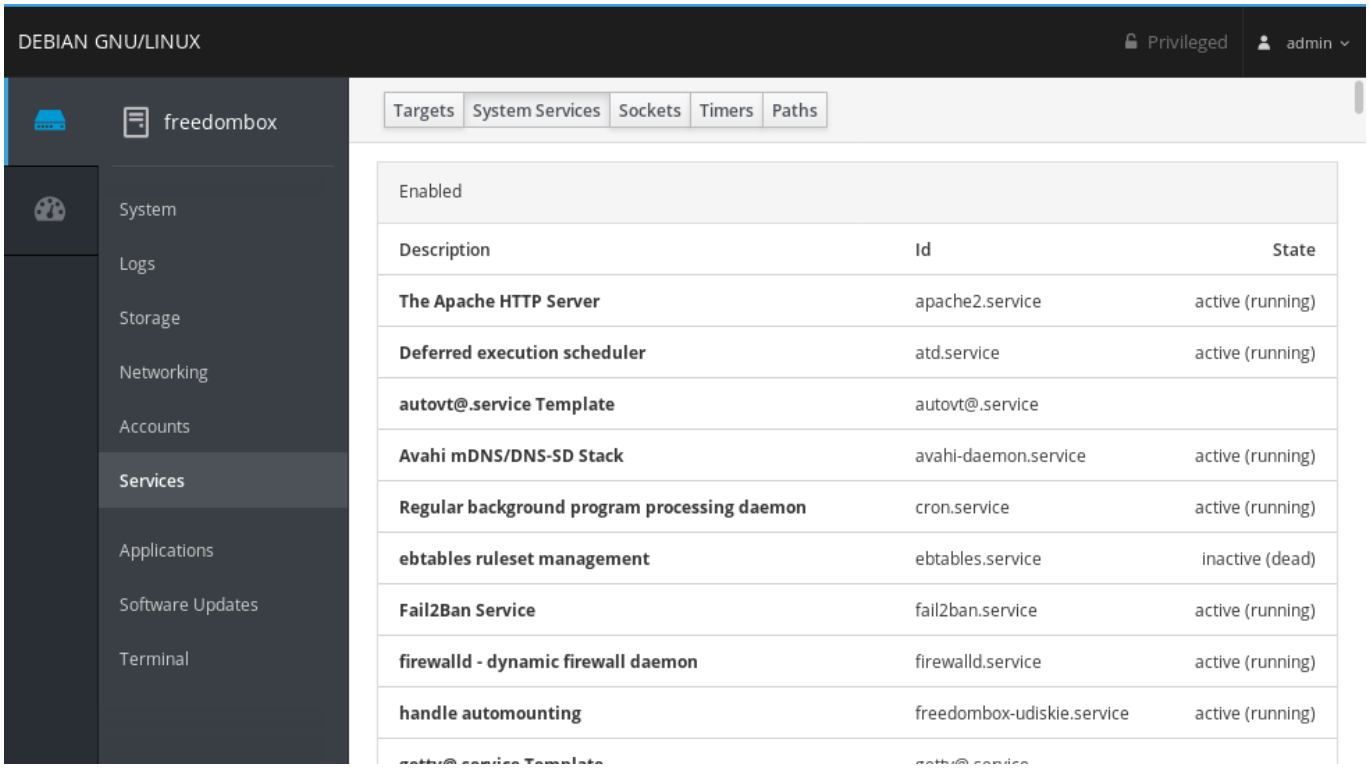


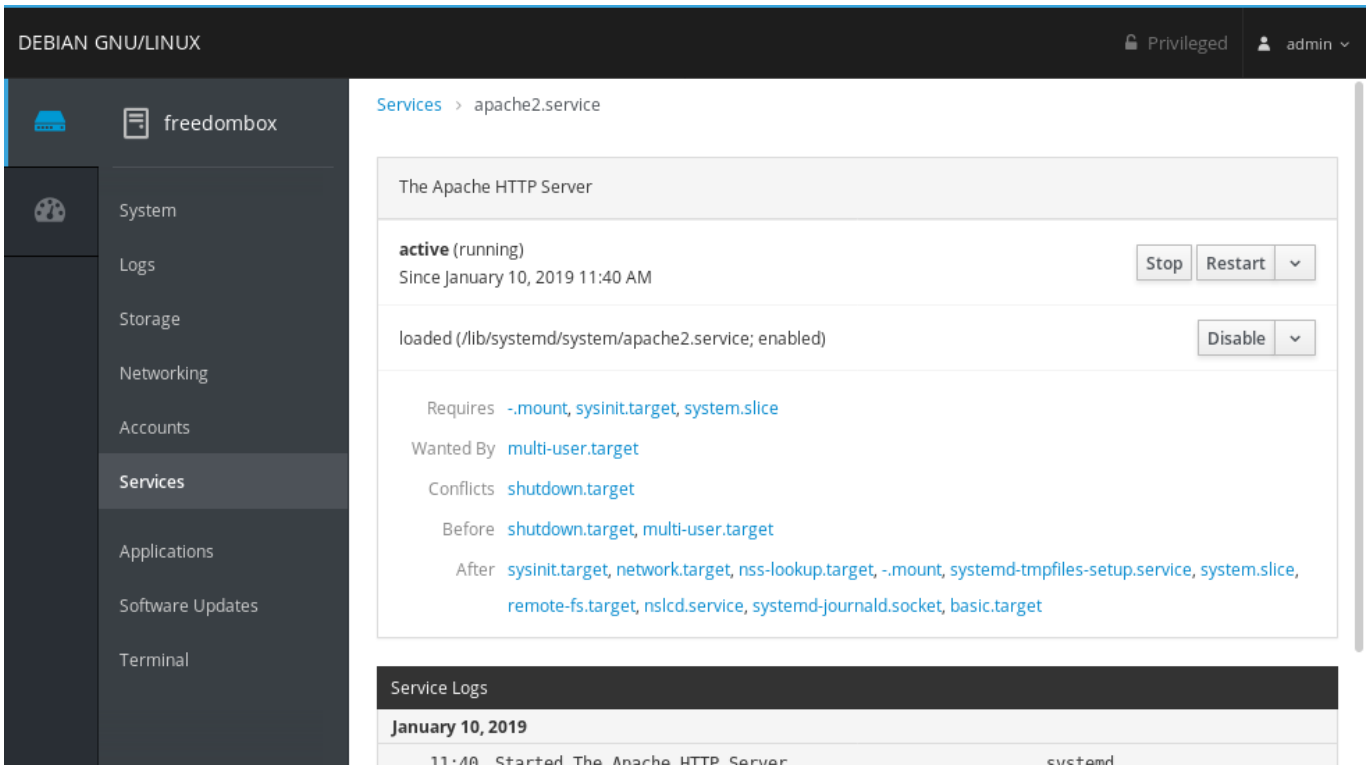




6.3.2.5 Services

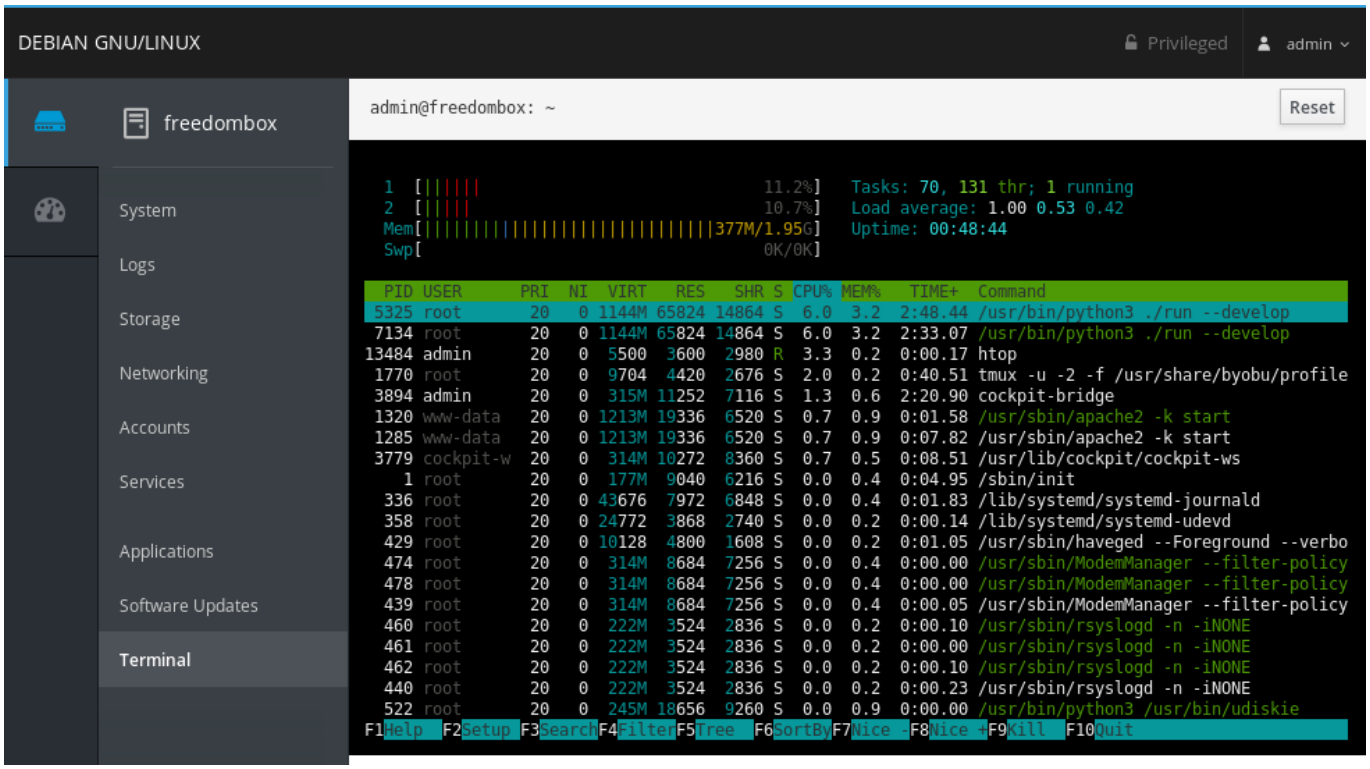
Cockpit allows management of services and periodic jobs (similar to cron).





6.3.2.6 Web Terminal

Cockpit offers a web based terminal that can be used perform manual system administration tasks.



6.3.3 Troubleshooting

Cockpit requires a domain name to be properly setup on your FreedomBox and will only work when you access it using a URL with that domain name. Cockpit will not work when using IP address in the URL. Using *freedombox.local* as the domain name also does not work. For example, the following URLs will not work:

```
https://192.168.0.10/_cockpit/
https://freedombox.local/_cockpit/
```

Starting with FreedomBox version 19.15, using *.local* domain works. You can access Cockpit using the URL `https://freedombox.local/_cockpit/`. The *.local* domain is based on your hostname. If your hostname is *mybox*, your *.local* domain name will be *mybox.local* and the Cockpit URL will be `https://mybox.local/_cockpit/`.

To properly access Cockpit, use the domain name configured for your FreedomBox.Cockpit will also work well when using a Tor Onion Service. The following URLs will work:

```
https://mybox.freedombox.rocks/_cockpit/
https://exampletorhs.onion/_cockpit/
```

The reason for this behaviour is that Cockpit uses WebSockets to connect to the backend server. Cross site requests for WebSockets must be prevented for security reasons. To implement this, Cockpit maintains a list of all domains from which requests are allowed. FreedomBox automatically configures this list whenever you add or remove a domain. However, since we can't rely on IP addresses, they are not added by FreedomBox to this domain list. You can see the current list of allowed domains, as managed by FreedomBox, in */etc/cockpit/cockpit.conf*. You may edit this, but do so only if you understand web security consequences of this.

6.3.4 External links

- Upstream project: `https://cockpit-project.org`
- User documentation: `https://cockpit-project.org/guide/latest/`

6.4 Configure

Configure has some general configuration options:

6.4.1 Hostname

- Hostname is the local name by which other devices on the local network can reach your FreedomBox. The default hostname is *freedombox*.

6.4.2 Domain Name

- Domain name is the global name by which other devices on the Internet can reach your FreedomBox. The value set here is used by the Chat Server (XMPP), Matrix Synapse, and Certificates (Let's Encrypt).

6.4.3 Webserver Home Page

- This is an advanced option that allows you to set something other than FreedomBox Service as the home page to be served on the domain name of the FreedomBox. For example, if your FreedomBox's domain name is `https://myfreedombox.rocks` and you set MediaWiki as the home page, visiting `https://myfreedombox.rocks` will take you to `https://myfreedombox.rocks/mediawiki/` instead of the usual `https://myfreedombox.rocks/plinth/`.

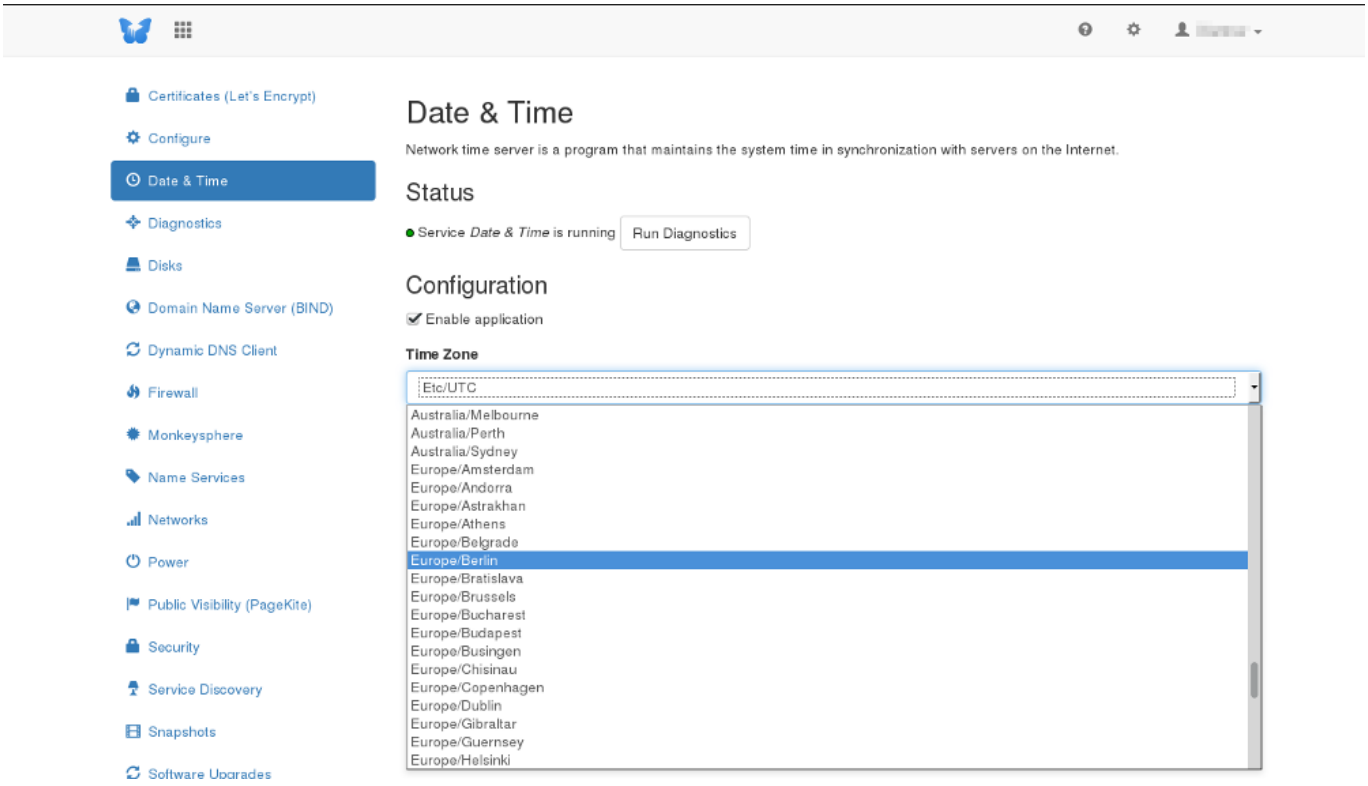
Caution
Once some other app is set as the home page, you can only navigate to the FreedomBox Service by typing `https://myfreedombox.rocks/plinth/` into the browser.
/freedombox can also be used as an alias to */plinth*

- You can set any web application, Ikiwiki wikis and blogs or Apache's default index.html page as the web server home page. Since release **20.20** you can also select a user's website among those users who have created their **public_html** directory.
- *Tip:* Bookmark the URL of FreedomBox Service before setting the home page to some other app.

6.5 Date & Time

This network time server is a program that maintains the system time in synchronization with servers on the Internet.

You can select your time zone by picking a big city nearby (they are sorted by *Continent/City*) or select directly the zone with respect to GMT (Greenwich Mean Time).



6.6 Diagnostics

The system diagnostic test will run a number of checks on your system to confirm that applications and services are working as expected.

Just click *Run Diagnostics*. This may take some minutes.

6.7 Dynamic DNS Client

6.7.1 What is Dynamic DNS?

In order to reach a server on the Internet, the server needs to have permanent address also known as the static IP address. Many Internet service providers don't provide home users with a static IP address or they charge more providing a static IP address.

Instead they provide the home user with an IP address that changes every time the user connects to the Internet. Clients wishing to contact the server will have difficulty reaching the server.

Dynamic DNS service providers assist in working around a problem. First they provide you with a domain name, such as 'myhost.example.org'. Then they associate your IP address, whenever it changes, with this domain name. Then anyone intending to reach the server will be to contact the server using the domain name 'myhost.example.org' which always points to the latest IP address of the server.

For this to work, every time you connect to the Internet, you will have to tell your Dynamic DNS provider what your current IP address is. Hence you need special software on your server to perform this operation. The Dynamic DNS function in FreedomBox will allow users without a static public IP address to push the current public IP address to a Dynamic DNS Server. This allows you to expose services on FreedomBox, such as ownCloud, to the Internet.

6.7.2 A Free Domain for your FreedomBox

You can get a free domain name for your FreedomBox using the free Dynamic DNS service provided by the FreedomBox community at `https://ddns.freedombox.org` . With this service, your domain name will look like *myaccount.fbx.one* or *myaccount.freedombox.rocks* where *myaccount* is the account you created on the service.

The service also provides you with free unlimited number of subdomains. For this, login to the account, go to *Change Settings*, check the option *Wild Card*, and click *Save Changes*. After this you can add any number of subdomains like *foo.myaccount.fbx.one* to your FreedomBox in *System -> Names -> Add Domain (regular)*. TLS certificates (for HTTPS) will be automatically obtained by FreedomBox using Let's Encrypt for these subdomains. Subdomains are especially useful for applications that require being hosted on a separate domain, such as Home Assistant.

6.7.3 GnuDIP vs. Update URL

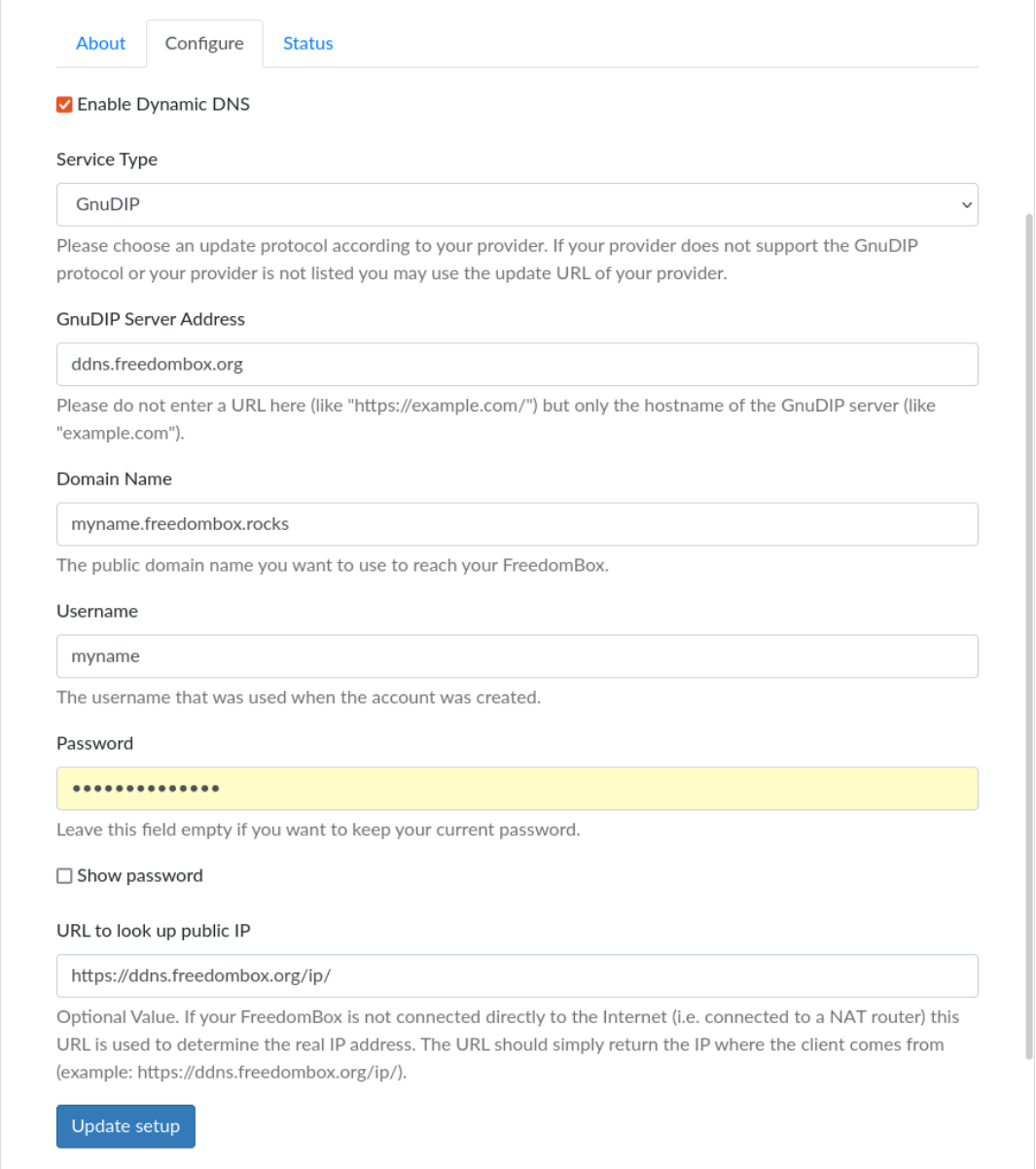
There are two main mechanism to notify the Dynamic DNS server of your new IP address; using the *GnuDIP* protocol and using the *Update URL* mechanism.

If a service provided using update URL is not properly secured using HTTPS, your credentials may be visible to an adversary. Once an adversary gains your credentials, they will be able to replay your request your server and hijack your domain.

On the other hand, the GnuDIP protocol will only transport a salted MD5 value of your password, in a way that is secure against replay attacks.

6.7.4 Using the GnuDIP protocol

1. Register an account with any Dynamic DNS service provider. A free service provided by the FreedomBox community is available at `https://ddns.freedombox.org` .
2. In FreedomBox UI, enable the Dynamic DNS Service.
3. Select *GnuDIP* as *Service type*, enter your Dynamic DNS service provider address (for example, ddns.freedombox.org) into *GnuDIP Server Address* field.



4. Fill *Domain Name*, *Username*, *Password* information given by your provider into the corresponding fields.

6.7.5 Using an Update URL

This feature is implemented because the most popular Dynamic DNS providers are using Update URLs mechanism.

1. Register an account with a Dynamic DNS service provider providing their service using Update URL mechanism. Some example providers are listed in the configuration page itself.
2. In FreedomBox UI, enable the Dynamic DNS service.
3. Select *other Update URL* as *Service type*, enter the update URL given by your provider into *Update URL* field.
4. If you browse the update URL with your Internet browser and a warning message about untrusted certificate appears, then enable *accept all SSL certificates*. WARNING: your credentials may be readable here because man-in-the-middle attacks are possible! Consider choosing a better service provider instead.
5. If you browse the update URL with your Internet browser and the username/password box appears, enable *use HTTP basic authentication* checkbox and provide the *Username* and *Password*.
6. If the update URL contains your current IP address, replace the IP address with the string *<Ip>*.

6.7.6 Checking If It Works

1. Make sure that external services you have enabled such as /jwchat, /roundcube and /ikiwiki are available on your domain address.
2. Go to the *Status* page, make sure that the NAT type is detected correctly. If your FreedomBox is behind a NAT device, this should be detected over there (Text: *Behind NAT*). If your FreedomBox has a public IP address assigned, the text should be "Direct connection to the Internet".
3. Check that the last update status is not *failed*.



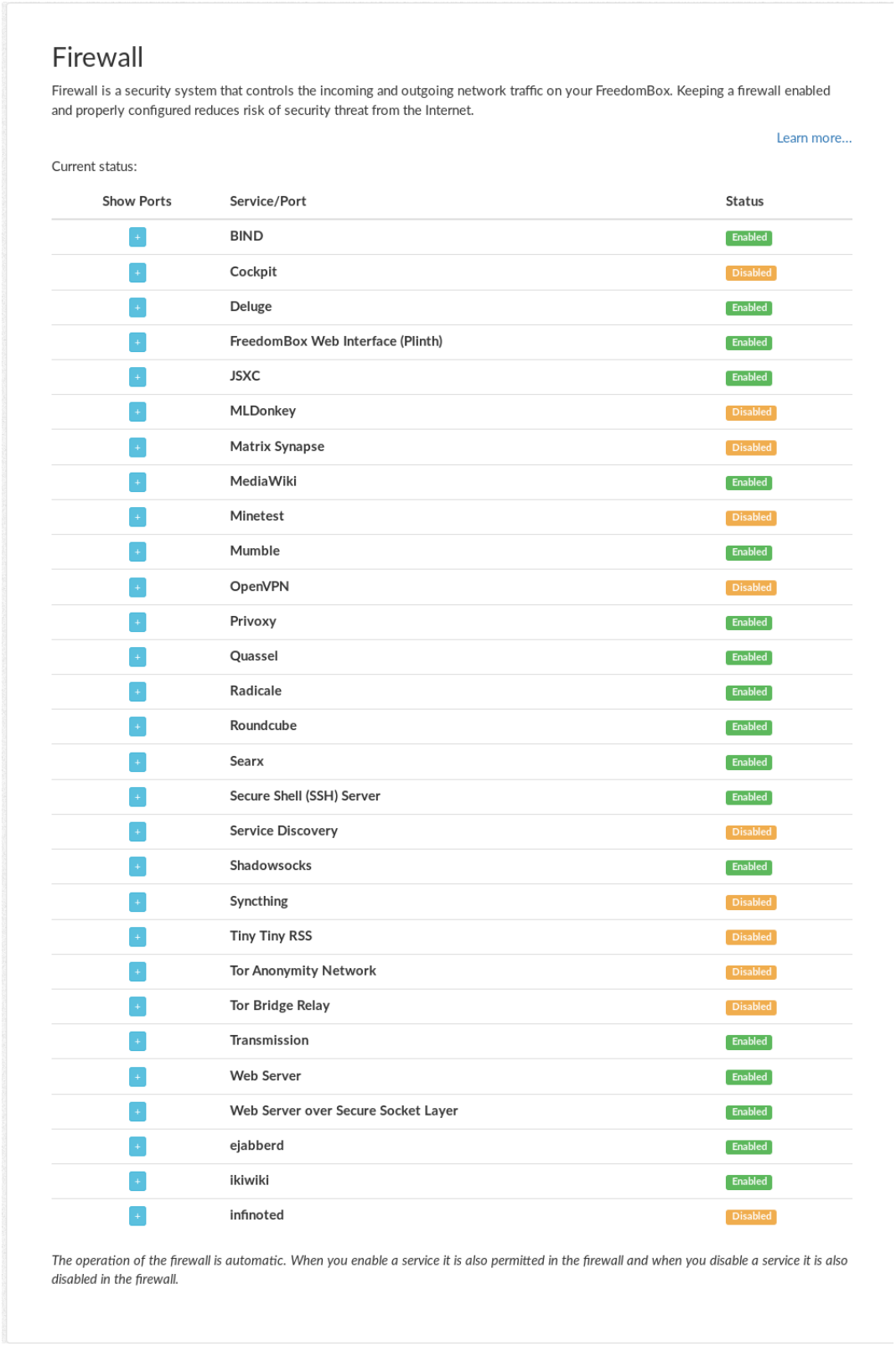
Recap: How to create a DNS name with GnuDIP

1. Access to GnuIP login page (answer Yes to all pop ups)
2. Click on "Self Register"
3. Fill the registration form (Username and domain will form the public IP address [username.domain])
4. Take note of the username/hostname and password that will be used on the FreedomBox app.
5. Save and return to the GnuDIP login page to verify your username, domain and password (enter the datas, click login).
6. Login output should display your new domain name along with your current public IP address (this is a unique address provided by your router for all your local devices).
7. Leave the GnuDIP interface and open the Dynamic DNS Client app page in your FreedomBox.
8. Click on "Set Up" in the top menu.
9. Activate Dynamic DNS
10. Choose GnuDIP service.
11. Add server address (ddns.freedombox.org)
12. Add your fresh domain name (username.domain, ie [username].freedombox.rocks)
13. Add your fresh username (the one used in your new IP address) and password
14. Add your GnuDIP password
15. Fill the option with `https://ddns.freedombox.org/ip/` (try this url in your browser, you will figure out immediately)

6.8 Firewall

Firewall is a network security system that controls the incoming and outgoing network traffic. Keeping a firewall enabled and properly configured reduces risk of security threat from the Internet.

The operation of the firewall in FreedomBox web interface is automatic. When you enable a service it is automatically permitted in the firewall and when you disable a service it is automatically disabled in the firewall. For services which are enabled by default on FreedomBox, firewall ports are also enabled by default during the first run process.



Firewall

Firewall is a security system that controls the incoming and outgoing network traffic on your FreedomBox. Keeping a firewall enabled and properly configured reduces risk of security threat from the Internet.

Learn more...

Current status:

Show Ports	Service/Port	Status
+	BIND	Enabled
+	Cockpit	Disabled
+	Deluge	Enabled
+	FreedomBox Web Interface (Plinth)	Enabled
+	JSXC	Enabled
+	MLDonkey	Disabled
+	Matrix Synapse	Disabled
+	MediaWiki	Enabled
+	Minetest	Disabled
+	Mumble	Enabled
+	OpenVPN	Disabled
+	Privoxy	Enabled
+	Quassel	Enabled
+	Radicale	Enabled
+	Roundcube	Enabled
+	Searx	Enabled
+	Secure Shell (SSH) Server	Enabled
+	Service Discovery	Disabled
+	Shadowsocks	Enabled
+	Syncthing	Disabled
+	Tiny Tiny RSS	Disabled
+	Tor Anonymity Network	Disabled
+	Tor Bridge Relay	Disabled
+	Transmission	Enabled
+	Web Server	Enabled
+	Web Server over Secure Socket Layer	Enabled
+	ejabberd	Enabled
+	ikiwiki	Enabled
+	infinoted	Disabled

The operation of the firewall is automatic. When you enable a service it is also permitted in the firewall and when you disable a service it is also disabled in the firewall.

Firewall management in FreedomBox is done using FirewallD.

6.8.1 Interfaces

Each interface is needs to be assigned to one (and only one) zone. If an interface is not assigned any zone, it is automatically assigned `external` zone. Whatever rules are in effect for a zone, those rules start to apply for that interface. For example,

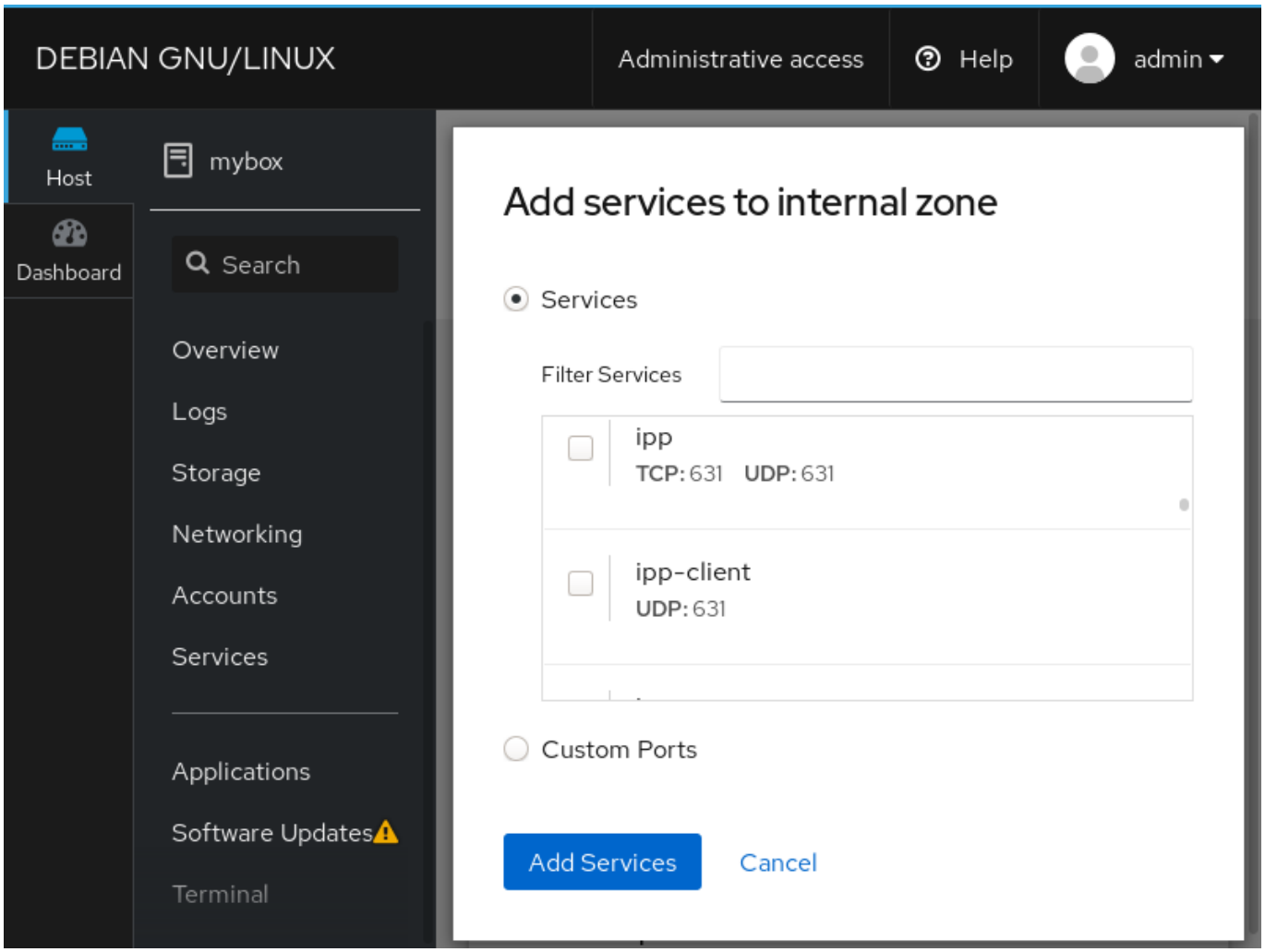
if HTTP traffic is allowed in a particular zone, then web requests will be accepted on all the addresses configured for all the interfaces assigned to that zone.

There are primarily two firewall zones used. The `internal` zone is meant for services that are provided to all machines on the local network. This may include services such as streaming media and simple file sharing. The `external` zone is meant for services that are provided publicly on the Internet. This may include services such as blog, website, email web client etc.

For details on how network interfaces are configured by default, see the Networks section.

6.8.2 Opening Custom Ports

Cockpit app provides advanced management of firewall. Both FreedomBox and Cockpit operate over firewalld and are hence compatible with each other. In particular, Cockpit can be used to open custom services or ports on FreedomBox. This is useful if you are manually running your own services in addition to the services provided by FreedomBox on the same machine.



6.8.3 FreedomBox Ports/Services

The following table attempts to document the ports, services and their default statuses in FreedomBox. If you find this page outdated, see the Firewall status page in FreedomBox interface.

Service	Port	External	Enabled by default	Status shown in FreedomBox	Managed by FreedomBox
Minetest	30000/udp	☆	⛔	✅	✅

XMPP Client	5222/tcp	☆	⛔	✔	✔
XMPP Server	5269/tcp	☆	⛔	✔	✔
XMPP Bosh	5280/tcp	☆	⛔	✔	✔
NTP	123/udp	☆	✔	✔	✔
FreedomBox Web Interface (Plinth)	443/tcp	☆	✔	✔	⛔
Quassel	4242/tcp	☆	⛔	✔	✔
SIP	5060/tcp	☆	⛔	✔	✔
SIP	5060/udp	☆	⛔	✔	✔
SIP-TLS	5061/tcp	☆	⛔	✔	✔
SIP-TLS	5061/udp	☆	⛔	✔	✔
RTP	1024-65535/udp	☆	⛔	✔	✔
SSH	22/tcp	☆	✔	✔	⛔
mDNS	5353/udp	☆	✔	✔	✔
Tor (Socks)	9050/tcp	☆	⛔	✔	✔
Obfsproxy	<random>/tcp	☆	⛔	✔	✔
OpenVPN	1194/udp	☆	⛔	✔	✔
Mumble	64378/tcp	☆	⛔	✔	✔
Mumble	64378/udp	☆	⛔	✔	✔
Privoxy	8118/tcp	☆	⛔	✔	✔
JSXC	80/tcp	☆	⛔	⛔	⛔
JSXC	443/tcp	☆	⛔	⛔	⛔
DNS	53/tcp	☆	⛔	⛔	⛔
DNS	53/udp	☆	⛔	⛔	⛔
DHCP	67/udp	☆	✔	⛔	⛔
Bootp	67/tcp	☆	⛔	⛔	⛔
Bootp	67/udp	☆	⛔	⛔	⛔
Bootp	68/tcp	☆	⛔	⛔	⛔
Bootp	68/udp	☆	⛔	⛔	⛔
LDAP	389/tcp	☆	⛔	⛔	⛔
LDAPS	636/tcp	☆	⛔	⛔	⛔

6.8.4 Manual operation

See FirewallD documentation for more information on the basic concepts and comprehensive documentation.

6.8.4.1 Enable/disable firewall

To disable firewall

```
service firewalld stop
```

or with systemd

```
systemctl stop firewalld
```

To re-enable firewall

```
service firewalld start
```

or with systemd

```
systemctl start firewalld
```

6.8.4.2 Modifying services/ports

You can manually add or remove a service from a zone.

To see list of services enabled:

```
firewall-cmd --zone=<zone> --list-services
```

Example:

```
firewall-cmd --zone=internal --list-services
```

To see list of ports enabled:

```
firewall-cmd --zone=<zone> --list-ports
```

Example:

```
firewall-cmd --zone=internal --list-ports
```

To remove a service from a zone:

```
firewall-cmd --zone=<zone> --remove-service=<service>
firewall-cmd --permanent --zone=<zone> --remove-service=<interface>
```

Example:

```
firewall-cmd --zone=internal --remove-service=xmpp-bosh
firewall-cmd --permanent --zone=internal --remove-service=xmpp-bosh
```

To remove a port from a zone:

```
firewall-cmd --zone=internal --remove-port=<port>/<protocol>
firewall-cmd --permanent --zone=internal --remove-port=<port>/<protocol>
```

Example:

```
firewall-cmd --zone=internal --remove-port=5353/udp
firewall-cmd --permanent --zone=internal --remove-port=5353/udp
```

To add a service to a zone:

```
firewall-cmd --zone=<zone> --add-service=<service>
firewall-cmd --permanent --zone=<zone> --add-service=<interface>
```

Example:

```
firewall-cmd --zone=internal --add-service=xmpp-bosh
firewall-cmd --permanent --zone=internal --add-service=xmpp-bosh
```

To add a port to a zone:

```
firewall-cmd --zone=internal --add-port=<port>/<protocol>
firewall-cmd --permanent --zone=internal --add-port=<port>/<protocol>
```

Example:

```
firewall-cmd --zone=internal --add-port=5353/udp
firewall-cmd --permanent --zone=internal --add-port=5353/udp
```

6.8.4.3 Modifying the zone of interfaces

You can manually change the assignment of zones of each interfaces after they have been autuomatically assigned by the first boot process.

To see current assignment of interfaces to zones:

```
firewall-cmd --list-all-zones
```

To remove an interface from a zone:

```
firewall-cmd --zone=<zone> --remove-interface=<interface>
firewall-cmd --permanent --zone=<zone> --remove-interface=<interface>
```

Example:

```
firewall-cmd --zone=external --remove-interface=eth0
firewall-cmd --permanent --zone=external --remove-interface=eth0
```

To add an interface to a zone:

```
firewall-cmd --zone=<zone> --add-interface=<interface>
firewall-cmd --permanent --zone=<zone> --add-interface=<interface>
```

Example:

```
firewall-cmd --zone=internal --add-interface=eth0
firewall-cmd --permanent --zone=internal --add-interface=eth0
```

6.9 Let's Encrypt (Certificates)

A digital certificate allows users of a web service to verify the identity of the service and to securely communicate with it. FreedomBox can automatically obtain and setup digital certificates for each available domain. It does so by proving itself to be the owner of a domain to Let's Encrypt, a certificate authority (CA).

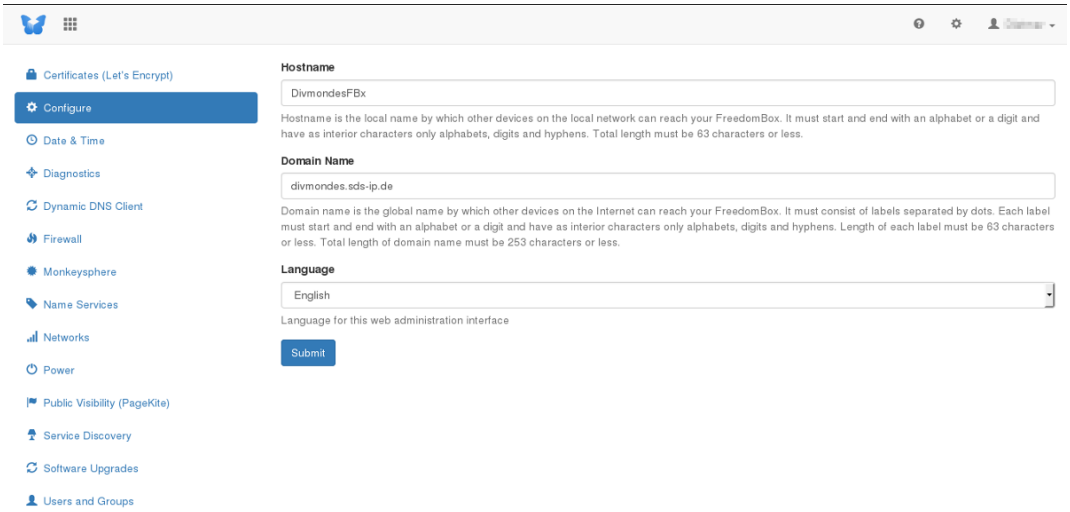
Let's Encrypt is a free, automated, and open certificate authority, run for the public's benefit by the Internet Security Research Group (ISRG). Please read and agree with the Let's Encrypt Subscriber Agreement before using this service.

6.9.1 Why using Certificates

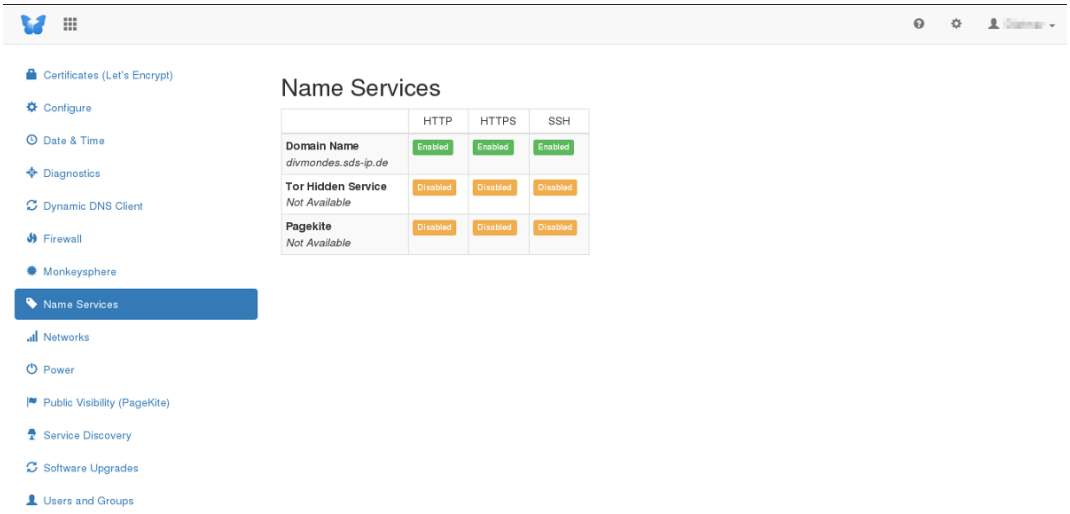
The communication with your FreedomBox can be secured so that it is not possible to intercept the content of the web pages viewed and about the content exchanged.

6.9.2 How to setup

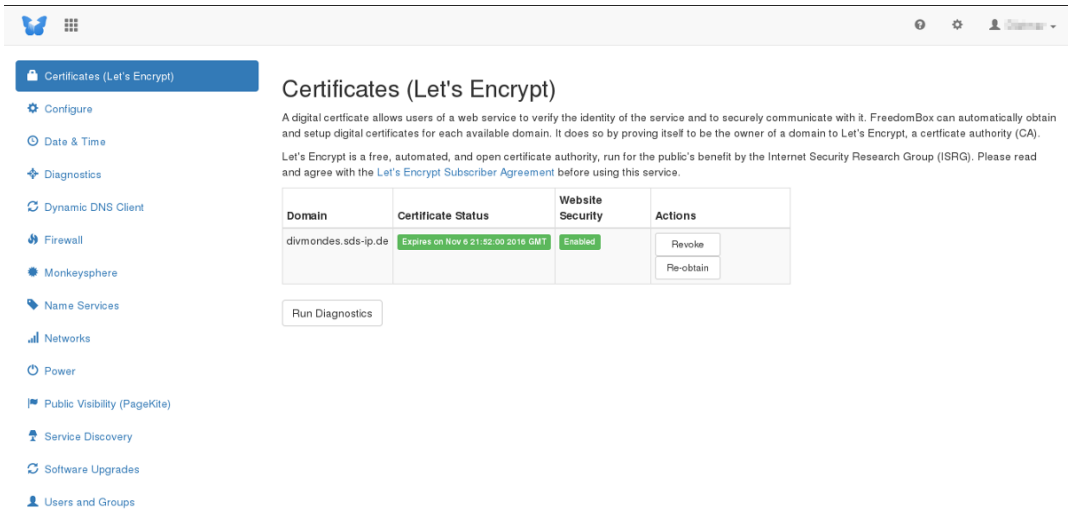
1. If your FreedomBox is behind a router, you will need to set up port forwarding on your router. You should forward the following ports:
   - TCP 80 (http)
   - TCP 443 (https)
2. Make the domain name known:
   - In Configure insert your *domain name*, e.g. `MyWebName.com`



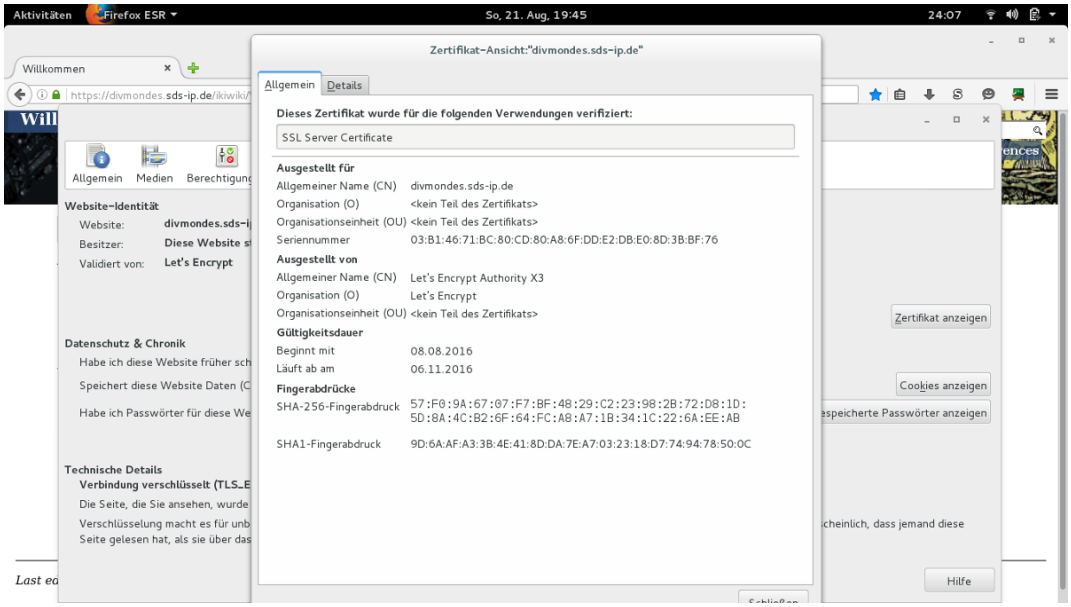
3. Verify the domain name was accepted
   - Check that it is enabled in Name Services



4. Go to the Certificates (Let's Encrypt) page, and complete the module install if needed. Then click the "Obtain" button for your domain name.
   - After some minutes a valid certificate is available



5. Verify in your browser by checking `https://MyWebName.com`



Screencast: Let's Encrypt

6.9.3 Using

The certificate is valid for 3 months. It is renewed automatically and can also be re-obtained or revoked manually.

With running *diagnostics* the certificate can also be verified.

6.9.4 External links

- Upstream project: https://letsencrypt.org
- User documentation: https://letsencrypt.org/docs/

6.10 Name Services

Name Services provides an overview of ways the box can be reached from the public Internet: domain name, Tor Onion Service, and Pagekite. For each type of name, it is shown whether the HTTP, HTTPS, and SSH services are enabled or disabled for incoming connections through the given name. It also shows and allows configuring how FreedomBox performs domain name resolutions.

6.10.1 Adding Domain Name to FreedomBox

Domain names are unique names assigned to computers and servers to identify and reach them easily without having remember their more technical (IP) addresses.

Adding at least one domain name to your FreedomBox through one of the supported methods is necessary if you wish to access various apps from outside of your home network. Even at home, adding a domain name allows you to have a TLS certificate that is essential for accessing apps securely through HTTPS. The following types of domains are supported:

- A regular domain that is purchased from a domain registrar. These domains looks like `mydomain.example`. These types of domains require that your FreedomBox machine or your home network router have a public IP address provided by your ISP. All apps can be accessed using these domains.
- A dynamic domain that can obtained from a dynamic DNS provider. FreedomBox foundation provides a free dynamic DNS domain for all FreedomBox users at `https://ddns.freedombox.org` . These domains look like `mydomain.fbx.one` or `mydomain.freedombox.rocks`. These types of domains require that your FreedomBox machine or your home network router have a public IP address provided by your ISP. The public IP may be changed by the ISP regularly and DNS records will be updated accordingly. All apps can be accessed using these domains. These types of domains are configured using the Dynamic DNS app in FreedomBox.
- A Tor Onion domain that can be obtained by setting up Tor app. These domains look like `mysitenameyx4fi3l6x2gyzmtmgxjyc` These types of domains don't require a public IP address from your ISP. However, only web or SSH based services/apps work on this type of domain. Further, only users of Tor will be able to reach these domains. A regular browser will not be able to reach these services. These types of domains are configured using the Tor app in FreedomBox.
- A PageKite domain that can obtained from a PageKite service provider. This service can be purchased at `https://pagekite.net` . These domains look like `https://mydomain.pagekite.net`. These types of domains don't require a public IP address from your ISP. However, only web or SSH based services/apps work on this type of domain. These types of domains are configured using the PageKite app in FreedomBox.

6.10.2 Subdomains

Subdomains are similar to domains, but reside under the a domain name. For example, for a domain `mydomain.example` a subdomain could be `myapp.mydomain.example`. Some applications such as Home Assistant in FreedomBox require you to setup and access them over a subdomain instead of a URL path. You can obtain and setup subdomains in the following ways:

- If you own a regular domain (as described above), you can login to the web console provided by your domain registrar and add a subdomain. Make the subdomain a CNAME of the original domain or assign A and AAAA records similar to the original domain. See your domain registrar's documentation for more information. After adding DNS entries for the subdomain, let FreedomBox know that it can use the subdomain by adding it as another regular domain. TLS certificates (for HTTPS) will be automatically obtained by FreedomBox using Let's Encrypt for these subdomains.
- If you have a dynamic domain name from FreedomBox foundation, you can also get an unlimited number of free subdomains. For this, login to your account at `https://ddns.freedombox.org,` go to `Change Settings`, check the option `Wild Card`, and click `Save Changes`. After this you can add any number of subdomains like `foo.myaccount.fbx.one` to your FreedomBox in `System` -> `Names` -> `Add Domain (regular)`. TLS certificates (for HTTPS) will be automatically obtained by FreedomBox using Let's Encrypt for these subdomains. Also see Dynamic DNS documentation page.

6.10.3 systemd-resolved

From release 24.19, FreedomBox uses systemd-resolved as caching DNS resolver and replaces resolvconf for managing DNS server configuration. This improves privacy and security. Newer installations will come with systemd-resolved and older machines will automatically switch after an upgrade to this new release.

systemd-resolved automatically acquires DNS servers from Network Manager, the default and recommended way to configure networks on FreedomBox. However, if you are manually managing network configuration by editing /etc/network/interfaces, you will need to ensure that the DNS servers acquired are passed on to systemd-resolved. Otherwise, Fallback DNS servers will be used. See below.

6.10.4 Support for DNS-over-TLS and DNSSEC

systemd-resolved supports DNS-over-TLS. This protocol allows encrypting DNS communication between FreedomBox and the DNS server if your DNS server (typically provided by your ISP, sometimes a separate service) has support for it. This improves both privacy and security as it makes it harder for intermediaries to see the communication or manipulate it. New settings for enabling DNS-over-TLS are available at the global level (for all network interfaces) in Name Services app and at the per-connection level in the Networks app's connection settings.

systemd-resolved supports DNSSEC. This standard allows website owners to sign their DNS records allowing clients to authenticate them. This improves security by making it harder to manipulate DNS responses. If your DNS server supports this feature, it can be turned on. New setting for enabling DNSSEC is available in the Name Services app.

You can detect whether your current DNS supports DNS-over-TLS and DNSSEC by turning them on in the settings one at a time and running the diagnostics for the Names app. There is a diagnostic check which detects whether you can successfully resolve the domain name deb.debian.org.

6.10.5 Setting a custom DNS server

If your current DNS server provided by your ISP does not support DNS-over-TLS or DNSSEC features, is censoring some domains names, or if you don't trust them enough, you can instead use one of the publicly available DNS servers. This can be done by editing network connections in the Networks app and adding DNS servers manually. You will need to deactivate and re-activate the network connection (or restart FreedomBox) for the settings to become active. After this, Names app will show you the currently configured DNS servers.

6.10.6 Fallback DNS servers

In some cases, when internet connection is available to the system by no DNS servers are known to systemd-resolved, the fallback DNS servers are used. This may happen, for example, due to misconfiguration when manually managing network configuration instead of using FreedomBox's default, the Network Manager. These fallback DNS servers, as defaulted by the upstream systemd project, include servers from Cloudflare and Google DNS servers. This has privacy implications but we felt that it was important to avoid FreedomBox from becoming unreachable due to misconfiguration. It was a difficult decision. Once you have proper DNS configuration and you know that it works, you can turn off fallback DNS servers using a new setting in the Privacy app. There is also a renewed notification in the web interface that will attract your attention towards this. You may also edit the list of Fallback DNS servers by creating a configuration file for systemd-resolved. See systemd-resolved documentation.

6.11 Networks

This section describes how networking is setup by default in FreedomBox and how you can customize it. See also the Firewall section for more information on how firewall works.

6.11.1 Default setup

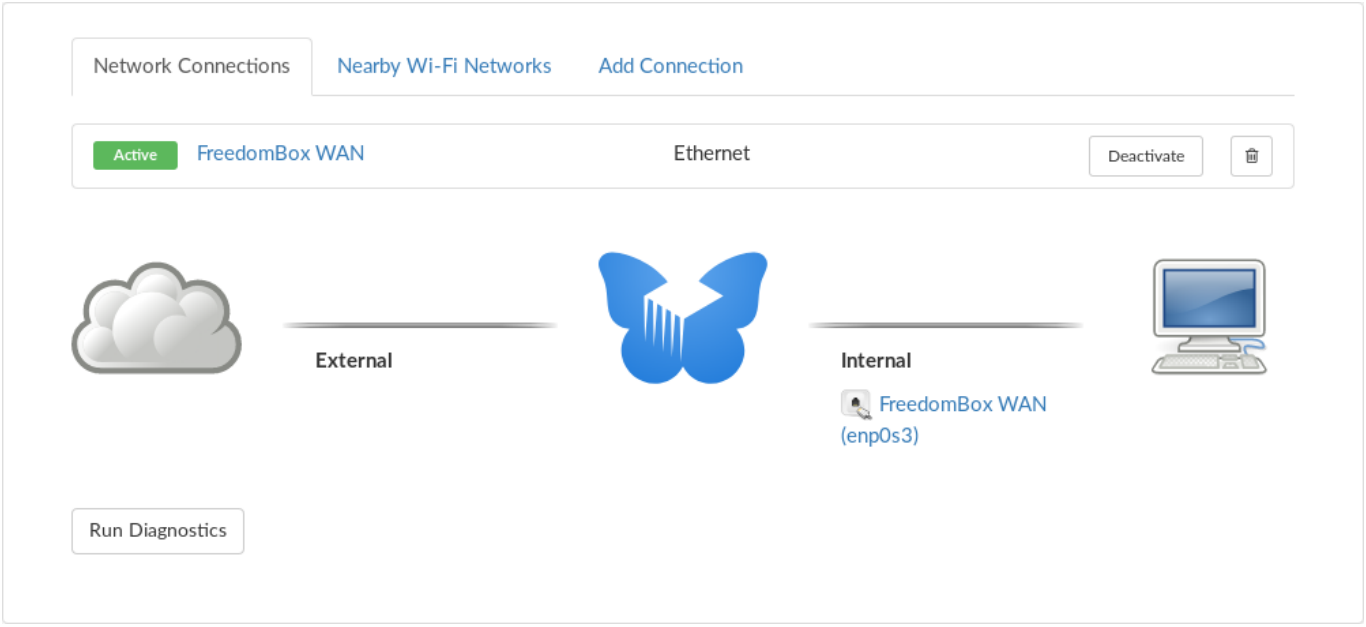
In a fresh image of FreedomBox, network is not configured at all. When the image is written to an SD card and the device boots, configuration is done. During first boot, FreedomBox setup package detects the networks interfaces and tries to automatically

configure them so that FreedomBox is available for further configuration via the web interface from another machine without the need to connect a monitor. Automatic configuration also tries to make FreedomBox useful, out of the box, for the most important scenarios FreedomBox is used for.

There are two scenarios it handles: when is a single ethernet interface and when there are multiple ethernet interfaces.

6.11.1.1 Single ethernet interface

When there is only single ethernet interface available on the hardware device, there is not much scope for it to play the role of a router. In this case, the device is assumed to be just another machine in the network. Accordingly, the only available interface is configured to be an `internal` interface in `automatic` configuration mode. This means that it connects to the Internet using the configuration provided by a router in the network and also makes all (internal and external) of its services available to all the clients on this network.



6.11.1.2 Multiple ethernet interface

When there are multiple ethernet interfaces available on the hardware device, the device can act as a router. The interfaces are then configured to perform this function.

The first network interface is configured to be an WAN or `external` interface in automatic configuration mode. This means that it connects to the Internet using network configuration provided by the Internet Service Provider (ISP). Only services that are meant to be provided across the entire Internet (external services) will be exposed on this interface. You must plug your Internet connection into the port of this ethernet interface. If you wish to continue to have your existing router manage the Internet connection for you, then plug a connection from your router to the port on this interface.

The remaining network interfaces are configured for the clients of a router. They are configured as LAN or `internal` interfaces in `shared` configuration mode. This means that all the services (both external and internal) services are provided to who ever connects on this interface. Further, the `shared` mode means that clients will be able to receive details of automatic network connection on this interface. Specifically, DHCP configuration and DNS servers are provided on this interface. The Internet connection available to the device using the first network interface will be `shared` with clients using this interface. This all means that you can connect your computers to this network interface and they will get automatically configured and will be able to access the Internet via the FreedomBox.

Currently, it is not very clear which interface will be come the WAN interface (and the remaining being LAN interfaces) although the assignment process is deterministic. So, it take a bit of trail and error to figure out which one is which. In future, for each device, this will be well documented.

6.11.1.3 Wi-Fi configuration

All Wi-Fi interfaces are configured to be LAN or `internal` interfaces in `shared` configuration mode. They are also configured to become Wi-Fi access points with following details.

- Name of the access point will be `FreedomBox` plus the name of the interface (to handle the case where there are multiple of them).
- Password for connecting to the interface will be `freedombox123`.

6.11.2 Internet Connection Sharing

Although the primary duty of FreedomBox is to provide decentralized services, it can also act like a home router. Hence, in most cases, FreedomBox connects to the Internet and provides other machines in the network the ability to use that Internet connection. FreedomBox can do this in two ways: using a `shared` mode connection or using an `internal` connection.

When an interface is set in `shared` mode, you may connect your machine directly to it. This is either by plugging in an ethernet cable from this interface to your machine or by connecting to a Wi-Fi access point. This case is the simplest to use, as FreedomBox automatically provides your machine with the necessary network configuration. Your machine will automatically connect to FreedomBox provided network and will be able to connect to the Internet given that FreedomBox can itself connect to the Internet.

Sometimes the above setup may not be possible because the hardware device may have only one network interface or for other reasons. Even in this case, your machine can still connect to the Internet via FreedomBox. For this to work, make sure that the network interface that your machine is connecting to is in `internal` mode. Then, connect your machine to network in which FreedomBox is present. After this, in your machine's network configuration, set FreedomBox's IP address as the gateway. FreedomBox will then accept your network traffic from your machine and send it over to the Internet. This works because network interfaces in `internal` mode are configured to `masquerade` packets from local machines to the Internet and receive packets from Internet and forward them back to local machines.

6.11.3 Customization

The above default configuration may not be fit for your setup. You can customize the configuration to suit your needs from the `Networks` area in the 'setup' section of the FreedomBox web interface.

6.11.3.1 PPPoE connections

If your ISP does not provide automatic network configuration via DHCP and requires you to connection via PPPoE. To configure PPPoE, remove any network connection existing on an interface and add a PPPoE connection. Here, optionally, provide the account username and password given by your ISP and activate the connection.

6.11.3.2 Connect to Internet via Wi-Fi

By default Wi-Fi devices attached during first boot will be configured as access points. They can be configured as regular Wi-Fi devices instead to connection to a local network or an existing Wi-Fi router. To do this, click on the Wi-Fi connection to edit it. Change the mode to `Infrastructure` instead of `Access Point` mode and `IPv4 Addressing Method` to `Automatic (DHCP)` instead of `Shared` mode. Then the SSID provided will mean the Wi-Fi network name you wish to connect to and passphrase will be the used to while making the connection.

6.11.3.2.1 Problems with Privacy Feature

NetworkManager used by FreedomBox to connect to the Wi-Fi networks has a privacy feature that uses a different identity when scanning for networks and when actually connecting to the Wi-Fi access point. Unfortunately, this causes problems with some routers that reject connections from such devices. Your connection won't successfully activate and disconnect after trying to activate. If you have control over the router's behaviour, you could also turn off the feature causing problem. Otherwise, the solution

is to connect with a remote shell using SSH or Cockpit, editing a file `/etc/NetworkManager/NetworkManager.conf` and adding the line `wifi.scan-rand-mac-address=no` in the `[device]` section. This turns off the privacy feature.

Edit a file:

```
$ sudo nano /etc/NetworkManager/NetworkManager.conf
```

Add the following:

```
[device]
wifi.scan-rand-mac-address=no
```

Then reboot the machine.

6.11.3.3 Adding a new network device

When a new network device is added, network manager will automatically configure it. In most cases this will not work to your liking. Delete the automatic configuration created on the interface and create a new network connection. Select your newly added network interface in the add connection page.

- Then set firewall zone to `internal` and `external` appropriately.
- You can configure the interface to connect to a network or provide network configuration to whatever machine connects to it.
- Similarly, if it is a Wi-Fi interface, you can configure it to become a Wi-FI access point or to connect to an existing access points in the network.

6.11.3.4 Configuring a mesh network

FreedomBox has rudimentary support for participating in BATMAN-Adv based mesh networks. It is possible to either join an existing network in your area or create a new mesh network and share your Internet connection with the rest of the nodes that join the network. Currently, two connections have to be created and activated manually to join or create a mesh network.

6.11.3.4.1 Joining a mesh network

To join an existing mesh network in your area, first consult the organizers and get information about the mesh network.

1. Create a new connection, then select the connection type as *Wi-Fi*. In the following dialog, provide the following values:

Field Name	Example Value	Explanation
Connection Name	Mesh Join - BATMAN	The name must end with 'BATMAN' (uppercase)
Physical Interface	wlan0	The Wi-Fi device you wish to use for joining the mesh network
Firewall Zone	External	Since you don't wish that participants in mesh network to use internal services of FreedomBox
SSID	ch1.freifunk.net	As provided to you by the operators of the mesh network. You should see this as a network in *Nearby Wi-Fi Networks*
Mode	Ad-hoc	Because this is a peer-to-peer network
Frequency Band	2.4Ghz	As provided to you by the operators of the mesh network
Channel	1	As provided to you by the operators of the mesh network

BSSID	12:CA:FF:EE:BA:BE	As provided to you by the operators of the mesh network
Authentication	Open	Leave this as open, unless you know your mesh network needs it be otherwise
Passphrase		Leave empty unless you know your mesh network requires one
IPv4 Addressing Method	Disabled	We don't want to request IP configuration information yet

Save the connection. Join the mesh network by activating this newly created connection.

2. Create a second new connection, then select the connection type as *Generic*. In the following dialog, provide this following values:

Field Name	Example Value	Explanation
Connection Name	Mesh Connect	Any name to identify this connection
Physical Interface	bat0	This interface will only show up after you successfully activate the connection in first step
Firewall Zone	External	Since you don't wish that participants in mesh network to use internal services of FreedomBox
IPv4 Addressing Method	Auto	Mesh networks usually have a DHCP server somewhere that provide your machine with IP configuration. If not, consult the operator and configure IP address setting accordingly with *Manual* method

Save the connection. Configure your machine for participation in the network by activating this connection. Currently, this connection has to be manually activated every time you need to join the network. In future, FreedomBox will do this automatically.

You will now be able reach other nodes in the network. You will also be able to connect to the Internet via the mesh network if there is an Internet connection point somewhere in mesh as setup by the operators.

6.11.3.4.2 Creating a mesh network

To create your own mesh network and share your Internet connection with the rest of the nodes in the network:

1. Follow the instructions as provided above in step 1 of *Joining a mesh network* but choose and fix upon your own valid values for *SSID* (a name for you mesh network), *Frequency Band* (usually 2.4Ghz), *Channel* (1 to 11 in 2.4Ghz band) and *BSSID* (a hex value like 12:CA:DE:AD:BE:EF). Create this connection and activate it.

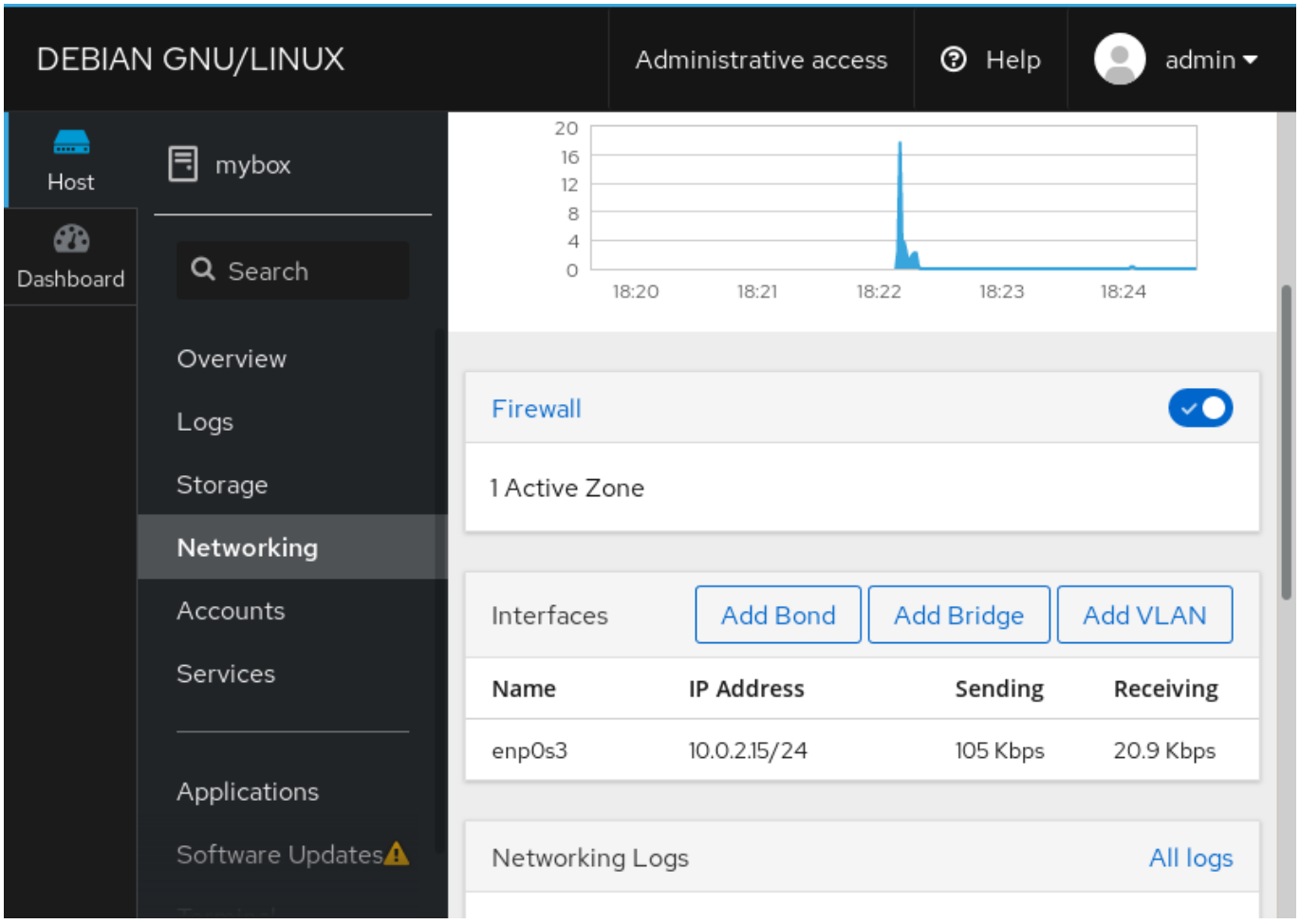
2. Follow the instructions as provided above in step 2 of *Joining a mesh network* but select *IPv4 Addressing Method* as *Shared*. This will provide automatic IP configuration to other nodes in the network as well as share the Internet connection on your machine (achieved using a second Wi-Fi interface, using Ethernet, etc.) with other nodes in the mesh network.

Spread the word about your mesh network to your neighbors and let them know the parameters you have provided when creating the network. When other nodes connect to this mesh network, they have to follow steps in *Joining a mesh network* but use the values for *SSID*, *Frequency Band* and *Channel* that you have chosen when you created the mesh network.

6.11.4 Advanced Network Operations

Cockpit provides many advanced networking features over those offered by FreedomBox. Both FreedomBox and Cockpit operate over Network Manager and are hence compatible with each other. Some of the functions provided by Cockpit include:

- Set the maximum transmission unit (MTU) for a network connection
- Change the hardware address (MAC address) of a network interface
- Add more DNS servers and configure routing of a network connection
- Creating bonded devices for highly available network interfaces
- Creating bridge devices to join network interfaces for aggregating separate networks
- Manage VLAN for creating virtual partitions in the physical network



6.11.5 Manual Network Operation

FreedomBox automatically configures networks by default and provides a simplified interface to customize the configuration to specific needs. In most cases, manual operation is not necessary. The following steps describe how to manually operate network configuration in the event that a user finds FreedomBox interface to insufficient for task at hand or to diagnose a problem that FreedomBox does not identify.

On the command line interface:

For text based user interface for configuring network connections:

```
nmtui
```

To see the list of available network devices:

```
nmcli device
```

To see the list of configured connections:

```
nmcli connection
```

To see the current status of a connection:

```
nmcli connection show '<connection_name>'
```

To see the current firewall zone assigned to a network interface:

```
nmcli connection show '<connection_name>' | grep zone
```

or

```
firewall-cmd --zone=internal --list-all
firewall-cmd --zone=external --list-all
```

To create a new network connection:

```
nmcli con add con-name "<connection_name>" ifname "<interface>" type ethernet
nmcli con modify "<connection_name>" connection.autoconnect TRUE
nmcli con modify "<connection_name>" connection.zone internal
```

To change the firewall zone for a connection:

```
nmcli con modify "<connection_name>" connection.zone "<internal|external>"
```

For more information on how to use `nmcli` command, see its man page. Also for a full list of configuration settings and type of connections accepted by Network Manager see:

https://developer.gnome.org/NetworkManager/stable/ref-settings.html

To see the current status of the firewall and manually operate it, see the Firewall section.

6.12 PageKite (Public Visibility)

6.12.1 What is PageKite?

PageKite makes local websites and services publicly accessible immediately without creating yourself a public IP address. It does this by tunneling protocols such as HTTPS or SSH through firewalls and NAT. Using PageKite requires an account on a PageKite relay service. One such service is https://pagekite.net.

A PageKite relay service will allow you to create kites. Kites are similar to domain names, but with different advantages and drawbacks. A kite can have a number of configured services. PageKite is known to work with HTTP, HTTPS, and SSH, and may work with some other services, but not all.

6.12.2 Using PageKite

1. Create an account on a PageKite relay service.
2. Add a kite to your account. Note your kite name and kite secret.
3. In FreedomBox, go to the "Configure PageKite" tab on the Public Visibility (PageKite) page.

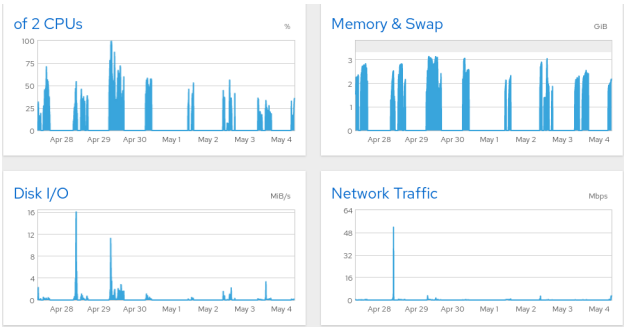
4. Check the "Enable PageKite" box, then enter your kite name and kite secret. Click "Save settings".
5. On the "Standard Services" tab, you can enable HTTP and HTTPS (recommended) and SSH (optional).
   - HTTP is needed to obtain the Let's Encrypt certificate. You can disable it later.
6. On the Certificates (Let's Encrypt) page, you can obtain a Let's Encrypt certificate for your kite name.

6.13 Performance (System Monitoring)

Available since: version 20.9.7

Performance app allows you to collect, store and view information about utilization of the hardware. This can gives you basic insights into usage patterns and whether the hardware is overloaded by users and services.

Performance metrics are collected by Performance Co-Pilot and can be viewed using the Cockpit app. When this system app is installed and enabled, cockpit's graphs shows the past (up to one year at a time).



6.14 Power

To restart or shut down FreedomBox, click the user dropdown menu on the top right of the page. After you select "Restart" or "Shut Down", you will be asked to confirm.

6.15 Secure Shell (SSH) Server

6.15.1 What is Secure Shell?

FreedomBox runs `openssh-server` server by default allowing remote logins from all interfaces. If your hardware device is connected to a monitor and a keyboard, you may login directly as well. Regular operation of FreedomBox does not require you to use the shell. However, some tasks or identifying a problem may require you to login to a shell.

6.15.2 Setting Up A User Account

6.15.2.1 FreedomBox First Log In: Admin Account

When creating an account in FreedomBox's web interface for the first time, this user will automatically have administrator capabilities. `Admin` users are able to log in using ssh (see Logging In below) and have superuser privileges via sudo.

6.15.2.2 Default User Account

- Note: If you can access FreedomBox's web interface, then you don't need to do this. You can use the user account created in FreedomBox's web interface to connect to SSH.

The pre-built FreedomBox images have a default user account called "fbx". However the password is not set for this account, so it will not be possible to log in with this account by default.

There is a script included in the freedom-maker program, that will allow you to set the password for this account, if it is needed. To set a password for the "fbx" user:

1. Decompress the image file.
2. Get a copy of freedom-maker from https://salsa.debian.org/freedombox-team/freedom-maker/.
3. Run `sudo ./bin/passwd-in-image <image-file> fbx`.
4. Copy the image file to SD card and boot device as normal.

The "fbx" user also has superuser privileges via sudo.

6.15.3 Logging In

6.15.3.1 Who can log in to FreedomBox by SSH?

FreedomBox administrative users may use SSH to to log in to FreedomBox. The user 'fbx' is created by FreedomBox and is an administrative super-user. There are options which allow ordinary users to log in:

- SSH access can be granted to specific users in the Edit User page by selecting the option, "Remotely login using Secure Shell (SSH) (freedombox-ssh)"
- SSH access can be granted globally to all users in the SSH configuration page by selecting the, "Allow all users to login remotely," option.

With a new FreedomBox you may log in as fbx using ssh, and other ordinary users will be able to log in after adjusting the user or Secure Shell settings above in this section. The root user account will have no password set and will not be able to log in.

6.15.3.2 SSH Client Software

SSH client in included in many operating systems including Linux, Microsoft Windows, and Apple MacOS. SSH is included in Chromebooks, but requires some configuration by the user. In most cases you can run SSH from a terminal or command prompt as shown here, using your FreedomBox hostname or IP address:

```
$ ssh freedombox.local
```

If your client computer does not have SSH available, PuTTY is a popular free software client program which complies with the Debian Free Software Guidelines. PuTTY has a graphical interface to remember and manage your SSH connections. See External links below for more information about PuTTY.

6.15.3.2.1 Cockpit as an SSH Alternative

The Cockpit Server Administration Terminal app available from the Cockpit Tools menu is an alternative shell access tool to SSH. Like SSH your connection to a FreedomBox terminal is secured. Cockpit is a good choice for users who do not wish to enable the SSH server or those who prefer to connect through a web browser. With either tool you will be presented with the FreedomBox bash command line interface.

Some users prefer to run SSH instead of, or in addition to, Cockpit. Command shell users tend to like SSH because it's something that they are already using. Users with Linux or Unix system administration experience tend to rely on this connection method because it is a simpler service which is thought to be more likely to be available if problems arise.

Refer to the Let's Encrypt and Cockpit sections of this manual to configure Cockpit and SSL certificates for security.

6.15.3.3 SSH over Local Network

To login via SSH, to your FreedomBox:

```
$ ssh fbx@freedombox.local
```

Replace `fbx` with the name of the user you wish to login as. `fbx` and users in `admin` group will be able to login on the terminal directly. Other users will be denied access.

`freedombox` should be replaced with the hostname. Alternatively, you can substitute the hostname by its IP address as found in the Quick Start process:

```
$ ssh fbx@192.168.1.1
```

If your FreedomBox has a domain name you can also use it:

```
$ ssh fbx@myfreedombox.freedombox.rocks
```

If you repeatedly try to login as a user and fail, you will be blocked from logging in for some time. This is due to `libpam-abl` package that FreedomBox installs by default. To control this behavior consult `libpam-abl` documentation.

Here we've used a .local network name (using multicast DNS), a local network IP address, and a DNS name to connect to FreedomBox using SSH.

6.15.3.4 SSH over the Internet

If your router is configured accordingly or your FreedomBox is directly exposed to the internet, you can also use a public domain name or a public IP address in the same fashion we'd do for the local network. Multicast DNS won't work here, though.

Let's look at connecting by SSH to FreedomBox using other networks now.

6.15.3.5 SSH over Tor

If in FreedomBox you have enabled onion services via Tor, you can access your FreedomBox using ssh over Tor. On a GNU/Linux computer, install netcat-openbsd.

```
$ sudo apt-get install netcat-openbsd
```

Edit ~/.ssh/config to enable connections over Tor.

```
$ nano ~/.ssh/config
```

Add the following:

```
Host *.onion
  user USERNAME
  port 22
  ProxyCommand nc -X 5 -x 127.0.0.1:9050 %h %p
```

Replace USERNAME with, e.g., an `admin` username (see above).

Note that in some cases you may need to replace 9050 with 9150.

Now to connect to the FreedomBox, open a terminal and type:

```
$ ssh USERNAME@ADDRESS.onion
```

Replace USERNAME with, e.g., an `admin` username, and ADDRESS with the onion service address for your FreedomBox.

6.15.3.6 SSH Over Pagekite

If in FreedomBox you are using Pagekite to expose services to the Internet, you can access your FreedomBox using SSH over Pagekite. On a GNU/Linux computer install netcat-openbsd.

```
$ sudo apt-get install netcat-openbsd
```

Edit ~/.ssh/config to enable connections over Pagekite.

```
$ nano ~/.ssh/config
```

Add the following:

```
Host *.pagekite.me
  CheckHostIP no
  ProxyCommand /bin/nc -X connect -x %h:443 %h %p
```

Now to connect to FreedomBox, open a terminal and type:

```
$ ssh USERNAME@KITENAME.pagekite.me
```

Replace USERNAME with, e.g., an `admin` username, and KITENAME with your kite name provided by pagekite.net as configured in FreedomBox.

6.15.4 Becoming Superuser

After logging in, if you want to become the superuser for performing administrative activities:

```
$ sudo su
```

Make a habit of logging in as root *only when you need to*. If you aren't logged in as root, you can't accidentally break everything.

6.15.5 Changing Password

To change the password of a user managed by FreedomBox's web interface, use the change password page. However, the `fbx` default user is not managed by FreedomBox's web interface and its password cannot be changed through it.

To change password on the terminal, log in to your FreedomBox as the user whose password you want to change. Then, run the following command:

```
$ passwd
```

This will ask you for your current password before giving you the opportunity to set a new one.

6.15.6 SSH Keys

The next step for SSH security and convenience is to understand and use ssh keys. If you logged in to FreedomBox the first time using ssh following the instructions above you specified a username and password to log in. In this section you'll learn about Server Fingerprints and host keys, authorized keys, and reasons to use these to make connection easier and more secure.

By default SSH is configured to prefer to use keys while still allowing you to use a username and password to log in. At the end of this section you will be able to:

- Connect to FreedomBox and know that you are connecting to the right computer.
- Connect instantly without giving a username and password.
- Further improve the security of your FreedomBox by disabling SSH password authentication.

6.15.6.1 SSH Public and Private Keys

SSH keys are generated in pairs called a key pair. There is a public key and a private key for each key pair. The public key encrypts data which can only be read using the private key, and the private key encrypts data which can only be read using the public key. This is called an asymmetric cryptography system. SSH will distribute your public keys automatically to the other connected system while keeping your private keys safe.

Using SSH keys creates a powerful set of security features:

- You are assured that you are connected to your FreedomBox.
- Nobody will be able to read or modify your ssh communication to FreedomBox.
- The FreedomBox SSH server will know you are the remote user connected.
- Nobody will be able to read or modify your ssh communication from FreedomBox.
- Connection is automatic with no username or password.
- Your FreedomBox can block any password guessing attack.

6.15.6.2 Create your personal SSH keys on your client computer using ssh-keygen

You will create an SSH key pair on your client computer. We'll use the defaults and will not specify a password. Just press the Enter key when you are prompted for an SSH key password. This is very simple using the ssh-keygen command with no arguments. Here is how to run the command and a sample of the output the ssh-keygen program will give to you:

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/username/.ssh/id_rsa):
Created directory '/home/username/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/username/.ssh/id_rsa
Your public key has been saved in /home/username/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:nHcTP5DBKxBOgt8BFMyb2QUs//t8ge+8vw2zjOuE71U username@clientpc
The key's randomart image is:
+---[RSA 3072]----+
|     ==++o ..    |
|    . +++ . .o   |
|     . O.+  +.   |
|      =.+.. .+   |
|        S...o.o E|
|         ..o...o |
|          ....+. |
|          .+ =o+.|
|           +O+*++|
+----[SHA256]-----+
```

That's all you need to do. You now have a personal SSH key pair on your client computer.

6.15.6.3 Verify your FreedomBox Server Fingerprint

On your first time connecting to FreedomBox using ssh you may have noticed a message similar to this:

```
$ ssh fbx@freedombox.local
The authenticity of host 'freedombox.local (192.168.1.4)' can't be established.
ED25519 key fingerprint is SHA256:TwJFdepq7OaTXcycoYfYE8/lRtuOxUGCrst0K/RUh4E.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

There are a few things to understand about this message:

- SSH is telling you that you have never connected to this server before and SSH cannot guarantee that this server is safe for you to use.
- You have an opportunity to tell SSH that this new server is known to you and safe to use by indicating, 'yes.'
- SSH has received an encryption key to communicate securely with this server (even if we're not certain which with server we're communicating with).
- SSH is giving you a piece of information that you will use to confirm that the remote SSH server is in fact your FreedomBox.

Go to FreedomBox in your web browser. Click on the System menu, and then Secure Shell. The second section of this page is, Server Fingerprints. There is an ED25519 key entry on this page:

Algorithm	Fingerprint
RSA	SHA256:ZGvgdxiDEpGKdw82Z6z0QRmDpT3Vgi07Ghba5IBJ4tQ
ECDSA	SHA256:BLMMfPxNHpHF0sqCazAwE6ONdLtMY+W2yrgjP7AeX
ED25519	SHA256:TwJFdepq7OaTXcycoYfYE8/lRtuOxUGCrst0K/RUh4E

Compare the ED25519 fingerprint on the FreedomBox Secure Shell page with the ED25519 fingerprint received by the ssh client in the first-connection example above. If these fingerprints are the same then you may be confident that you are connecting to your FreedomBox.

If you'd like to walk through these steps but you have already made the first connection, you can reset that. Issue this command on your ssh client computer.

```
$ ssh-keygen -R freedombox.local
```

This removes the record of your known connection to FreedomBox. Now open your Secure Shell system configuration page on FreedomBox to the Server Fingerprints section. Next connect to FreedomBox with your ssh client and properly verify the server fingerprint before indicating yes to the host authenticity question. Having done this correctly you can be certain that when you make an SSH connection to FreedomBox you are connecting to your server.

Each time you connect to a new SSH server you will be given the opportunity to verify the server fingerprint. If you connect to FreedomBox using different names or IP address (local IP, DNS name, Pagekite name, TOR .onion address...) you will be asked once for each name or address, but the fingerprint will not change.

Your server fingerprints are not private information. The fingerprint is a summary of a public key shared by the server which is used encrypt information sent to the SSH server. Your server public key is also not private information. You could share fingerprints and public keys with the world and the security of your FreedomBox will not be diminished.

6.15.6.4 Share your personal SSH key with FreedomBox using ssh-copy-id

By now you have a personal key pair, and you have verified the identity of FreedomBox. FreedomBox still does not know about your identity, and will ask you for your password when you try to log in by ssh. The ssh-copy-id program will tell FreedomBox to accept your personal key as your password.

```
$ ssh-copy-id username@freedombox.local
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any  ↩
    that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it  ↩
    is to install the new keys
username@freedombox.local's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'username@freedombox.local'"
and check to make sure that only the key(s) you wanted were added.
```

This step adds your personal public key to your user account on FreedomBox. With this step complete the FreedomBox SSH server will compare the key sent by the client computer with the key stored on FreedomBox. If these match then you will be logged in without the need to give a password. Try it now:

```
$ ssh freedombox.local
Linux freedombox 6.1.0-18-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.76-1 (2024-02-01) x86_64

                          .--._    _.--.
                         (     \  /     )
                          \     /\     /
                           \_   \/   _/
                            /        \
                           (    /\    )
                            '--'  '--'

                            FreedomBox

FreedomBox is a pure blend of Debian GNU/Linux. Web interface is available at
https://localhost/ . FreedomBox manual is available in /usr/share/doc/freedombox
and from the web interface.

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
Last login: Sun Mar 17 14:27:03 2024 from 192.168.144.101
username@freedombox:~$
```

Once you have added your client SSH key to FreedomBox you will be able to connect using that one key by every method of addressing your FreedomBox:

- Local network name
- Local network IP address
- ISP Public IP address
- DNS name if you are using Dynamic DNS
- Pagekite name if you are using Pagekite
- TOR .onion address if you are using TOR

6.15.6.5 Block SSH password guessing attempts by disabling password authentication

Once you are able to connect to FreedomBox by ssh using a key and not entering a password you can take a step to improve the security of FreedomBox. If your FreedomBox is accessible from the internet you may notice that there are repeated attempts to log in to your FreedomBox from the internet. A good password is your first line of defense, and FreedomBox has additional features which protect you from these intrusion attempts. You can stop this nonsense completely by disabling password authentication for Secure Shell.

Go to your FreedomBox System menu. Click the Secure Shell configuration link. Look under **Configuration** and select, "Disable password authentication" [x] Disable password authentication

Click the, "Update setup," button and it's done. This will stop all password guessing intrusion attempts using ssh. You can log in using your key, and nobody else will be able to log in by guessing a password.

6.15.6.6 Remote Host Identification has Changed : What it means and what to do

You may eventually experience an alarming message when you try to log in to your FreedomBox with SSH. You will see a message similar to this.

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
SHA256:ZGvgdxiDEpGKdw82Z6z0QRmDpT3Vgi07Ghba5IBJ4tQ.
Please contact your system administrator.
Add correct host key in /home/username/.ssh/known_hosts to get rid of this message.
Offending RSA key in /home/username/.ssh/known_hosts:2
  remove with:
  ssh-keygen -f "/home/username/.ssh/known_hosts" -R "freedombox.freedombox.rocks"
Host key for freedombox.freedombox.rocks has changed and you have requested strict checking ↩
    .
Host key verification failed.
```

This message tells you something important. It's usually not threatening, but there is the possibility that an attack could be made on a computer or network which can also produce this. What's important is that you'll do the same thing in either case.

The nature of this message is that the trust relationship you make with the SSH server through the fingerprint verification and key exchange with ssh-copy-id has been broken. Reading this error message closely, the issue is that the key fingerprint sent by FreedomBox at connection time does not match the key stored on the SSH client at the time you did the fingerprint verification. This could mean a few different things:

- Your .ssh/known_hosts file on the client was modified or corrupted such that the stored fingerprint was altered.
- You generated new keys for the FreedomBox SSH server.
- Your network has changed such that when you attempt SSH to FreedomBox your connection goes elsewhere. This could happen if you change IP addresses on FreedomBox and SSH to the old IP address, or it could happen by way of a sophisticated network attack.

Fix this by removing the FreedomBox entry from the client computer. On your laptop or desktop do the command as **written exactly as in the the error message you receive** (don't copy one from the message above!).

```
$ ssh-keygen -f /home/username/.ssh/known_hosts -R "freedombox.freedombox.rocks"
# Host freedombox.freedombox.rocks found: line 2
# Host freedombox.freedombox.rocks found: line 3
/home/username/.ssh/known_hosts updated.
Original contents retained as /home/username/.ssh/known_hosts.old
```

In so doing you have removed the FreedomBox fingerprint verification step we've done. Go back to the *Verify your FreedomBox Server Fingerprint* section above and complete the steps again. For good measure, make an effort to see that you are connected to your own FreedomBox in case you are being attacked.

6.15.7 External links

- Debian SSH wiki: https://wiki.debian.org/SSH
- Upstream project: https://www.openssh.com
- User documentation: https://www.openssh.com/manual.html
- PuTTY Client Software: https://www.chiark.greenend.org.uk/~sgtatham/putty/

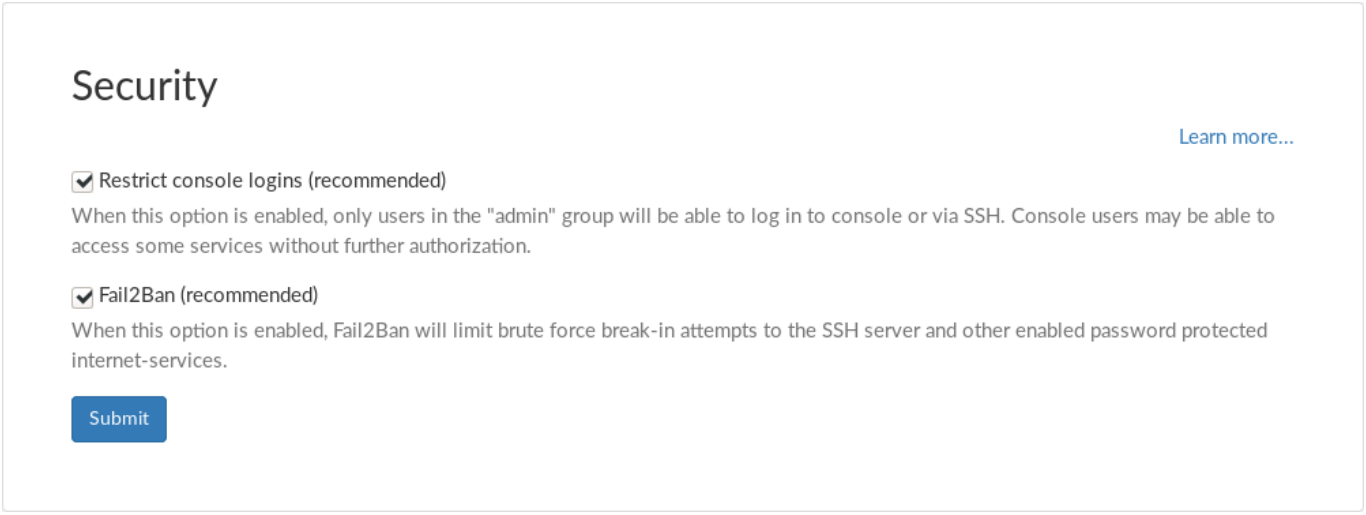
6.16 Security

Press the *Show security report* button to see a report including the following:

- Number of security vulnerabilities in installed version of FreedomBox.
- Number of security vulnerabilities for each installed app.
- Whether each installed app supports security sandboxing.
- For each enabled app, the security sandbox coverage as a percentage.

6.16.1 Configuration

When the *Restrict console logins* option is enabled, only users in the *admin* group will be able to log in via console, secure shell (SSH) or graphical login. When this option is disabled, any user with an account on FreedomBox will be able to log in. They may be able to access some services without further authorization. This option should only be disabled if all the users of the system are well trusted. If you wish to use your FreedomBox machine also as a desktop and allow non-admin users to login via GUI, this option must be disabled. You can define the list of users belonging to *admin* group in the Users section.



6.17 Service Discovery

Service discovery allows other devices on the network to discover your FreedomBox and services running on it. If a client on the local network supports mDNS, it can find your FreedomBox at <hostname>.local (for example: freedombox.local).

It also allows FreedomBox to discover other devices and services running on your local network.

Service discovery is not essential and works only on internal networks. It may be disabled to improve security especially when connecting to a hostile local network.

6.17.1 Troubleshooting

6.17.1.1 Unable to reach <hostname>.local

If <hostname>.local is not able to be reached, you may simply need to disable and re-enable the Service Discovery feature in FreedomBox. To do this, go to **System -> Service Discovery**, slide the toggle to the left position to disable it (it turns grey), followed by sliding it back to the right to re-enable it (it turns blue).

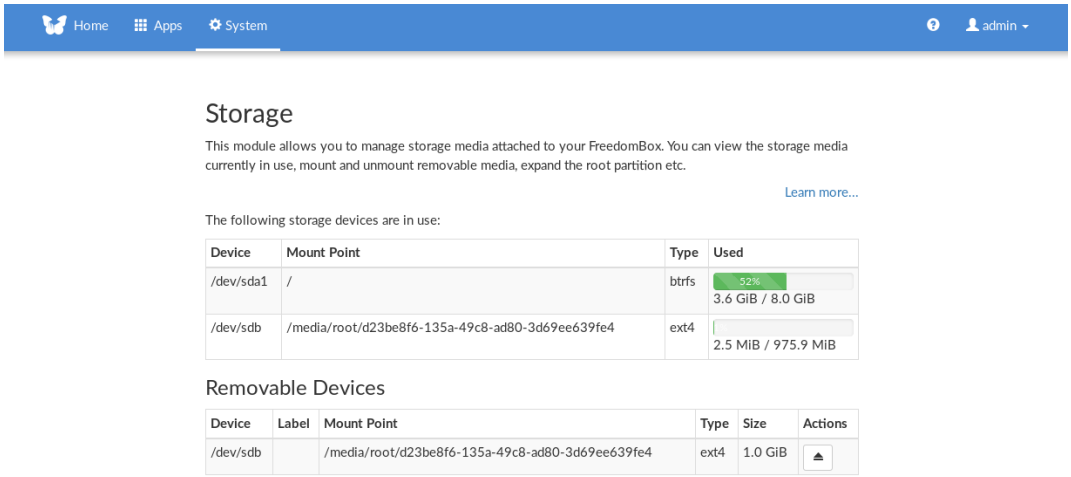
To do this you obviously need other means to reach your FreedomBox than <hostname>.local. See the Quick Start Guide for those.

6.18 Storage

Storage allows you to see the storage devices attached to your FreedomBox and their disk space usage.

FreedomBox can automatically detect and mount removable media like USB flash drives. They are listed under the *Removable Devices* section along with an option to eject them.

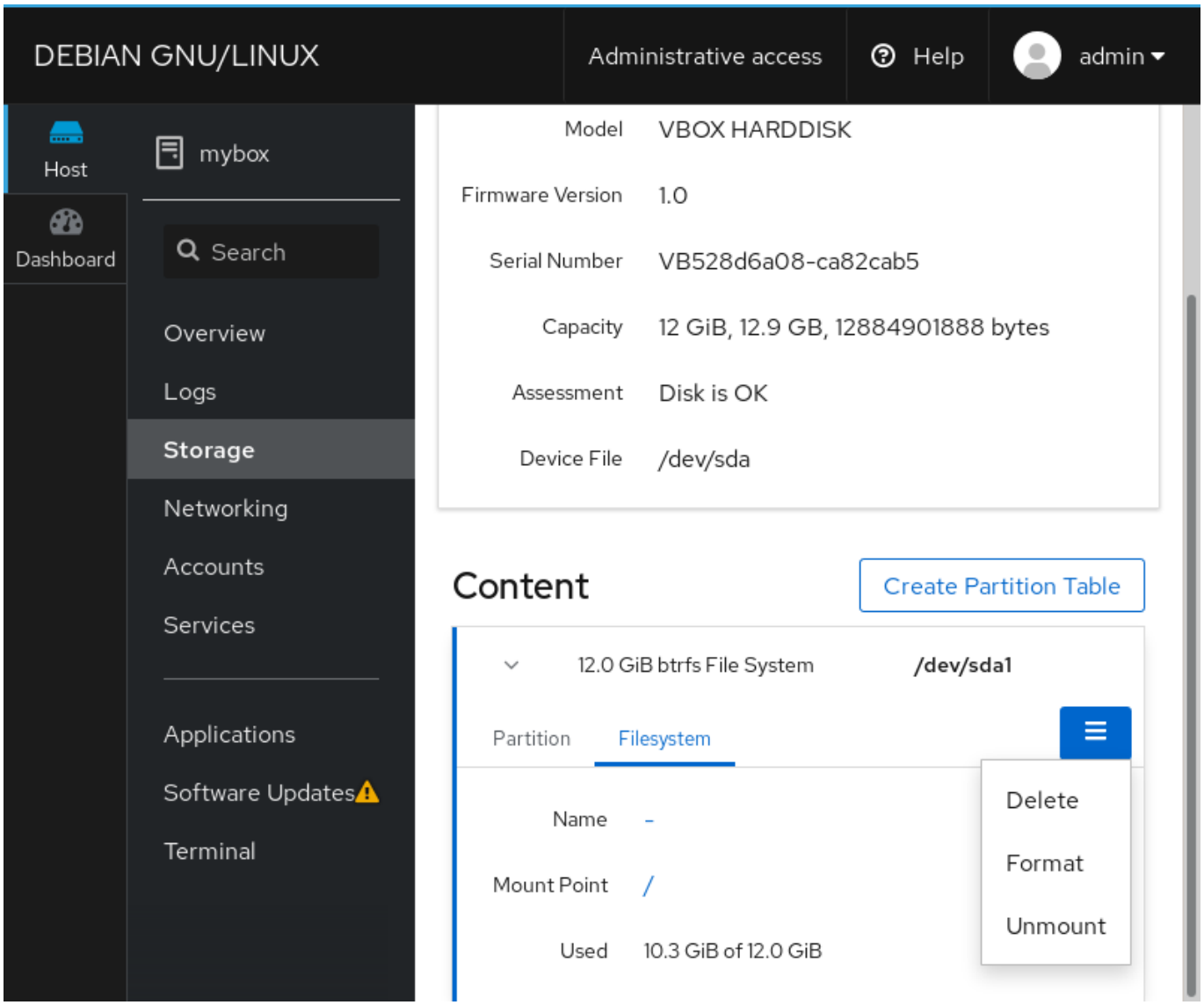
If there is some free space left after the root partition, the option to expand the root partition is also available. This is typically not shown, since expanding the root partition happens automatically when the FreedomBox starts up for the first time.



6.18.1 Advanced Storage Operations

Cockpit provides many advanced storage features over those offered by FreedomBox. Both FreedomBox and Cockpit operate over Udisks2 storage daemon and are hence compatible with each other. Some of the functions provided by Cockpit include:

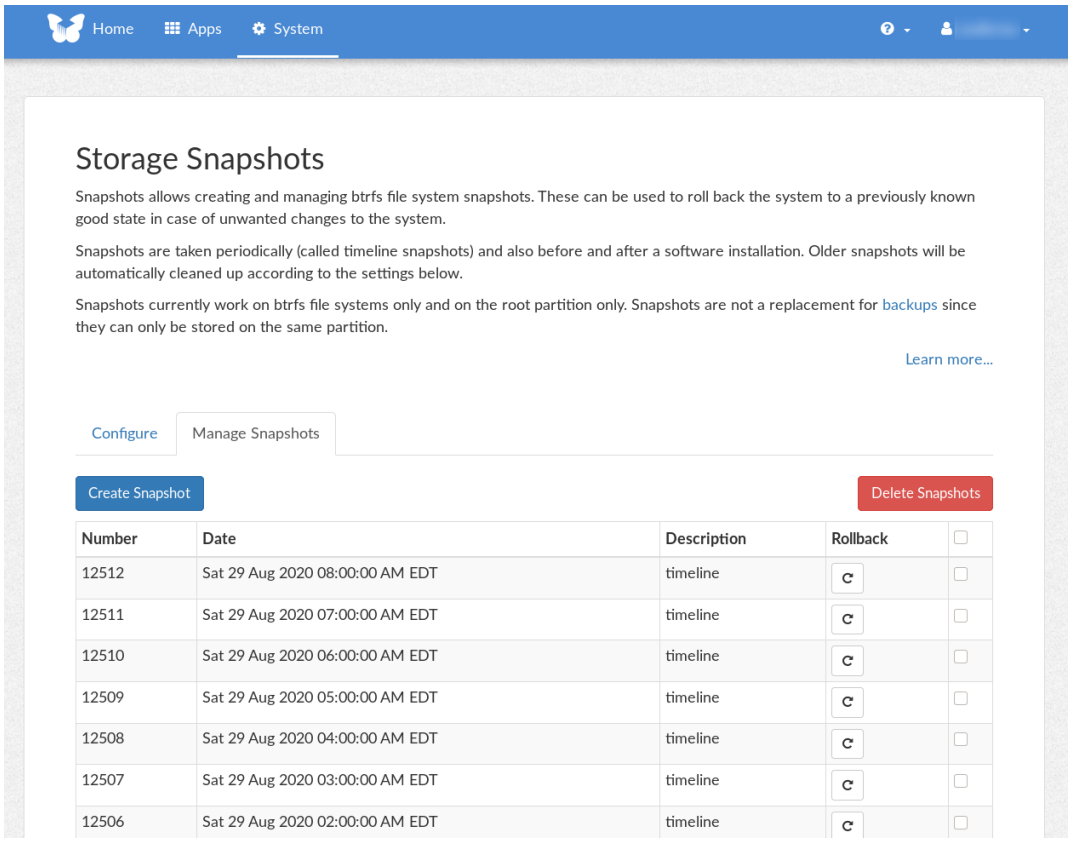
- Format a disk or partition with a fresh filesystem
- Add, remove partitions or wipe the partition table
- Create and unlock encrypted file systems
- Create and manage RAID devices



6.19 Storage Snapshots

Snapshots allows you to create filesystem snapshots, and rollback the system to a previous snapshot.

- Note: This feature requires a Btrfs filesystem. All of the FreedomBox stable disk images use Btrfs.



There are three types of snapshots:

- boot: Taken when the system boots up
- Software Installation (apt): Taken when software is installed or updated
- Timeline: Taken hourly, daily, weekly, monthly, or yearly

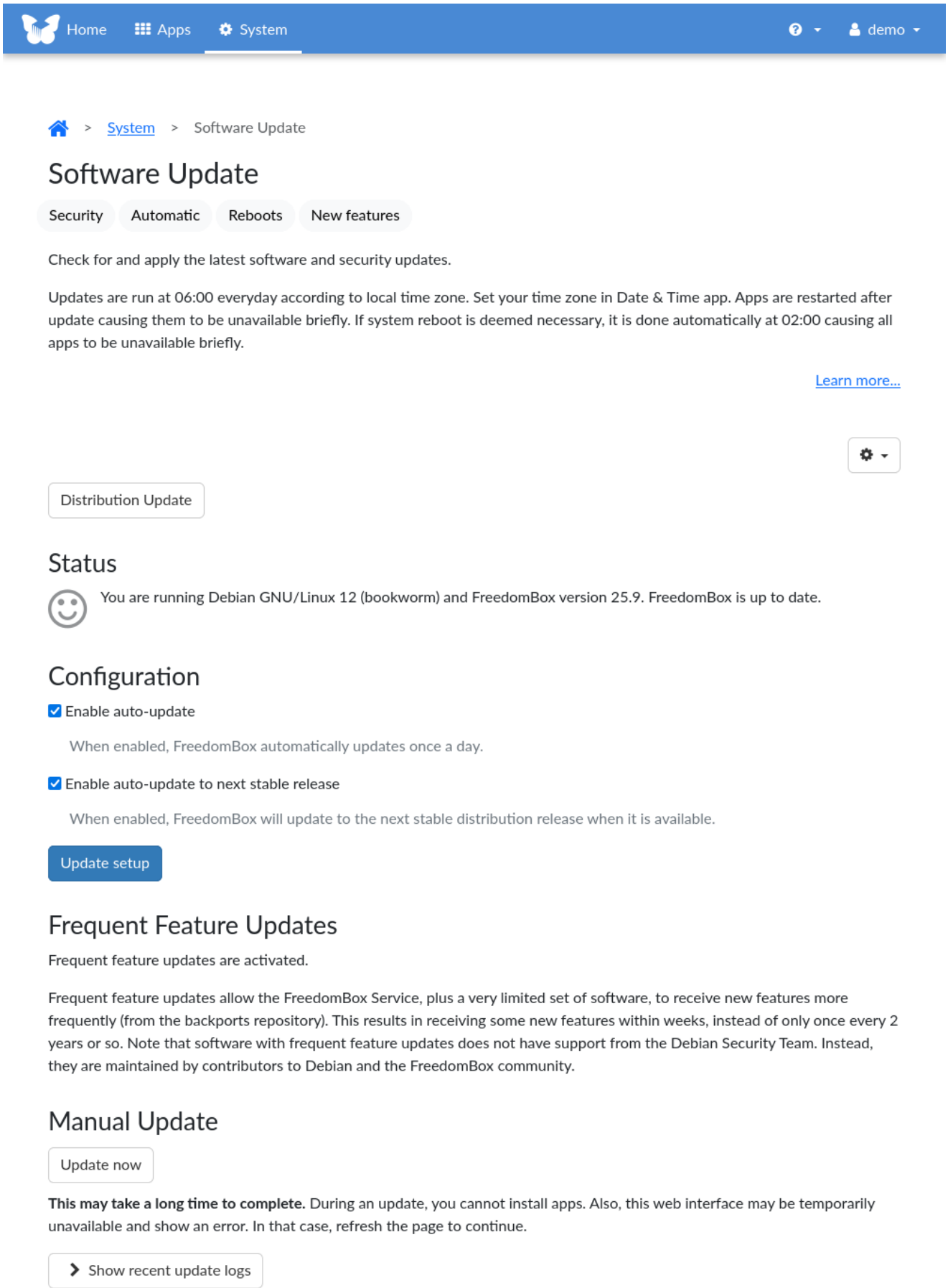
The Timeline and Software Installation snapshots can be turned on or off, and you can limit the number of each type of Timeline snapshot. You can also set a percentage of free disk space to be maintained.

6.20 Software Updates

FreedomBox can automatically install security updates. On the *Update* page of the *System* section in FreedomBox web interface you can turn on automatic updates. This feature is enabled by default and there is no manual action necessary. It is strongly recommended that you have this option enabled to keep your FreedomBox secure.

Updates are performed every day at night according to the time zone you have set on your machine. You can set the time zone with Date & Time. If you wish to shutdown FreedomBox every day after use, keep it running at night once a week or so as to allow automatic updates to happen. Alternatively, you can perform manual updates as described below.

Note that once the updates start, it may take a long time to complete. During automatic update process that runs every night or during manual update process, you will not be able to install apps from FreedomBox web interface.


Home
Apps
System
demo
System
Software Update
Software Update
Security
Automatic
Reboots
New features
Check for and apply the latest software and security updates.
Updates are run at 06:00 everyday according to local time zone. Set your time zone in Date & Time app. Apps are restarted after update causing them to be unavailable briefly. If system reboot is deemed necessary, it is done automatically at 02:00 causing all apps to be unavailable briefly.
Learn more...
Distribution Update
Status
You are running Debian GNU/Linux 12 (bookworm) and FreedomBox version 25.9. FreedomBox is up to date.
Configuration
Enable auto-update
When enabled, FreedomBox automatically updates once a day.
Enable auto-update to next stable release
When enabled, FreedomBox will update to the next stable distribution release when it is available.
Update setup
Frequent Feature Updates
Frequent feature updates are activated.
Frequent feature updates allow the FreedomBox Service, plus a very limited set of software, to receive new features more frequently (from the backports repository). This results in receiving some new features within weeks, instead of only once every 2 years or so. Note that software with frequent feature updates does not have support from the Debian Security Team. Instead, they are maintained by contributors to Debian and the FreedomBox community.
Manual Update
Update now
This may take a long time to complete. During an update, you cannot install apps. Also, this web interface may be temporarily unavailable and show an error. In that case, refresh the page to continue.
Show recent update logs

6.20.1 When Will I Get the Latest Features?

Although updates are done every day for security reasons, latest features of FreedomBox will not propagate to all the users. The following information should help you understand how new features become available to users.

Stable Users: This category of users include users who bought the FreedomBox Pioneer Edition, installed FreedomBox on a Debian stable distribution or users who downloaded the *stable* images from freedombox.org. If an app is available only in *testing* distribution but not in *stable* distribution, then that app will show up in the web interface but will not be installable by *stable* users. Every two years, a major release of Debian stable happens with the latest versions of all the software packages and FreedomBox developers will attempt to upgrade these users to the new release without requiring manual intervention. See the sections on upgrade to next stable release below.

As a general rule, only security updates to various packages are provided to these users. One exception to this rule is when *frequent feature updates* (recommended) option is enabled. When this option is enabled, FreedomBox service itself is updated when a release gains high confidence from developers. This means that latest FreedomBox features may become available to these users (after a few days of delay compared to *testing* users). Some apps (currently only Matrix Synapse) are also provided as an exception to the rule of "security updates only" when the app is severely broken or unavailable otherwise. Technically, when this option is enabled, after thorough testing, a select list of packages may be installed from *backports* or *unstable* repositories of Debian.

Testing Users: This category of users include users who installed FreedomBox on a Debian *testing* distribution or users who downloaded the *testing* images from freedombox.org. Users who use Debian *testing* are likely to face occasional disruption in the services and may even need manual intervention to fix the issue. As a general rule, these users receive all the latest features and security updates to all the installed packages. Every two weeks, a new version of FreedomBox is released with all the latest features and fixes. These releases will reach *testing* users approximately 2-3 days after the release.

Unstable Users: This category of users include users who installed FreedomBox on a Debian *unstable* distribution or users who downloaded the *unstable* images from freedombox.org. Users who use Debian *unstable* are likely to face occasional disruption in the services and may even need manual intervention to fix the issue. As a general rule, these users receive all the latest features to all the installed packages. Every two weeks, a new version of FreedomBox is released with all the latest features and fixes. Theses releases will reach *unstable* users on the day of the release. Only developers, testers and other contributors to the FreedomBox project should use the *unstable* distribution. End users are advised against using it.

6.20.2 Manual Updates from Web Interface

To get updates immediately and not wait until the end of the day, you may want to trigger updates manually. You can do this by pressing the *Update now* button under *Manual update* heading for *Software Update* page in *System* section. Note that this step is not necessary if you have enabled *Auto-updates* as every night this operation is performed automatically.

6.20.3 Manual Updates from Terminal

Some software packages may require manual interaction for updating due to questions related to configuration. In such cases, FreedomBox updates itself and brings in new knowledge necessary to update the package. Then, FreedomBox acts on behalf of the user and updates the packages by answering the questions. Until FreedomBox has a chance to update the package, such packages should not be be updated manually from the terminal. The manual update triggered from the web interface is already mindful of such packages and does not update them.

In some rare situations, FreedomBox itself might fail to update or the update mechanism might fall into a situation that might need manual intervention from a terminal. To perform manual upgrades on the terminal, login into FreedomBox on a terminal (if you have monitor and keyboard connected), via a web terminal (using Cockpit) or using a remote secure shell (see Secure Shell section). Then run the following commands:

```
$ sudo su -
Password: <enter user password here>
# dpkg --configure -a
# apt update
# apt -f install
# unattended-upgrade --debug
# apt install freedombox
# apt update
```

If `apt update` asks for a confirmation to change *Codename* or other release information, confirm *yes*. If during update of *freedombox* package, if a question about overwriting configuration files is asked, answer to install new configuration files from the latest version of the package. This process will upgrade only packages that don't require configuration file questions (except for *freedombox* package). After this, let FreedomBox handle the upgrade of remaining packages. Be patient while new releases of FreedomBox are made to handle packages that require manual intervention.

If you want to go beyond the recommendation to upgrade all the packages on your FreedomBox and if you are really sure about handling the configuration changes for packages yourself, run the following command:

```
$ apt full-upgrade
```

6.20.4 Auto-Update to Next Stable Release

FreedomBox can automatically update itself when there is a new stable release of Debian. This update feature is recommended, and enabled by default for *stable* systems. Note that it also requires "Enable auto-update" to be enabled, and there should be 5 GB of free space on the OS (root) partition.

In some special cases, such as advanced customization made to the system, the automatic update could fail. If you wish, you can disable it on the *System -> Software Update* page, by clearing the "Enable auto-update to next stable release" checkbox.

If you decide to stay on an older release, you should check Debian Releases and Long Term Support pages to see how long it will be supported by Debian security team. You stop using an older release if it stops receiving security updates. Note that older releases will not have new versions of FreedomBox (unless there is a security update), even through backports.

6.20.5 Manual Update to Next Stable Release

Thirty days after the release of a Debian stable release, all users will be upgraded to the next stable release. This behavior can be disabled as described above and users may choose to upgrade at a time for any the following reasons:

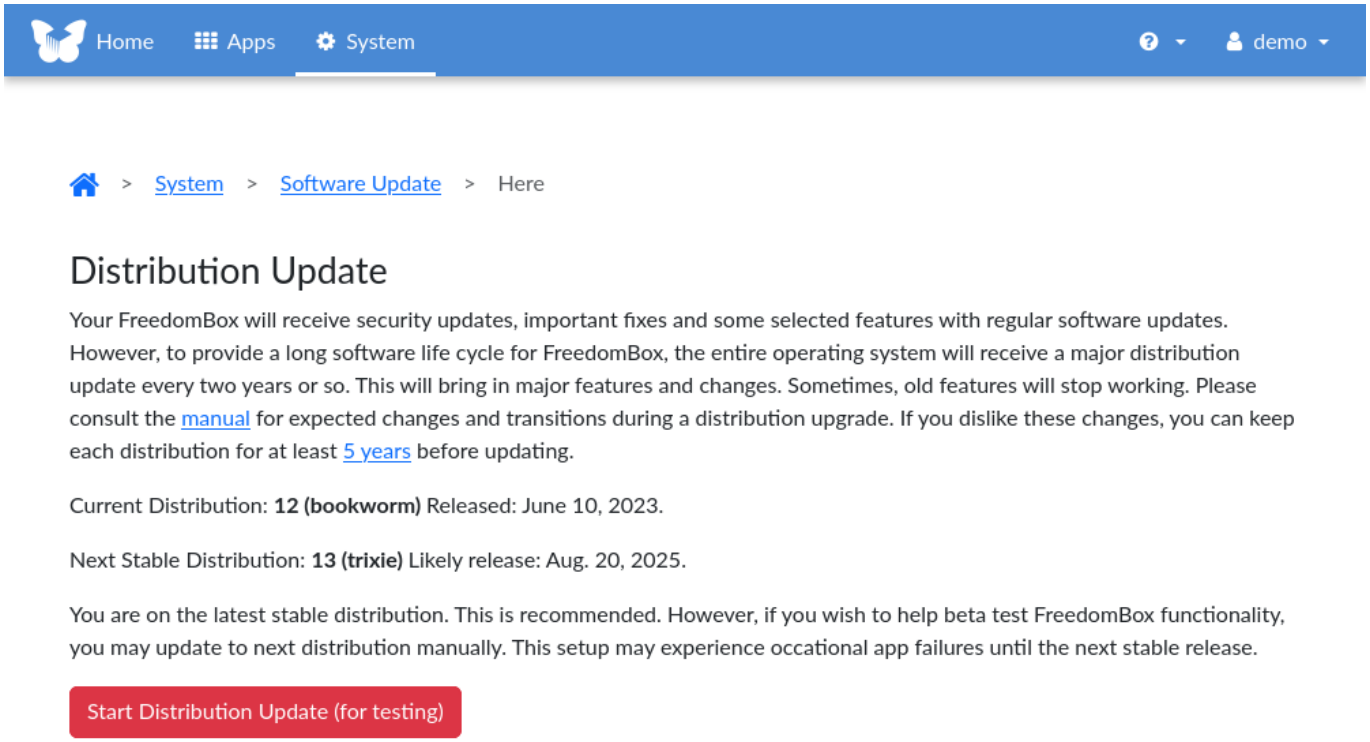
- If you don't want to wait 30 days and want to get the new stable release immediately.
- If you want to keep your working setup stable and not risk changing it (major OS upgrades are somewhat risky). You can potentially keep your distribution for years without change.
- If you want to minimize disruption during upgrade process by choosing a time convenient to you.
- If your favorite app is no longer supported in the new stable release and you will need time to migrate away to an alternative.
- If you want to take full backups before distribution upgrade happens and be on stand by to restore it if something goes wrong.
- If you want to test the upgrade process ahead of other users and report any problems to the development team.

Here are some tips before you begin manual update to next stable release:

- Take backups of your apps' data before performing the update. Making a copy of your entire disk using cloning software is even better.
- Create a system snapshot before you begin.
- General instructions for upgrading Debian distribution are available but this approach can't handle the configuration changes needed for the new versions of software.
- Some software requires additional update operations that FreedomBox performs as soon as it starts up after the distribution update. Please allow some time for these operations to complete after a distribution update. These apps include:
  - Postfix/Dovecot (requires a new configuration file format)
  - MediaWiki (requires database to be upgraded)
- Some packages are known to have prompts during upgrade, due to modified configuration files. It is recommended not to upgrade these packages manually, but rather to allow FreedomBox to handle their upgrade automatically. For upgrade to Debian 13 "trixie", this applies to the following packages:

  - Firewalld
  - Minidlna
  - Radicale
- Some packages may not be available after upgrade to next release. Please migrate to a similar application before or after the upgrade process. For upgrade to Debian 13 "trixie", the following apps are no longer available:
  - Janus. Use audio/video calling over Matrix or XMPP or audio calling over Mumble. This app will likely become available in future again.
  - TT-RSS. Migrate to Miniflux or Nextcloud News.
  - Searx.

To initiate the distribution update process, go to *Software Update* in *System* section. Then *Enable auto-update* and *Enable auto-update to next stable release* option. Then click on the *Distribution Update* button. Then click the *Start Distribution Update* button. The process may take several hours if your OS is on a slow disk such as an SD card. FreedomBox web UI may not be available during upgrade. Refresh the page if the page errors out. Finally, allow more time for post-upgrade processes and restart the machine.



6.21 Users and Groups

You can grant access to your FreedomBox for other users. Provide the Username with a password and assign a group to it. Currently the groups

- admin
- bit-torrent
- calibre
- ed2k
- feed-reader

- freedombox-share
- git-access
- minidlna
- syncthing
- web-search
- wiki

are supported.

The user will be able to log in to services that support single sign-on through LDAP, if they are in the appropriate group.

Users in the admin group will be able to log in to all services. They can also log in to the system through SSH and have administrative privileges (sudo).

A user's groups can also be changed later.

It is also possible to set an SSH public key which will allow this user to securely log in to the system without using a password. You may enter multiple keys, one on each line. Blank lines and lines starting with # will be ignored.

The interface language can be set for each user individually. By default, the language preference set in the web browser will be used.

A user's account can be deactivated, which will temporarily disable the account.

7 Guides

7.1 Expose a Local Service to Internet using FreedomBox

This guide describes to expose a local service you have on your LAN to the internet using a FreedomBox that is already exposed on the internet.

7.1.1 Goal

- *You have a FreedomBox running* with a working domain already setup. You are able to access your FreedomBox using a URL such as `https://www.mysite.example/` .
- *You have a different server running a local service* on LAN that you wish to expose to the Internet on the same domain as your FreedomBox. The local service must be a web service using the HTTP protocol (it is also possible to setup non-HTTP service, but that is a different approach). Optionally, you can have special domain assigned specifically for this local service.
- You wish for *FreedomBox to handle the TLS certificates* for this domain as FreedomBox typically does. Your local service need not setup and manage TLS certificates.
- Optionally, you may want to *limit access to local service using FreedomBox credentials*. Only users with a FreedomBox account (and belonging to a chosen group) will be able to access this service. The local service will not be available to general public on the Internet. This way you can host local services that don't implement their own authentication or local services with their own authentication mechanisms disabled.

7.1.2 Exposing Local LAN servers

- To expose the service, create an Apache configuration file in `/etc/apache2/conf-available/` and write a `ProxyPass` directive. Assuming your local service is running on a computer with IP address `192.168.0.20` on port `3000`, you can do this by running a command (as a root user on the terminal):

```
cat > /etc/apache2/conf-available/my-local-service.conf <<EOL
ProxyPass /myservice http://192.168.0.20:3000/
EOL
```

- Then, enable this Apache configuration. Run the following command:

```
a2enconf my-local-service
```

- Then reload Apache web server.

```
systemctl reload apache2
```

You can now access your local service at the following URL `https://www.mysite.example/myservice/` . Note that the URL is a secure URL with `https://`. The certificates will be managed by FreedomBox/Let's Encrypt. All your other apps and services work as usual. You can add any number services you want this way.

7.1.3 Running the Service on FreedomBox Itself

The above approach will also work if the local service is running on FreedomBox itself. In this case, the `ProxyPass` directive in the configuration should be modified to look like `ProxyPass /myservice http://127.0.0.1:3000/`.

7.1.4 Disabling the service

To disable exposing the local service:

1. Disable the Apache configuration file.

   ```
   a2disconf my-local-service
   ```

2. Then, reload Apache web server.

   ```
   systemctl reload apache2
   ```

7.1.5 Adding Authentication

You can configure the service to be available to only users with FreedomBox account. This is a good choice if:

- The local service that is exposed to the Internet should not be available to everyone. And,
- The local service does not provide its own authentication mechanism. Or,
- The local service provides it's own authentication but you don't want to maintain a different set of user accounts for that service and want to reuse your FreedomBox user accounts.
- Update the Apache configuration file `/etc/apache2/conf-available/my-local-service.conf` to look like the following.

```
<Location /myservice/>
    Include includes/freedombox-single-sign-on.conf

    ProxyPass http://192.168.0.20:3000/
</Location>
```

- Then, reload Apache web server.

```
systemctl reload apache2
```

If you wish for only some groups of users to be able to access the service, another change to the configuration is needed. Make the configuration look like the following if you wish to allow users of "web-search" group and also users of "admin" group to access the local service. All other users even if they have a valid FreedomBox account will be denied access.

```
<Location /myservice/>
    Include includes/freedombox-single-sign-on.conf

    <IfModule mod_auth_pubtkt.c>
        TKTAuthToken "web-search" "admin"
    </IfModule>

    ProxyPass http://192.168.0.20:3000/
</Location>
```

7.1.6 Hosting the Service on a Separate Domain

Some services and web applications hosted under a URL fragment such as `/myservice/` do not work well. They require an entire domain or subdomain to be dedicated to them. In these cases too, FreedomBox can expose local services.

1. First acquire a domain or a subdomain.
   - If your domain is a custom domain that you own, go to the DNS settings on your domain name provider and add a subdomain. You can create a CNAME record that simply points to the main domain.
   - If you are using FreedomBox Dynamic DNS service, then login into `https://ddns.freedombox.org` and enable the "Wildcard" option. With this change, for a domain (such as `example.fbx.one`), all subdomains (such as `myservice.example.fbx.one`) of your domain will point to your main domain.
2. Add the new domain into FreedomBox. Goto System → Name Services → Domain (regular) → Add. For example, add `myservice.mysite.example`.
3. Then create an Apache configuration file at `/etc/apache2/includes/<domainname>-include.conf`. For our example this is: `/etc/apache2/includes/myservice.mysite.example-include.conf`. The contents of the file should be:

   ```
   ProxyPass / http://192.168.0.20:3000/
   ```

4. Modify the above file for authentication if necessary by adding the `Include` and `<IfModule>` configuration directives as shown in the authentication section above.

7.1.7 Troubleshooting

When your setup does not work as expected, to debug, run a temporary web service on your FreedomBox.

1. Create a temporary directory with an index.html page on the FreedomBox.

   ```
   mkdir temp
   cd temp
   cat > index.html <<EOL
   <!DOCTYPE html>
   <html>
     <head>
       <title>My test page</title>
     </head>
     <body>
   ```

```
    <h1>Hello, World!</h1>
  </body>
</html>
EOL
```

2. Run a temporary web server (press Control-C to terminate):

```
python3 -m http.server 3000
```

3. Update the Apache configuration to point to this temporary service on FreedomBox.

```
ProxyPass /myservice http://127.0.0.1:3000/
```

7.1.8 Extras

There are many additional features that Apache can provide to improve the local service. For these, explore Apache documentation. Some features include:

- Serving static files such as JS, CSS, and media files from local directory without invoking the local service to improve performance.
- Caching the responses from the local service to improve performance.
- Setting additional headers for security, cache control, etc.
- Modify the HTTP header responses from the service when it does not work well to hosting under a URL fragment such as `/myservice/`.
- Perform redirects from `http://` URL to `https://` and setting HSTS header.
- Load balancing among multiple local services for high availability, scaling, etc.
- And more.

8 Hardware

FreedomBox is designed to be the software for a consumer electronics device that is easy to setup, maintain and use. The project does not aim to create a custom hardware device ourselves, but instead we intend to partner with hardware vendors to build FreedomBox devices and also support existing hardware. Typically, it is run on single board computers because of their small form factor, low power consumption and favourable price. Some users also run it on old/refurbished desktop or laptop computers or even on virtual machines running on their primary computers.

In addition to supporting various single board computers and other devices, any Debian machine can be turned into a FreedomBox by installing the `freedombox` package. Debian, the universal operating system, supports a much wider range on hardware. After installing Debian, see the manual page for installing FreedomBox on Debian.

8.1 Recommended Hardware

8.1.1 Libre Crafts FreedomBox

Libre Crafts in an endeavor from the FreedomBox developers themselves to bring you a powerful FreedomBox device capable of hosting even the most demanding home server needs. The device is crafted, tested, and delivered to you by FreedomBox developers. Your purchase helps FreedomBox development.

This hardware features a powerful CPU, plenty of main memory, a fast OS disk, ability to add two high capacity hard disk drives, dual multi-gigabit Ethernet ports, all with a low power consumption. Use it to host all your photos, to backup all home devices, as a NAS, as home automation hub, as a desktop computer, and more all at once.



Libre Crafts FreedomBox

8.1.2 Olimex's FreedomBox Pioneer Edition

On April 22nd, 2019, the FreedomBox Foundation announced the sales of the Pioneer Edition FreedomBox Home Server Kits. This pre-installed hardware is for all users who don't wish to build their own FreedomBox by choosing the right components, downloading the image and preparing an SD card with FreedomBox.

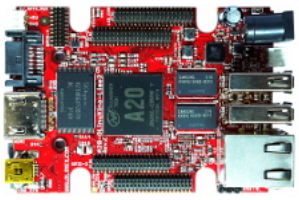
The kit includes all the hardware needed for launching a FreedomBox home server on an Olimex A20-OLinuXino-LIME2 board. This product provides the perfect combination of open source hardware and free and open source software. By purchasing this product, you also support the FreedomBox Foundation's efforts to create and promote its free and open source server software.



Pioneer Edition FreedomBox Home Server Kits

8.2 Supported Hardware

Use these hardware if you are able to download FreedomBox images and prepare an SD card by following the manual. If you wish for simper setup process, please buy the FreedomBox kits from recommended hardware instead. Look at the list of known issues with a hardware before buying it.



A20 OLinuXino Lime2



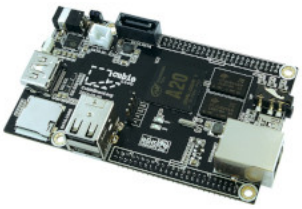
A20 OLinuXino MICRO



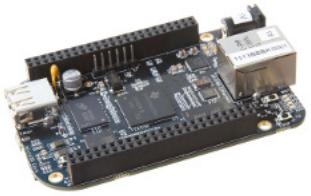
PC Engines APU



Cubietruck



Cubieboard2



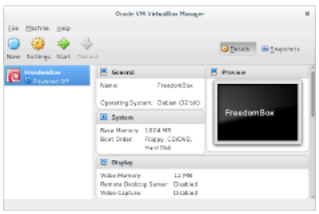
BeagleBone Black



pcDuino3



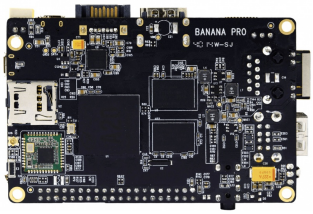
Debian



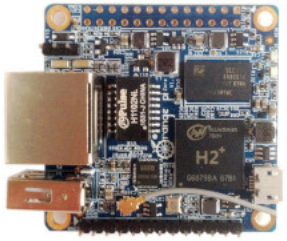
VirtualBox



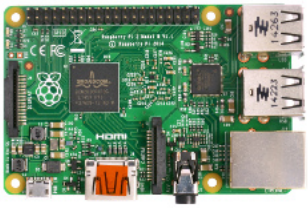
Pine A64+



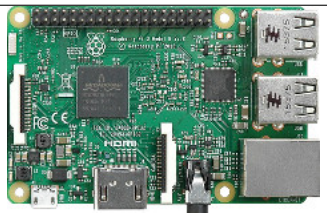
Banana Pro



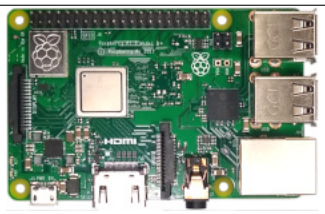
Orange Pi Zero



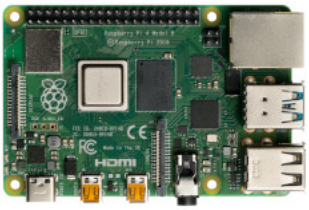
Raspberry Pi 2



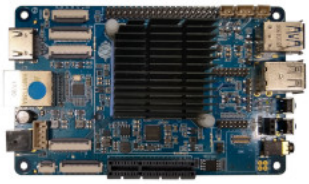
Raspberry Pi 3 Model B



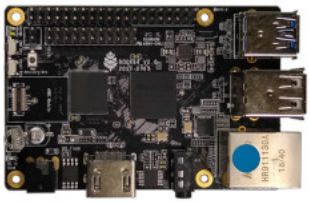
Raspberry Pi 3 Model B+



Raspberry Pi 4 B



RockPro64



Rock64

8.2.1 Hardware Comparison

Name	Speed (GHz)	Debian arch	Ram (GB)	disk (GB)	battery	SATA	Ethernet speed	OSHW
APU.1D	1x2	amd64	2	-	-	✔	1000x3	⛔
APU.1D4	1x2	amd64	4	-	-	✔	1000x3	⛔
BeagleBone Black C	1	armhf/omap	½	4	-	-	100	✔

Cubieboard2	1x2	armhf/sunxi	1	4	✔	✔	100	⛔
Cubieboard2-Dual	1x2	armhf/sunxi	1	-	✔	✔	100	⛔
Cubieboard3/Cubietruck	1x2	armhf/sunxi	2	8	✔	✔	1000	⛔
OLinuXino A20 LIME	1x2	armhf/sunxi	½	-	✔	✔	100	✔
OLinuXino A20 LIME2	1x2	armhf/sunxi	1	-	✔	✔	1000	✔
OLinuXino A20 MICRO	1x2	armhf/sunxi	1	-	✔	✔	100	✔
pcDunino3	1x2	armhf/sunxi	1	4	✔	✔	100	⛔
Pine A64+	1.2x4	arm64/sunxi	½,1,2	-	-	-	1000	⛔
Banana Pro	1.2x2	armhf/sunxi	1	-	-	✔	1000	⛔
Orange Pi Zero	?x4	armhf/sunxi	¼,½	-	-	-	100	⛔
RockPro64	1.4x4+1.8x2	arm64	2,4	16,32,64,128	-	(USB3 or via PCIe card)	1000	⛔
Rock64	1.5x4	arm64	1,2,4	16,32,64,128	-	(USB3)	1000	⛔

8.3 Additional Hardware

8.3.1 Hardware Supported with Generic Images

If you already have hardware that you wish turn into a FreedomBox, don't let the limited list of supported hardware discourage you. If you are using AMD or Intel architecture machines, you can download the generic images of that specific architecture that image will work on any machine of that architecture. For ARM 32-bit or ARM 64-bit architectures, we have a similar solution.

Starting with August 2020, we started building generic images that would work for all single board computers based on a solution involving UEFI standards and u-boot firmware. In this approach, a small board specific firmware resides on an SPI flash or an SD card. It is responsible for loading a generic FreedomBox image that is placed in an SD card, a USB drive, a SATA drive or an NVMe drive. So, for your hardware, find and get a u-boot based firmware from your board manufacturer and place it on an SPI flash or an SD card. Next, ensure that that kernel in FreedomBox has support for your board and place it on any of the other storage disks. This approach should work well for a lot of boards that are not listed as specifically supported. See firmware section for more details.

We continue to build images specific to some hardware as we used to earlier. These images have the slight advantage that they are easier to setup because of less step involved. We intend, however, to phase out these images because they can't be booted from all the storage devices and involve development overhead limiting the number of boards we support.

8.3.2 Adding Hardware Support

If your hardware is not listed above but you were able to get it working using the above described method of using a generic image, drop us a line and we will list it as supported. Further, take a look at the list of targeted hardware for boards to support.

8.3.3 Deprecated Hardware

This hardware was supported earlier but is no longer supported. If you downloaded an earlier image and are running FreedomBox on one of these hardware, you will keep getting software updates. However, no new images will be provided for these hardware. It is recommended that you migrate to newer, supported hardware using backup and restore.

- DreamPlug
- Raspberry Pi

Note: *Supported Hardware* means that FreedomBox images are built for said hardware and at least one developer has reported the basic functions to be working.

8.4 Common Hardware Information

The following sections document common advice related to hardware and peripherals when using them with FreedomBox.

8.4.1 Wi-Fi

FreedomBox can use Wi-Fi hardware for two separate purposes. It can be used to provide internet connectivity or it can be used to share internet connectivity already available to FreedomBox (via Ethernet, 3G/4G or another Wi-Fi interface) with devices on the network. See the Networks manual page for instructions on how to configure FreedomBox for these two cases.

Unfortunately, most built-in Wi-Fi adapters and add-on Wi-Fi adapters require firmware that is not free software. So, FreedomBox recommends attaching a USB Wi-Fi device that does not require non-free firmware. Supported devices automatically show up in the network interface list when configuring networks.

If you have a Wi-Fi device, either built-in or as an add-on, that requires non-free firmware and you are willing to install non-free firmware to get it working, see the Debian wiki page. Once the firmware is installed and the device shows up, it can be configured and used by FreedomBox.

8.4.2 Power Supply

On single board computers, one can easily encounter situations where the board and its peripherals are not provided sufficient power and malfunction in unpredictable ways. To avoid this, use a power adapter that can supply the minimum current recommended by the hardware manufacturer. When additional peripherals such as USB drives, Wi-Fi devices, SATA drives or NVMe drives are attached, the power requirements increase. A power supply that can provide higher current than needed is preferable but voltage should match the manufacturer recommendation exactly. Keep in mind that some cheap power supplies don't supply the current they promise to.

8.4.3 Firmware

Desktops, laptops and virtual machines have software that runs during machine start-up called UEFI/BIOS. This software, sometimes called firmware, can load and hand over control to the operating system (in our case FreedomBox), when it is present on any of the storage devices. This is not the case with most single board computers.

Single board computers ship with very small amount of software that is typically limited to booting OS from SD cards or eMMCs. They usually can't boot from USB disks, SATA disks or NVMe disks. To remedy this situation, hardware manufacturers started adding a special storage device called SPI flash which is only a few MiB in size. A special software, which we call firmware here, typically based on free and open source boot loader called u-boot is placed in this SPI flash. When the computer starts up, it starts the boot-loader from SPI flash which will in turn load the operating system. Since the firmware is much more powerful, it has the ability to load the OS from any of the storage media. Examples of single board computers with SPI flash include A20-OLinuXino-Lime2 and RockPro64.

This firmware approach can be used even when SPI flash is not available. Say, one wants to boot from a USB drive and the board does not support booting from it. Firmware can be installed on an SD card (a very tiny one is sufficient) and inserted into the board. Then USB disk will contain FreedomBox as we wish it. When the board starts, it boots the firmware from SD card which in turn boots the operating system from USB drive or any other storage.

This firmware approach also allows us to use generic download images that work for a large number of hardware boards. While increasing the effort for the user a bit more, it has the advantage of allowing us to support a lot more hardware and allow the OS to be present on any storage media.

When special firmware is needed for a single board computer, FreedomBox manual for the board discusses how to to obtain and install the firmware before proceeding with installation of FreedomBox.

8.4.4 Storage

FreedomBox can run from various storage media supported by your computer. Choosing the storage is about balancing reliability, capacity and speed against cost. A minimum storage capacity of 8GB is recommended for running FreedomBox.

8.4.4.1 Secure Digital (SD) Card

SD cards are common on single board computers. Most single board computers can boot directly from an SD card without any additional tweaks.

SD cards are typically slowest among the available storage media. Expect your FreedomBox to perform certain operations slower on these disks. Not all SD cards perform similarly and some perform much better than others. When buying an SD card, pick a card with a speed class of at least 10 (written on the card as a circle around the number 10) or UHS speed class 1 (written on the card as a number 1 inside a bucket). UHS speed class 3 (written on the card as number 3 inside a bucket) or application speed class 1 or above (written as A1 and A2) will perform much better. Finally, users of FreedomBox have reported cases where SD cards have failed. So, other storage media should be preferred for higher reliability.

8.4.4.2 Embedded MultiMediaCard (eMMC)

Many recently released single board computers support eMMC cards. Most single board computers can boot directly from an eMMC without any additional tweaks.

eMMC is sometimes soldered onto the board and you will need to choose the size of eMMC when buying the board. An example of this is the Olimex's A20-OLinuXino-Lime2 board. Other times, a manufacturer will provide eMMC as pluggable peripheral. With this approach, you can add eMMC after you buy the board or upgrade existing one with higher capacity. Do not detach and reattach such pluggable eMMCs too often. They have a very limited number of wear cycles (< 100).

eMMC are much faster than SD cards and spinning disk HDDs but are significantly slower than SSDs. They have much better random write speeds which are needed for many FreedomBox operations. In general, they should be preferred over SD cards.

FreedomBox image can be setup on an eMMC in two ways. For a detachable eMMC, there are eMMC to USB converters available. Detach the eMMC from the board, attach it to the USB converter and plug it into your machine and proceed with writing FreedomBox on it as one would for an SD card. In case the eMMC is not detachable, boot the computer with a media other than the eMMC such as an SD card or USB disk. It could be any operating system. After booting, the eMMC will show up as an additional disk. Download and write FreedomBox image onto it as one would for an SD card.

8.4.4.3 USB Disk Drive

Most computers and single board computers have USB ports. These ports accept storage media such as USB flash drives, SSDs or HDDs.

A USB flash drive can also serve as a storage medium for running FreedomBox. USB 2.0 flash drives are much slower and comparable to SD cards in their performance. USB 3.0 flash drives yield much better performance. Both USB flash drives and SD cards use similar technology so the read/write cycles and hence the reliability as similarly limited.

Apart from USB flash drives, solid state drives (SSDs) and hard disk drives (HDDs) can be inserted into USB ports. This is possible either by buying drives with USB interface or by using convertors such as USB to SATA or USB to M.2 interface. Both SSDs and HDDs have much higher reliability compared to SD cards, eMMC or USB flash drives. These should be preferred whenever possible. In addition, SSDs provide excellent performance when connected via USB 3.0 interface.

When connecting SSDs and HDDs to USB ports on single board computers, care should be taken about the power supply to the drive. If the drive has an extra power supply there is nothing to worry about. Otherwise, ensure that the single board computer is able to power the drive by checking the power requirements of the drive and what the board supports. For the board, always use a power adapter that can supply the minimum current recommended by the hardware manufacturer. Power supply that can provide higher current than needed is preferable but the voltage supplied should match the manufacturer recommendation exactly. Keep in mind that some cheap power supplies don't supply the current they promise to.

Setting up a FreedomBox image on a USB (flash, SSD or HDD) drive can be straight forward as most computers have USB ports. Plug-in the USB drive to your computer, download and write the FreedomBox image to the USB drive. While laptops, desktops and virtual machines can boot from a USB drive without intervention, many single board computers can't boot from USB drives. To address this, a separate firmware is needed. See firmware section for setting this up.

8.4.4.4 SATA disk drive

Some desktops, laptops and single board computers support a SATA interface to connect a solid state drive (SSD) or a hard disk drive (HDD). An example of a single board computer supporting SATA interface is the Olimex's A20-OLinuXino-Lime2. SATA protocol is also used for mSATA ports or M.2 slots (with a B-Key or an M-key). Both SSDs and HDDs have much higher reliability compared to SD cards, eMMC or USB flash drives. SATA interface provides very good data transfer rates (but not as good as NVMe drives based on PCIe). These should be preferred over SD cards, eMMCs or USB flash drives whenever possible.

When connecting SSDs and HDDs to SATA ports on single board computers, care should be taken about the power supply to the drive. If the drive has an extra power supply there is nothing to worry about. Otherwise, ensure that the single board computer is able to power the drive by checking the power requirements of the drive and what the board supports. Always use a power adapter that can supply the minimum current recommended by the hardware manufacturer. Power supply that can provide higher current than needed is preferable but voltage should match the recommendation exactly. Keep in mind that some cheap power supplies don't supply the current they promise to.

To setup FreedomBox image on a SATA disk drive, boot the computer with a media other than the SATA disk such as an SD card. It could be any operating system. After booting, the SATA disk will show up as an additional disk. Download and write FreedomBox image onto it as one would for an SD card. While laptops, desktops and virtual machines can boot from a SATA drives without additional intervention, many single board computers can't boot from SATA drives. To address this, a separate firmware disk is needed. See firmware section for setting this up.

8.4.4.5 NVMe disk drive

Most desktops, laptops and some single board computers support an NVMe interface to connect a solid state drive (SSD). This support is provided either with an M.2 slot (with a B-key or an M-key) or by providing a PCIe expansion slot. If a PCIe expansion slot is provided, a PCIe to M.2 convertor can be used to accommodate an NVMe drive. An example of a single board computer supporting an M.2 slot is the Radxa's Rock Pi 4 board. An example of single board computer providing PCIe slot is the Pine64's RockPro64 board. NVMe based SSD have much higher reliability compared to SD cards, eMMC or USB flash drives. NVMe drives provide the fastest data transfer rates. These should be preferred over all other types of drives whenever possible.

When connecting NVMe drives to single board computers, care should be taken about the power supply to the drive. Ensure that the single board computer is able to power the drive by checking the power requirements of the drive and what the board supports. Always use a power adapter that can supply the minimum current recommended by the hardware manufacturer. Power supply that can provide higher current than needed is preferable but voltage should match the manufacturer recommendation exactly. Keep in mind that some cheap power supplies don't supply the current they promise to.

To setup FreedomBox image on an NVMe disk drive, boot the computer with a media other than the NVMe disk such as an SD card. It could be any operating system. After booting NVMe disk will show up as an additional disk. Download and write FreedomBox image onto it as one would for an SD card. While laptops, desktops and virtual machines can boot from NVMe drives without intervention, many single board computers can't boot from NVMe drives. To address this a separate firmware disk is needed. See firmware section for setting this up.

8.5 Building Your Own Images

All FreedomBox disk images for different hardware is built by the project using a tool known as Freedom Maker. If for some reason, you wish to build your own images instead of downloading the provided images, use this tool. The README file in the project provides information about the list of hardware build targets available and how to build images.

8.5.1 Status of Software Used

- All the software present in FreedomBox images is from Debian repositories. There are some minor tweaks done by the Freedom Maker script.
- All images use the Linux kernel from Debian which is in turn based on the mainline Linux kernel.
- A few images include non-free firmware from Debian's **non-free-firmware** component (see Firmware). Other than firmware, all software present in the images is DFSG compliant free software. The table below lists the components included in each image. (Note: this work is in progress, planned for images built after Debian 12 (bookworm) release.)

Image	Includes main?	Includes non-free-firmware?	Non-free firmware included
32-bit ARM (armhf)	✓		
64-bit ARM (arm64)	✓		
64-bit x86 (amd64)	✓	✓	DebianPkg:amd64-microcode, DebianPkg:intel-microcode (see Microcode)
A20 OLinuXino Lime	✓		
A20 OLinuXino Lime 2	✓		
A20 OLinuXino MICRO	✓		
Beagle Bone Black	✓		
Cubieboard 2	✓		
Cubietruck (Cubieboard 3)	✓		
Lamobo R1	✓		
LeMaker Banana Pro	✓		
LinkSprite pcDuino3S	✓		
Orange Pi Zero	✓		
PC Engines APU 1D	✓		
Pine A64 LTS	✓		
Pine A64+	✓		
Pioneer Edition FreedomBox	✓		
QEMU/KVM amd64	✓		
Raspberry Pi 2	✓	✓	DebianPkg:raspi-firmware
Raspberry Pi 3 Model B	✓	✓	DebianPkg:raspi-firmware, DebianPkg:firmware-brcm80211
Raspberry Pi 3 Model B+	✓	✓	DebianPkg:raspi-firmware, DebianPkg:firmware-brcm80211
Raspberry Pi 4 Model B	✓	✓	DebianPkg:raspi-firmware, DebianPkg:firmware-brcm80211
Rock64	✓		
RockPro64	✓		
VirtualBox for amd64	✓		

8.6 Pioneer Edition FreedomBox

Pioneer FreedomBox Home Servers are produced and sold by Olimex, a company which specializes in Open Source Hardware (OSHW). The kit includes pocket-sized server hardware, an SD card with the operating system pre-installed, and a backup battery which can power the hardware for 4-5 hours in case of outages. It sells for 69 euro. An optional storage add-on for high capacity hard disk (HDD) or solid-state drive (SSD) is also available from Olimex. By purchasing this product, you also support the FreedomBox Foundation's efforts to create and promote its free and open source server software.



8.6.1 Product Features

The Pioneer Edition FreedomBox Home Server Kit includes all the hardware needed for launching a FreedomBox home server on an Olimex A20-OLinuXino-LIME2 board:

- the A20-OlinuXino-LIME2,
- a custom metal case with a laser-engraved FreedomBox logo,
- a high-speed 32GB micro SD card with the FreedomBox software pre-installed,
- a backup battery,
- a power adapter, and
- an Ethernet cable.
- an optional storage add-on for hard disk (HDD) or solid-state drive (SSD)

8.6.2 Recommended Hardware

This is a hardware recommended for all users who just want a turn-key FreedomBox, and **don't** want to **build** their own one.

(Building your own FreedomBox means some technical stuff like choosing and buying the right components, downloading the image and preparing the SD card).

8.6.3 Availability

The Pioneer Edition FreedomBox Home Server is the first commercially available version of FreedomBox.

- Price: 69 EUR
- Olimex Store
- The US version is also available through Mouser Electronics.

8.6.4 Hardware Specifications

Pioneer Edition FreedomBox Home Server is based on A20-OLinuXino-LIME2.

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3

- Storage: 32GB class 10+ microSD card pre-loaded with FreedomBox
- SATA: 1 SATA port 2.6 compliant 3Gb/s
- USB: 2 USB 2.0 Hi-Speed host ports
- Battery: 3.3V Li-Po, 1400mAh (4-5 hours of backup without additional devices connected via USB)
- Ethernet: 10/100/1000, RJ45 (1 meter cable included)
- Power adapter: 110-220 V input, 5V output, EU style (with optional UK or US sockets)
- Power consumption: 1.5W and 5W depending on load (0.3A to 1A current)
- Box: Custom metallic box with FreedomBox decal

Further information:

- Quick start leaflet
- Hardware source files
- A20-OLinuXino-LIME2 schematic
- A20 SOC datasheet

The kits run entirely on Free Software. They work with Kernel and u-boot from Debian repositories. Even the boot firmware in ROM called BROM is free software (GPLV2+).

8.6.5 Storage Add-on

You can order a storage add-on along with the Pioneer Edition FreedomBox Home Server. The storage add-on is a SATA disk drive enclosure case optionally with a hard disk or solid-state drive of size 128GB to 2000GB. If you have already purchased the Home Server without the add-on, you can order the add-on separately.

- Olimex Store
- Price: 9 EUR (without the hard disk, only for the case, you need to add your own HDD/SSD to it)
- Price: 42 EUR (with 128GB Solid-State Drive)
- Price: 69 EUR (with 512GB Solid-State Drive)
- Price: 42 EUR (with 320GB Hard Disk)
- Price: 53 EUR (with 500GB Hard Disk)
- Price: 64 EUR (with 1000GB Hard Disk)
- Price: 86 EUR (with 2000GB Hard Disk)

8.6.6 Download

The kits come with an SD card pre-loaded with FreedomBox. There's **NO need to download images**.

However, if you wish to reset your devices to a pristine state, then you can do so with the image provided. Follow the instructions on the download page to create a FreedomBox SD card and boot the device. Make sure to download the Pioneer Edition images. These SD card images are meant for use with the on-board SD card slot and won't work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.6.7 Build Image

FreedomBox images for this hardware can be built using Freedom Maker.

8.6.8 Known Issues

- The image that shipped with the kits uses a slightly modified u-boot from Debian and not stock Debian like the rest of FreedomBox. So, if you wish to get the source code, please use the FreedomBox team's u-boot repository.

8.6.9 Obtaining Source Code

After you purchase and receive your Pioneer Edition FreedomBox, you may want to obtain the source code of the software running in it. Continue reading this section for instructions.

FreedomBox is fully free software and you can obtain the source code to study, modify and distribute improvements.

8.6.9.1 From within FreedomBox

FreedomBox is made up of several software programs and you can obtain the source code to any of them. These instructions are similar to obtaining and building source code for Debian since FreedomBox is a pure blend of Debian. Using this process you can obtain the source code to the exact version of the package you are currently using in FreedomBox.

1. To see the list of software packages installed on your FreedomBox, run the following in a terminal:

   ```
   dpkg -l
   ```

2. To obtain the source code for any of those programs, then run:

   ```
   apt source <package_name>
   ```

   This requires that the file /etc/apt/sources.list file contains the information about the source code repositories. These are present by default on all FreedomBox images. If you have installed FreedomBox using a package from Debian, you need to ensure that source repositories are added in the file.

3. To build the package from source code, first install its dependencies

   ```
   apt build-dep <package_name>
   ```

   Switch to the source directory created by the *apt source* command:

   ```
   cd <source_directory>
   ```

   Then build the package

   ```
   dpkg-buildpackage -rfakeroot -uc
   ```

4. Install the package:

   ```
   dpkg -i ../<built_package>.deb
   ```

8.6.9.2 Other Ways to Obtain Source Code

1. Source code for any of the packages can be browsed and searched using the web interface at sources.debian.org. For example, see the plinth package.
2. Source code and pre-built binary package for any version of a package including historic versions can be obtained from snapshot.debian.org. For example, see the plinth package.
3. You can also obtain the links to upstream project homepage, upstream version control, Debian's version control, changelog, etc. from the Debian tracker page for a project at tracker.debian.org. For example, see the tracker page for plinth package.
4. You can build and install a package from its Debian's version control repository. For example,

```
git clone https://salsa.debian.org/freedombox-team/freedombox.git
cd freedombox
apt build-dep .
dpkg-buildpackage -rfakeroot -uc
dpkg -i ../freedombox*.deb
```

8.6.9.3 Building Disk Images

You can also build FreedomBox disk images for various hardware platforms using the freedom-maker tool. This is also available as a Debian package and source code for it may be obtained using the above methods. Build instructions for creating disk images are available as part of the source code for freedom-maker package.

FreedomBox disk images are built and uploaded to official servers using automated Continuous Integration infrastructure. This infrastructure is available as source code too and provides accurate information on how FreedomBox images are built.

8.6.9.4 U-boot on Pioneer Edition Images

There is one minor exception to the u-boot package present on the hardware sold as FreedomBox Home Server Kits Pioneer Edition. It contains an small but important fix that is not part of Debian sources. The fork of the Debian u-boot source repository along with the minor change done by the FreedomBox is available as a separate repository. We expect this change to be available in upstream u-boot eventually and this repository will not be needed. This package can be built on a Debian armhf machine as follows (cross compiling is also possible, simply follow instructions for cross compiling Debian packages):

```
apt install git git-buildpackage
git clone https://salsa.debian.org/freedombox-team/u-boot.git
cd u-boot
pbuilder create --distribution=buster
gbp buildpackage --git-pbuilder
```

The u-boot Debian package will be available in *u-boot-sunxi*.deb*. This package will contain

```
mkdir temp
dpkg -x u-boot-suxi*.deb temp
unxz <lime2_image_built_with_freedom_maker>
dd if=temp/usr/lib/u-boot/A20-OLinuXino-Lime2/u-boot-sunxi-with-spl.bin of=<lime2.img> seek ↩
    =8 bs=1k conv=notrunc
```

The resulting image will have the modified u-boot in it.

8.6.10 Power and reset buttons

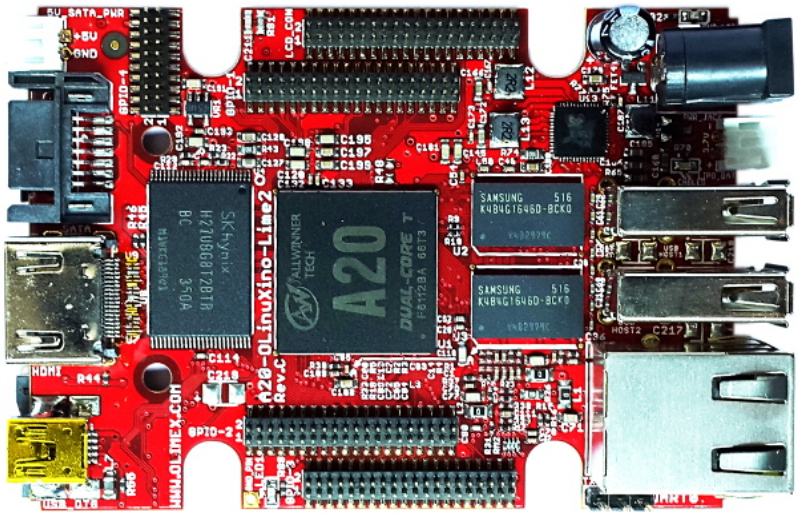
The Pioneer Edition Kit has 3 undocumented buttons for RESET, BOOT, and POWER. They are legacy features originally designed to be used with Android operating system but are useful in particular cases.



Warning: **Excessive force will break the buttons**. The three buttons are not intended to be pushed regularly, which is why they are fragile and not easily to push.

More details in our Forum.

8.7 A20 OLinuXino Lime2



Olimex's A20 OLinuXino Lime2 is a fully Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. It uses the Allwinner A20 Dual Core ARM processor.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.7.1 Similar Hardware

The following similar hardware will also work well with FreedomBox.

- Olimex's A20 OLinuXino Lime2 4GB. This hardware merely has extra 4GB NAND storage that is not used by FreedomBox.

8.7.2 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device. These SD card images are meant for use with the on-board SD card slot and won't work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.7.3 Availability

- Price: 45 EUR (A20 OLinuXino Lime2)
- Price: 55 EUR (A20 OLinuXino Lime2 4GB)
- Olimex Store

8.7.4 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3
- Storage: 4 GB NAND flash built-in (only on 4GB model), 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: None, use a USB WiFi device
- SATA: 1x port

8.7.5 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Boot Firmware: BROM (GPLV2+)

8.7.6 Known Issues

- Revision G2 (written on the board): The current stable Debian11/bullseye image (from 2021-08-27) **will not start an Ethernet connection!** This means you can not continue with the installation process at first. This can be fixed by following this description after you copied the image to your SD card. Or you Download and use the Pioneer Edition image to fix the issue. It contains a slightly modified u-boot.
- Revision G2 hardware has also poor performance when transmitting Ethernet data in Gigabit mode for older FreedomBox releases. The procedure above or for rev. C fixes this.
- Revision C hardware has poor performance when receiving Ethernet data in Gigabit mode. To workaround the problem, you can switch to 100 Mbps mode instead of Gigabit mode. Login to your FreedomBox as root (or plugin the SD card into another computer) and create the file /etc/NetworkManager/dispatcher.d/20-fix-ethernet-problem with the following contents:

```
#!/bin/bash

set -e  # Exit with code on error

IFACE="$1"
ACTION="$2"

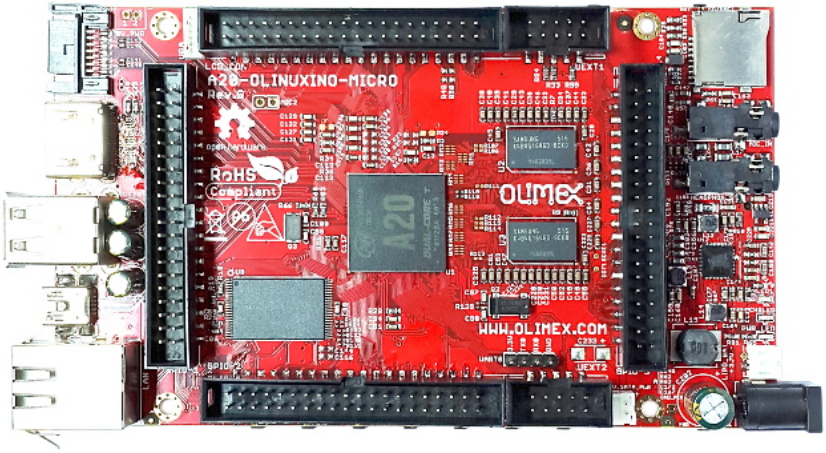
if [[ "$IFACE" != "eth0" ]]; then
    exit 0
fi

case ${ACTION} in
    up)
```

```
        logger "Setting up $IFACE in 100Mbps mode"
        mii-tool eth0 -A 100BaseTx-FD
        ;;
    *)
        ;;
esac
```

- Revision K hardware is not working properly.

8.8 A20 OLinuXino MICRO



Olimex's A20 OLinuXino MICRO is a fully Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. It uses the Allwinner A20 Dual Core ARM processor.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.8.1 Similar Hardware

The following similar hardware will also work well with FreedomBox.

- Olimex's A20 OLinuXino MICRO 4GB. This hardware merely has extra 4GB NAND storage that is not used by FreedomBox.

8.8.2 Download

FreedomBox MicroSD card images are available for this device. Follow the instructions on the download page to create a FreedomBox MicroSD card and boot the device. These MicroSD card images are meant for use with the on-board MicroSD card slot and won't work on the SD card slot or when using a separate MicroSD card reader connected via USB.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.8.3 Availability

- Price: 50 EUR (A20 OLinuXino MICRO)
- Price: 63 EUR (A20 OLinuXino MICRO 4GB)
- Olimex Store

8.8.4 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 1 GiB DDR3
- Storage: 4 GB NAND flash built-in (only on 4GB model), 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: 1x port

8.8.5 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available
- Boot Firmware: BROM (GPLV2+)

8.8.6 Known Issues

- Not visible on local network
- When booting the 'stable' image (made on 2017-06-18) the board does not automatically get an IP address from the router's DHCP server over ethernet. Booting the 'testing' image (2018-06) the board does get an IP address. Tested on MICRO hardware revision J. see also: https://www.olimex.com/forum/index.php?topic=5839.msg24167#msg24167

8.9 APU



PC Engines APU boards are single board computers with 3 Gigabit ethernet ports, a powerful AMD64 APU and Coreboot firmware. FreedomBox images built for AMD64 machines are tested to work well for APU1D and APU3B models and are expected to work also well on the other, very similar versions.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.9.1 Similar Hardware

Although untested, the following similar hardware is also likely to work well with FreedomBox.

- Using amd64 image:
  - apu1c
  - apu1c4
  - apu1d4
  - apu2b2
  - apu2b4
  - apu2c0
  - apu2c2
  - apu2c4
  - apu3a2
  - apu3a4
  - apu3b4

8.9.2 Installation

FreedomBox disk images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card, USB disk, SSD or hard drive and boot into FreedomBox. Pick the image meant for all amd64 machines.

An alternative to downloading these images is to install Debian on the APU and then install FreedomBox on it.

An installation manual tested on the APU3B is available on GitHub, including flashing with UEFI-BIOS

8.9.3 Networking

The first network port, the left most one in the above picture, is configured by FreedomBox to be an upstream Internet link and the remaining 2 ports are configured for local computers to connect to.

8.9.4 Availability

PCEngines announced the phase-out of these boards in June 2023.

In 2024 Dasharo announced the support of APU-boards : coreboot + SeaBIOS and coreboot+UEFI.

- Price: 110 - 170 USD (depending on the board and supplier)
- PC Engines
- Full list of suppliers

8.9.5 Hardware

- Open Hardware: No
- CPU: AMD G series T40E; GX-412TC, 1 GHz quad core (depending on model)
- RAM: 2 GB DDR3-1066 DRAM - 4 GB (depending on model)
- Storage: SD card, External USB, mSATA module
- Architecture: amd64
- Ethernet: 3 Gigabit Ethernet ports
- WiFi: wle200nx / wle600vx / wle900vx miniPCI express wireless modules
- SATA: 1 mSATA-module and 1 SATA

8.9.6 Non-Free Status

- Non-free blobs required: No
- Boot firmware: Coreboot

8.10 Cubietruck

8.10.1 FreedomBox Danube Edition



FreedomBox Danube Edition is a custom casing around Cubietruck and an SSD-hard drive.

8.10.2 Cubietruck / Cubieboard3

Cubietruck (Cubieboard3) is a single board computer with very good performance compared to many other boards. FreedomBox images are built for this device.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.10.3 Download

FreedomBox SD card images are provided for this hardware. These SD card images are meant for use with the on-board SD card slot and do not work when used with a separate SD card reader connected via USB.

An alternative to downloading these images is to install Debian on the Cubietruck and then install FreedomBox on it.

8.10.4 Availability

Cubietruck / Cubieboard3

- Price: 89 USD
- List of suppliers

8.10.5 Hardware

- Open Hardware: No
- CPU: Allwinner A20, ARM Cortex-A7 @ 1GHz dual-core
- RAM: 2 GiB DDR3 @ 480 MHz
- Storage: 8 GB NAND flash built-in, 1x microSD slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: Broadcom BCM4329/BCM40181 (no free WiFi drivers + firmware available)
- SATA: 1x 2.0 port

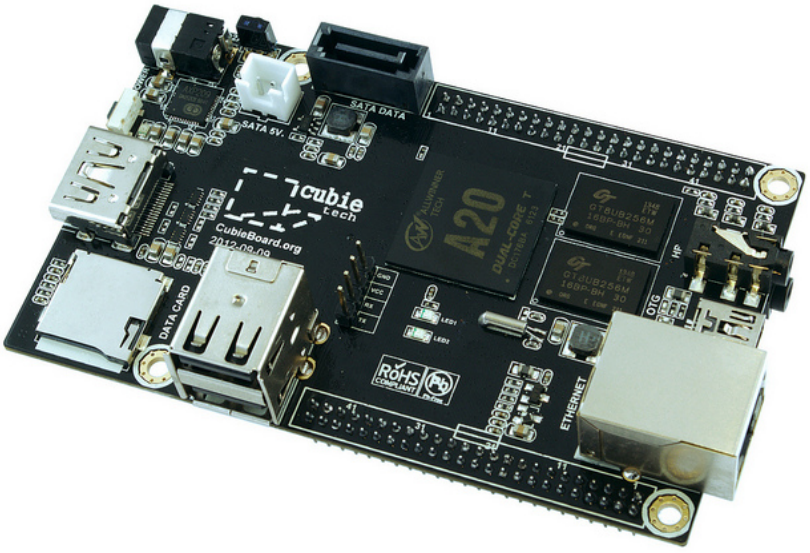
8.10.6 Non-Free Status

- Non-free blobs required: ?
- WiFi: no free WiFi drivers + firmware available

8.10.7 Known Issues

- The on-board WiFi does not work with free software. A separate USB WiFi device is recommended.

8.11 Cubieboard 2



The Cubieboard 2 is a single board computer based on the Allwinner A20 processor. It doesn't require any non-free firmware to run FreedomBox, and Wifi capability can be added via a USB adaptor if needed. This board is available in two versions, one with on-board flash and a microSD slot, and a version with two microSD card slots.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.11.1 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device.

8.11.2 Availability

- Full list of suppliers

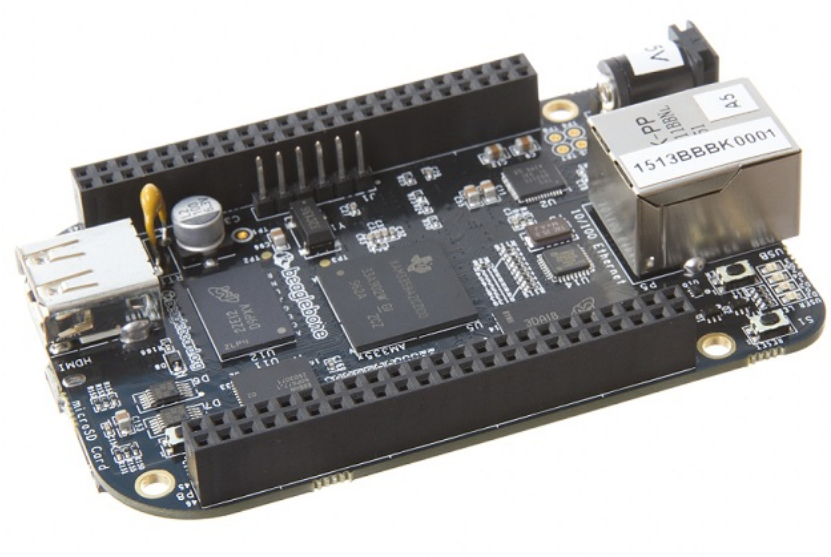
8.11.3 Hardware

- CPU: ARM Cortex A7 Dual-Core
- RAM: 1GB DDR3 @960M
- Storage: 4GB internal NAND flash, up to 64GB on uSD slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: Yes

8.11.4 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available

8.12 Beagle Bone Black



Beagle Bone Black (Revision C.1) is an Open Source Hardware (OSHW) single board computer. This means that the designer is actively helping people using the platform for their own designs, and supports them in adding hardware functionality and production advice. This is a part of freedom that is often overlooked, but very much aligned with the FreedomBox goals. FreedomBox images are built and tested for this device.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.12.1 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device.

Note: This image is for BeagleBone Black (Revision C.1) only. It will not work on the BeagleBone Green, and also not on the Revisions A & B.

An alternative to downloading these images is to install Debian on the BeagleBone and then install FreedomBox on it.

8.12.2 Availability

- Price: ~ 59 USD (50 EUR)
- Mouser Electronics
- Full list of suppliers

8.12.3 Hardware

- Open Source Hardware (OSHW): Yes
- CPU: AM335x 1GHz ARM Cortex-A8
- RAM: 512MB DDR3L 800 Mhz
- Storage: Onboard 4GB, 8bit Embedded MMC and microSD
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: None

8.12.4 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available

8.13 pcDuino3



LinkSprite pcDuino3S is a single board computer running on Allwinner A20 and sold with a good case. FreedomBox images are built and tested for this device.

Note: The FreedomBox logo is simply a sticker on top of device brought from store.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.13.1 Similar Hardware

Although untested, the following similar hardware is also likely to work well with FreedomBox.

- https://www.linksprite.com/linksprite-pcduino3/ also covers pcDuino3B

8.13.2 Download

FreedomBox disk images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card, USB disk, SSD or hard drive and boot into FreedomBox. Pick the image meant for pcduino3.

An alternative to downloading these images is to install Debian on the APU and then install FreedomBox on it.

8.13.3 Availability

- Price: 89 USD
- LinkSprite
- Full list of suppliers

8.13.4 Hardware

- Open Hardware: No
- CPU: AllWinner A20 SoC, 1GHz ARM Cortex A7 Dual Core
- RAM: 1 GB
- Storage: SD card, 4 GB onboard flash
- Architecture: armhf
- Ethernet: 10/100 Mbps
- WiFi: Built-in WiFi requires non-free firmware, use a USB WiFi device instead
- SATA: 1 SATA host socket

8.13.5 Non-Free Status

- Non-free blobs required: No
- WiFi: Requires non-free firmware
- Boot Firmware: BROM (GPLV2+)

8.14 Debian

FreedomBox is a pure blend of Debian. This means that all the work on FreedomBox is available in Debian as packages. It also means that any machine running Debian can be turned into a FreedomBox.

This page describes the process of installing FreedomBox on a Debian system. Currently, FreedomBox works in Debian Stable (bookworm), Testing (trixie), and Unstable (sid).

Important: Read general advice about hardware before building a FreedomBox with this approach.



Caution
Use a fresh Debian installation
Installing FreedomBox changes your Debian system in many important ways. This includes installing a firewall and regenerating server certificates. It is hence recommended that you install FreedomBox on a fresh Debian installation instead of an existing setup.

8.14.1 Installing on Debian 12 (bookworm) or newer

Check the Troubleshooting section below, for any tips or workarounds that might help during the install.

1. Install Debian 12 (Bookworm) or newer on your hardware. During installation choosing "Btrfs" for root filesystem type is recommended as FreedomBox uses it take regular snapshots of the system.
2. Update your package list.

   ```
   $ sudo apt-get update
   ```

3. Install `freedombox` package.

   ```
   $ sudo DEBIAN_FRONTEND=noninteractive apt-get install freedombox
   ```

   - The "DEBIAN_FRONTEND=noninteractive" will avoid several configuration prompts that would otherwise appear during the install.

4. During the installation, you will be provided a secret key that needs to be entered during the initial configuration process. Note this down. The secret can also be read at a later time from the file `/var/lib/plinth/firstboot-wizard-secret`.
5. You can start using FreedomBox. During initial wizard, you will need to enter the secret noted above.

8.14.2 Tips and Troubleshooting

1. FreedomBox uses NetworkManager to manage network configuration. If you have configured your network interfaces using Debian installer or by editing `/etc/network/interfaces`, FreedomBox will not manage those interfaces. (See bug #797614.) To let FreedomBox/NetworkManager manage your network interfaces, edit the `/etc/network/interfaces` manually and ensure that it contains only the following:

   ```
   auto lo
   iface lo inet loopback
   ```

   If you have already completed the setup process without doing this step, you will need to clear out the `/etc/network/interfa` file keeping only the above lines. Then perform a reboot. Network interfaces will then be in the `internal` or `external` firewall zone. This is essential for the FreedomBox's web interface to be reachable from other machines in the network. You can tweak network manager connections with the `nmtui` command if you wish.

2. FreedomBox will use an automatically configured IP address by default. You can assign a static IP address if necessary. Network configuration changes can be done using FreedomBox web interface or by using the `nmtui` or `nmcli` commands. `nmcli` can be used as follows:

```
nmcli con mod "Ethernet connection 1"  \
ipv4.addresses A.A.A.A/X  \
ipv4.gateway G.G.G.G  \
ipv4.dns N.N.N.N  \
ipv4.dns-search somedomain.com  \
ipv4.method "manual"  \
ipv4.ignore-auto-dns yes  \
ipv6.method ignore
```

...with the block capitals and somedomain.com replaced with your actual address, mask description, gateway and dns server details.

8.15 VirtualBox



This page will help you get started with using FreedomBox on a virtual machine using VirtualBox. While VirtualBox images are primarily used for testing and development, they can also be used for regular use if you have spare resources on one of your machines. This setup is useful if:

- You don't own one of the supported hardware devices.
- You don't use Debian GNU/Linux as your operating system.
- You don't want to disturb your Debian installation to try out FreedomBox.

Prebuilt FreedomBox images for VirtualBox are routinely made available in VirtualBox's own VDI image file format. They contain a Debian GNU/Linux operating system and an installation of FreedomBox with all dependencies ready to run on any OS supported by VirtualBox (Windows, Linux, Macintosh, and Solaris).

A more adventurous alternative to downloading one of these images is to install Debian on VirtualBox and then install FreedomBox on it.

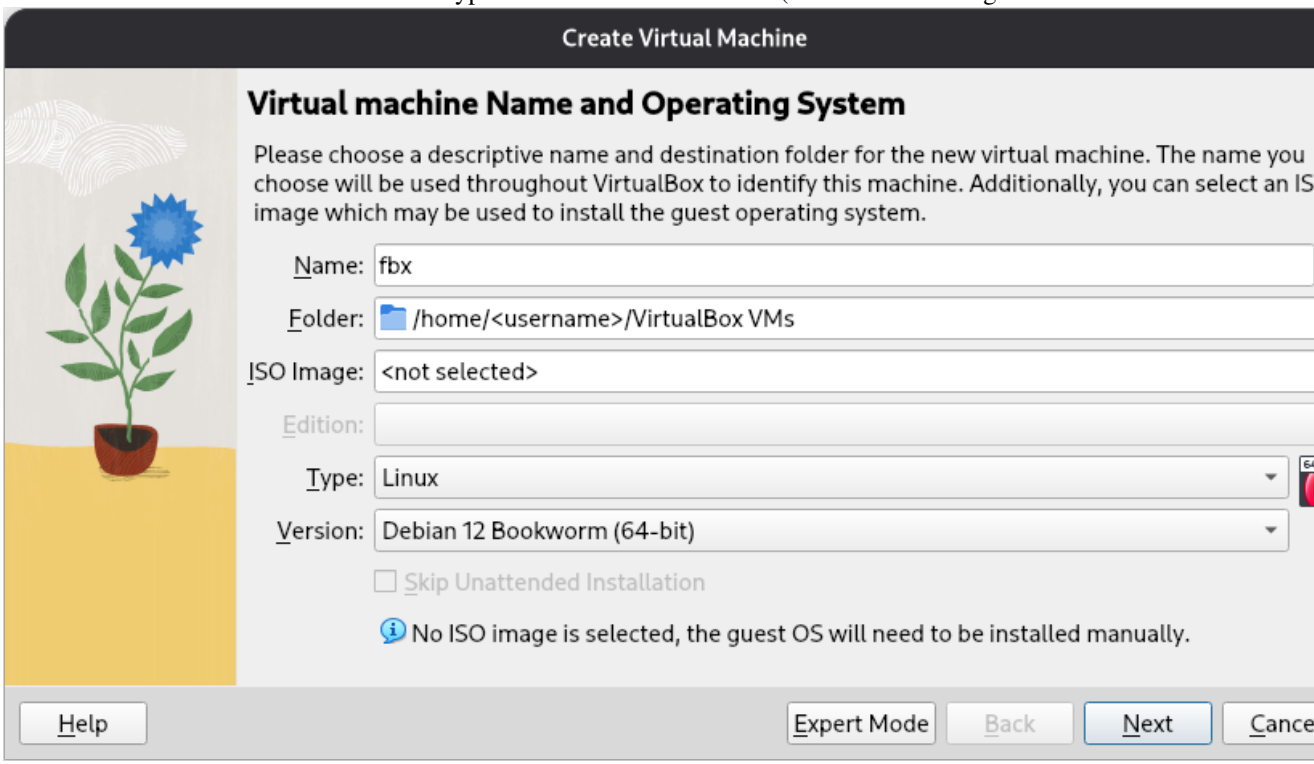
VirtualBox itself is available from `https://www.virtualbox.org/` (or your distribution's package manager).

8.15.1 Download

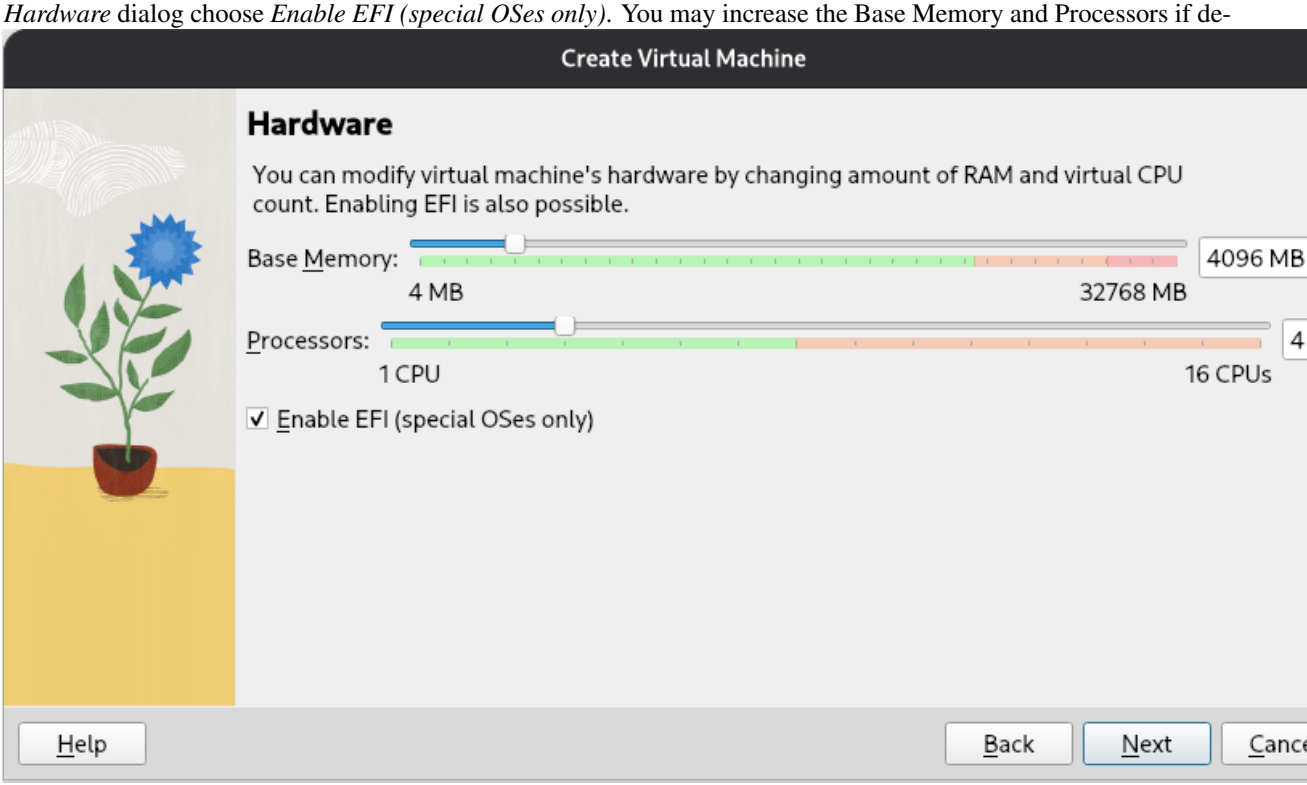
Follow the instructions on the download page to download and verify a VirtualBox image. The latest images are available on freedombox.org.

8.15.2 Creating a Virtual Machine

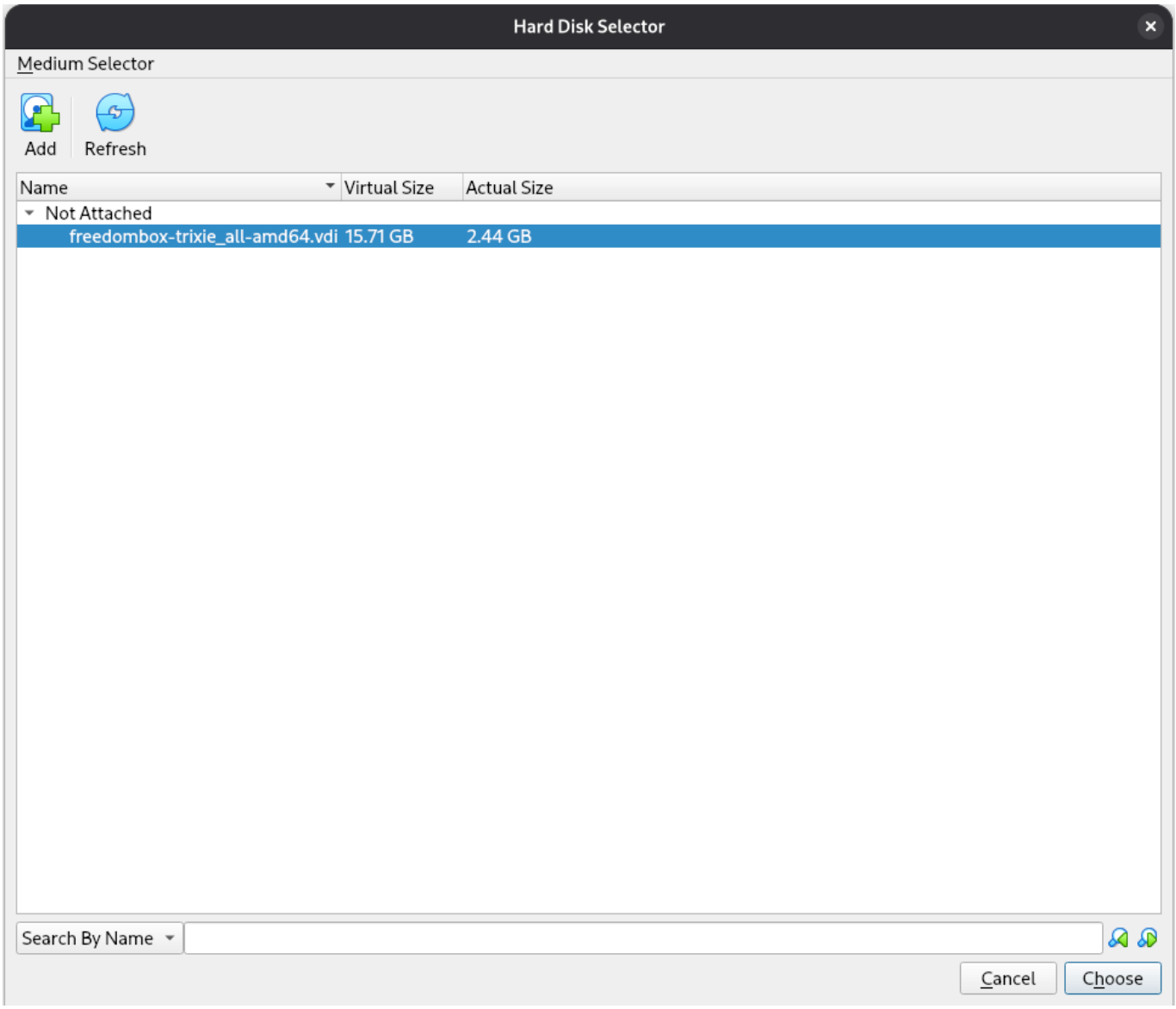
1. Decompress the downloaded VDI image (tool for Windows, Mac).
2. Create a new VM in the VirtualBox UI with OS type *Linux* and Version *Debian* (32/64-bit according to the downloaded image).



3. In the *Hardware* dialog choose *Enable EFI (special OSes only)*. You may increase the Base Memory and Processors if desired.

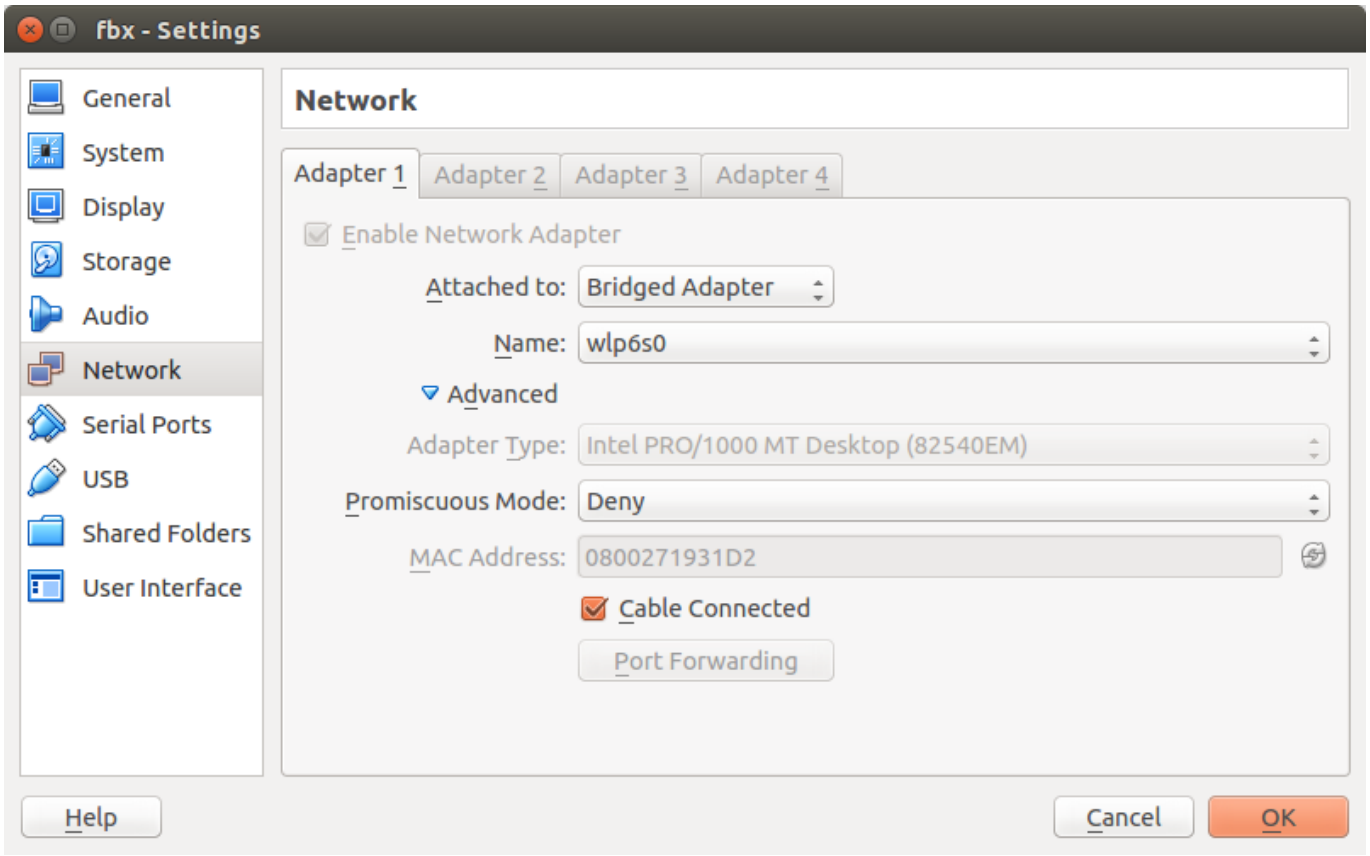


4. In the *Hard disk* dialog choose *Use an existing virtual hard disk file* and select the .vdi file you extracted in step 1.


Hard Disk Selector
Medium Selector
Add
Refresh
Name
Virtual Size
Actual Size
Not Attached
freedombox-trixie_all-amd64.vdi 15.71 GB 2.44 GB
Search By Name
Cancel
Choose



5. When created, go to the virtual machine's Settings -> [Network] -> [Adapter 1]->[Attached to:] and choose the network type your want the machine to use according to the explanation in Network Configuration below. The recommended type is the *Bridged adapter* option, but be aware that this exposes the FreedomBox's services to your entire local network.



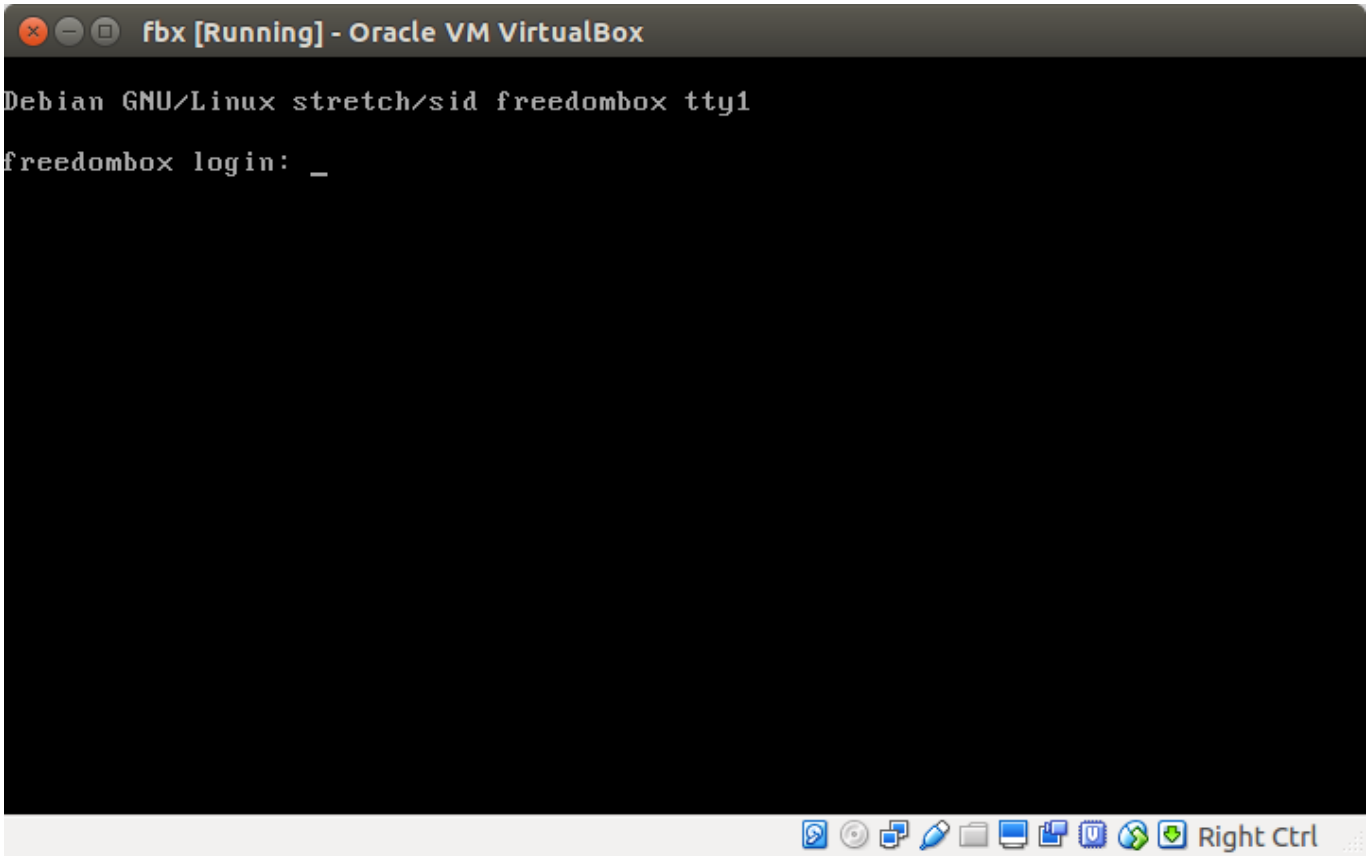
Note: It is important to make sure that you have provided the correct network interface in the above step. For example, if the

virtual machine is running on a laptop connected to a Wi-Fi network, then the wireless interface (starts with *wlp*) must be chosen as shown in the screenshot.

8.15.3 First Boot

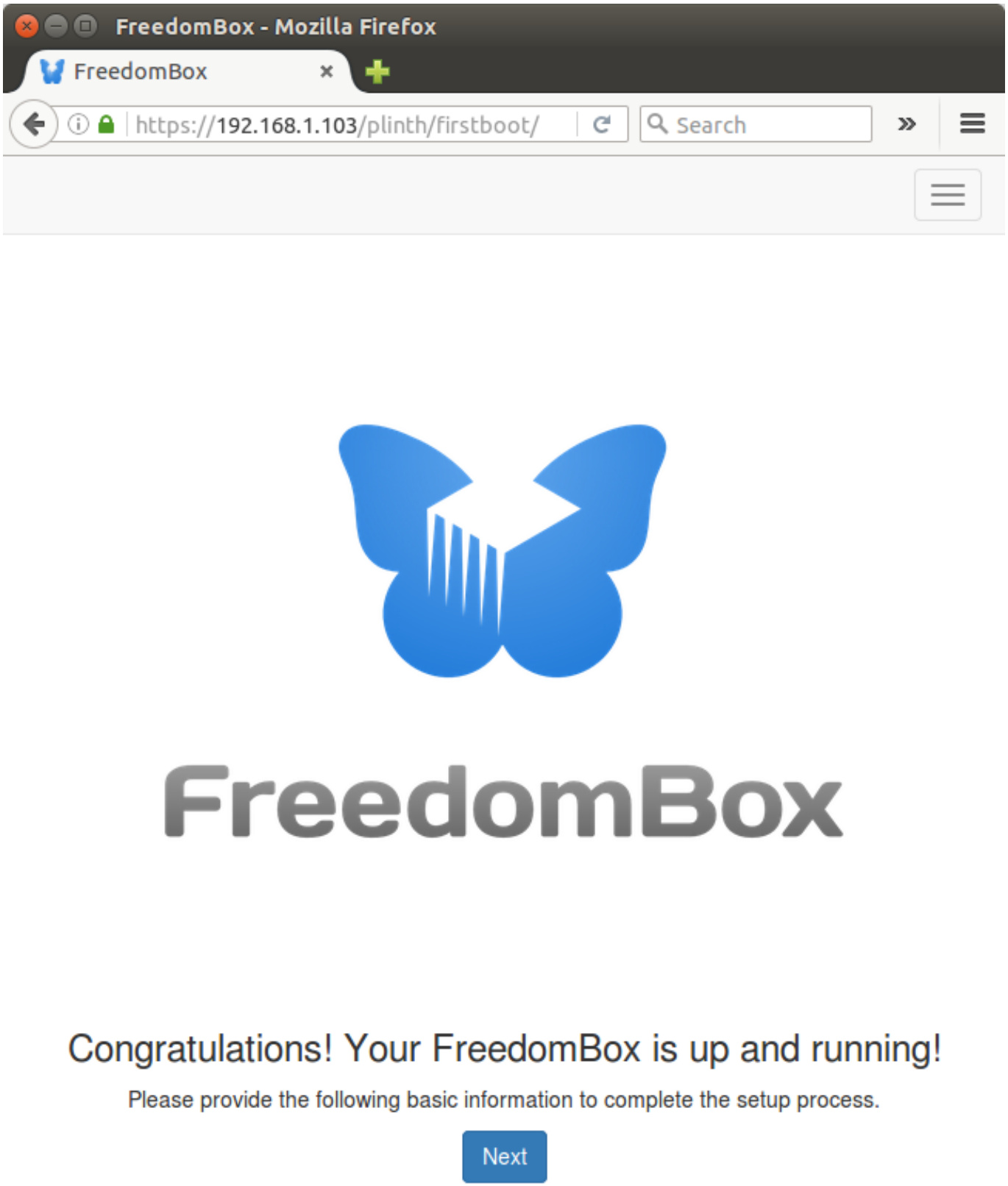
When satisfied with the VM settings click the start button in the VirtualBox UI and your new FreedomBox will boot.

The console of the VM will show the textual screen below when finished booting, from here most interaction with FreedomBox will be through the web interface in a browser.



If everything went well so far, you should be able to access the web interface of FreedomBox by pointing a browser on the host machine to `https://freedombox.local`.

In case freedombox.local cannot be resolved, you need to find out your FreedomBox's IP address as described in Finding out the IP address of the virtual machine. Then access this IP from a web browser which is on the same network as the VM (for example, the host). If all is well, you are now presented with a welcome message and invited to complete the *first boot* process.



This mainly consist of creating an administrative user for the system.

8.15.4 Using

See the FreedomBox usage page for more details.

You can log in to the Debian GNU/Linux system as the user created during FreedomBox first boot on the VirtualBox console or remotely via ssh.

After logging in, you can become root with the command `sudo su`.

8.15.5 Build Image

If you wish to build your own images instead of downloading available images, it can be done using Freedom Maker.

8.15.6 Tips & Troubleshooting

8.15.6.1 Network Configuration

VirtualBox provides many types of networking options. Each has its advantages and disadvantages. For more information about how various networking types work in VirtualBox, see VirtualBox's networking documentation. https://www.virtualbox.org/manual/ch06.html

For a simple setup, it is recommended that you use a single network interface in your guest machine. This will make the first boot script automatically configure that interface as an `internal` network with `automatic` network configuration. Inside the guest machine, the networking is configured automatically and all the services are made available on this network interface. For more information on how networks are configured by default in FreedomBox, see Networks section.

What remains is to make those services available to the host machine or to other machines in the network. You must then choose one of the following types of networking for the network interface on your guest machine. To set a particular type of network for the guest's network adapter, go to the guest VM's settings then the network options and then select the adapter you wish to configure. There, set the network type from the available list of networks.

1. First and the recommended option is to use the *Bridged* type of network. This option exposes the guest machine to the same network that host network is connected to. The guest obtains network configuration information from a router or DHCP server on the network. The guest will appear as just another machine in the network. A major advantage of this of setup is that the host and all other machines in the network will be able to access the services provided by guest without requiring any further setup.

   The only drawback of this approach is that if the host is not connected to any network, the guest's network will remain unconfigured making it inaccessible even from the host.

2. Second method is *Host only* type of networking. With a guest's network interface configured in this manner, it will only be accessible from the host machine. The guest will not able access any other machine but the host, so you do not have internet access on the guest. All services on the guest are available to the host machine without any configuration such as port forwarding.

3. The third option is to use the *NAT* type of network. This the networking type that VirtualBox assigns to a freshly created virtual machine. This option works even when host is not connected to any network. The guest is automatically configured and is able to access the internet and local networks that host is able to connect to. However, the services provided by the guest require port forwarding configuration setup to be available outside.

   To configure this go to VM settings -> [Network] -> [Adapter] -> [Port Forwarding]. Map a port such as 2222 from host to guest port 22 and you will be able to ssh into FreedomBox from host machine as follows:

   ```
   ssh -p 2222 fbx@localhost
   ```

   Map 4443 on host to 443 on the guest. This make FreedomBox HTTPS service available on host using the URL https://localhost:4443/

   You will need to add a mapping for each such services from host to guest.

4. The final option is to create two network interfaces, one *host only* and one *NAT* type. This way you can access the guest without any additional configuration, and you have internet access on the guest. The guest will be invisible to any other machines on the network.

Summary of various network types:

-	Guest accessible from other machines	Guest accessible from host	Works without port forwarding	Works without host connected to network	Guest has internet access
Bridged	✔	✔	✔	⛔	✔
Host only	⛔	✔	✔	✔	⛔
NAT	✔	✔	⛔	✔	✔
NAT and Host	⛔	✔	✔	✔	✔

8.15.6.2 Finding out the IP address of the virtual machine

This depends on the network configuration you chose. With a *bridged adapter*, your virtual machine gets its IP address from the DHCP server of your network, most likely of your Router. You can try the first couple of IP addresses or check your router web interface for a list of connected devices.

If you chose *host-only adapter*, the IP address is assigned by the DHCP server of your VirtualBox network. In the VirtualBox Manager, go to File -> Preferences -> Network -> Host-only Networks. You can see and edit the DHCP address range there, typically you get assigned addresses close to the *Lower Address Bound*.

Another possibility of finding the IP address is to login via the VirtualBox Manager (or similar software). The FreedomBox images do not have any default user accounts, so you need to set an initial user and password using the passwd-in-image script.

See also QuickStart for instructions on how to scan your network to discover the IP of the VM.

8.15.6.3 Networking Problems with macchanger

The package `macchanger` can cause network problems with VirtualBox. If you have a valid IP address on your guest's host network adapter (like 192.168.56.101) but are not able to ping or access the host (like 192.168.56.1), try uninstalling `macchanger`:

```
$ dpkg --ignore-depends=freedombox-setup --remove macchanger
```

You might have to manually remove the script `/etc/network/if-prep-up/macchanger`. If Debian complains about unmet dependencies when you use a package manager (apt-get, aptitude, dpkg), try to remove 'macchanger' from the dependencies of 'freedombox-setup' in the file `/var/lib/dpkg/status`.

8.15.6.4 Mounting Images Locally

If you want to mount images locally, use the following to copy built images off the VirtualBox:

```
$ mkdir /tmp/vbox-img1 /tmp/vbox-root1
$ vdfuse -f freedombox-unstable_2013.0519_virtualbox-amd64-hdd.vdi /tmp/vbox-img1/
$ sudo mount -o loop /tmp/vbox-img1/Partition1 /tmp/vbox-root1
$ cp /tmp/vbox-root1/home/fbx/freedom-maker/build/freedom*vdi ~/
$ sudo umount /tmp/vbox-root1
# $ sudo umount /tmp/vbox-img1 # corruption here.
```

8.15.6.5 Fixing the time after suspend and resume

The virtual machine loses the correct time/date after suspending and resuming. One way to fix this is to create a cron-job that restarts the time service `ntp`. You can add a crontab entry as root to restart ntp every 15 minutes by typing `'crontab -e'` and adding this line:

```
*/15 * *   *   *     /etc/init.d/ntp restart
```

Do not restart this service too often as this increases the load of publicly and freely available NTP servers.

8.15.6.6 UUID collision in VB

Whenever this happens VirtualBox shows following error message: *Cannot register the hard disk A with UUID ... because a hard disk B with UUID ... already exists in the media registry*

Creating several VMs from the same image causes collisions due to ID's (hostname, IP, UUID, etc) that are expected to be universally unique. Most can be handeled operating the running VM. But VirtualBox complains before that (at the very creation of the VM) about the hard disk's UUID. This is usual stuff when you develop/test e.g. FreedomBox.

You can change a clone's UUID in the terminal as follows:

```
$ VBoxManage internalcommands sethduuid path/to/the/hd/vdi/file
```

8.16 Pine A64+



Pine A64+ is an affordable single board computer with good performance.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.16.1 Similar Hardware

- Both 1GB and 2GB versions of Pine A64+ are supported with the same FreedomBox image.
- There is a separate Pine A64-LTS image.

8.16.2 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox. Pick the image meant for Pine A64+.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.16.3 Availability

- Price: 29 USD (for the 2 GB variant), 21 USD (for the 1 GB variant)
- Pine A64+ with 1 GB RAM at Pine64 Store
- Pine A64+ with 2 GB RAM at Pine64 Store

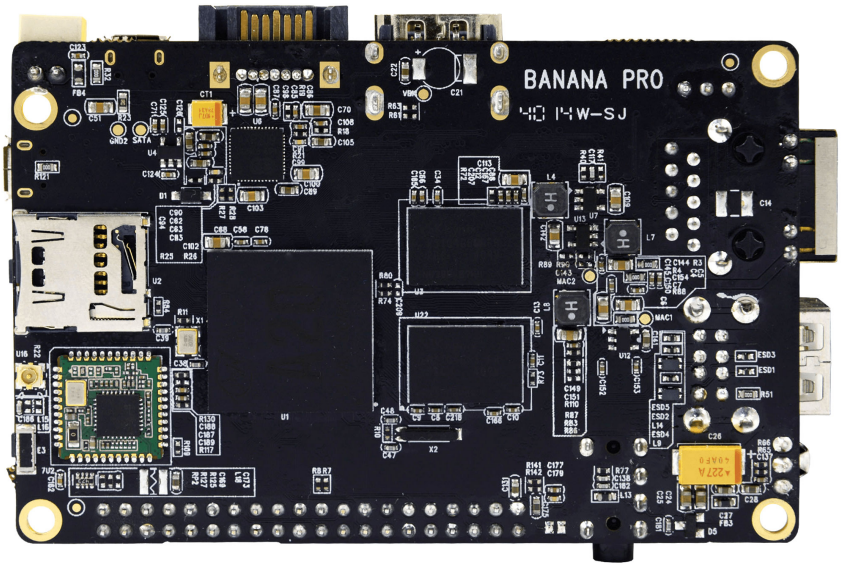
8.16.4 Hardware

- Open Source Hardware (OSHW): No
- CPU: Allwinner A64, Quad-core ARM Cortex A53 64-bit processor
- RAM: 3 variants - 512 MB (not recommended), 1 GB and 2 GB (recommended)
- Storage: SD card, eMMC (module sold separately but not tested with FreedomBox)
- Architecture: arm64
- Ethernet: Gigabit Ethernet port
- Battery: Supports battery backup using a Li-Po battery
- WiFi: None, use a USB WiFi device
- SATA: None

8.16.5 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available

8.17 Banana Pro



LeMaker Banana Pro is an updated version of its predecessor Banana Pi.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.17.1 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox. Pick the image meant for Banana Pro.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

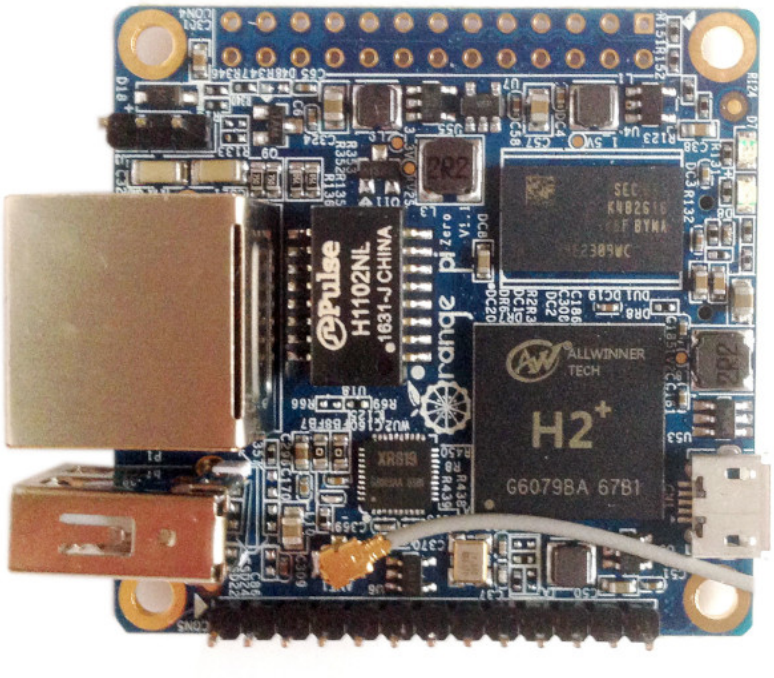
8.17.2 Hardware

- Open Source Hardware (OSHW): No
- CPU: Allwinner A20, Dual-core ARM Cortex A7 processor
- RAM: 3 variants - 1 GB
- Storage: SD card
- Architecture: armhf
- Ethernet: 10/100/1000 Mbps
- Battery: No
- WiFi: WiFi 802.11 b/g/n 2.4GHz (not tested with FreedomBox)
- SATA: SATA 2.0 (2.5 inch SSD or HDD recommended)

8.17.3 Non-Free Status

- Non-free blobs required: No
- WiFi: Unknown

8.18 Orange Pi Zero



Orange Pi Zero is a single board computer available at very low price. It uses the Allwinner H2 SoC, and has 256MB/512MB DDR3 SDRAM. It doesn't require any non-free firmware to run FreedomBox. However, the onboard Wi-Fi module needs proprietary firmware to work. The board is available in two versions: with 256MB RAM and 512MB RAM. The version with 512 MB RAM is recommended for FreedomBox. Even then, FreedomBox is expected to gracefully run only a small number of services.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.18.1 Download

FreedomBox SD card images are available for this device. Follow the instructions on the download page to create a FreedomBox SD card and boot the device.

8.18.2 Availability

- AliExpress

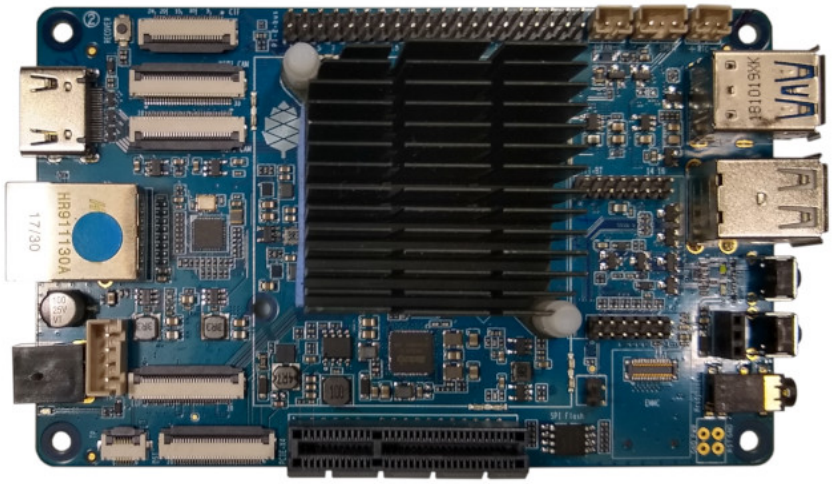
8.18.3 Hardware

- CPU: ARM Cortex-A7 Quad-Core (Allwinner H2)
- RAM: 256MB/512MB DDR3 SDRAM
- Storage: Up to 32GB on uSD slot, 2MB SPI Flash
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: Onboard 802.11 b/g/n, use a USB WiFi device

8.18.4 Non-Free Status

- Non-free blobs required: No (without Wi-Fi)
- Wi-Fi: no free Wi-Fi drivers + firmware available

8.19 RockPro64



Pine64's RockPro64 is a powerful single board computer. It uses the Rockchip RK3399 Hexa Core ARM64 processor. FreedomBox images are built and tested for this device.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.19.1 Download

Before downloading and using FreedomBox you need to ensure that latest u-boot based firmware is installed into the SPI flash chip. See instructions on how to write u-boot firmware into SPI flash. The gist is that you download and write an image to an SD card. Boot with SD card and wait for white LED blinking to stop. After that power off, remove the SD card and proceed with FreedomBox download.

FreedomBox images meant for all "arm64" hardware work well for this device. However, u-boot firmware must be present in SPI flash (or on a separate SD card, which is not explained here). Follow the instructions on the download page to create a FreedomBox SD card and boot the device. These images also work well for onboard eMMC, USB 2.0 and USB 3.0 disk drives. The process for preparing USB disk drives is same as for an SD card. For eMMC, boot the board using any OS and then write the image to the eMMC similar to writing to SD card.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.19.2 Availability

- Price: 60 USD (RockPro64 2GB)
- Price: 80 USD (RockPro64 4GB)

8.19.3 Hardware

- Open Source Hardware (OSHW): No
- CPU: Rockchip RK3399 SOC (2x Cortex A72@1.8Ghz, 4x Cortex A53@1.4Ghz)

- GPU: Mali T860 MP4 GPU
- RAM: 2 GiB or 4 GiB LPDDR4
- Storage: eMMC module slot, microSD slot, 16 MiB SPI Flash
- USB: 2x USB 2.0, 1x USB 3.0, 1x USB-C
- Expansion slot: 1x PCIe 4x slot (NVMe disks, etc.)
- Architecture: arm64
- Ethernet: 10/100/1000, RJ45
- WiFi: None, use a USB WiFi device

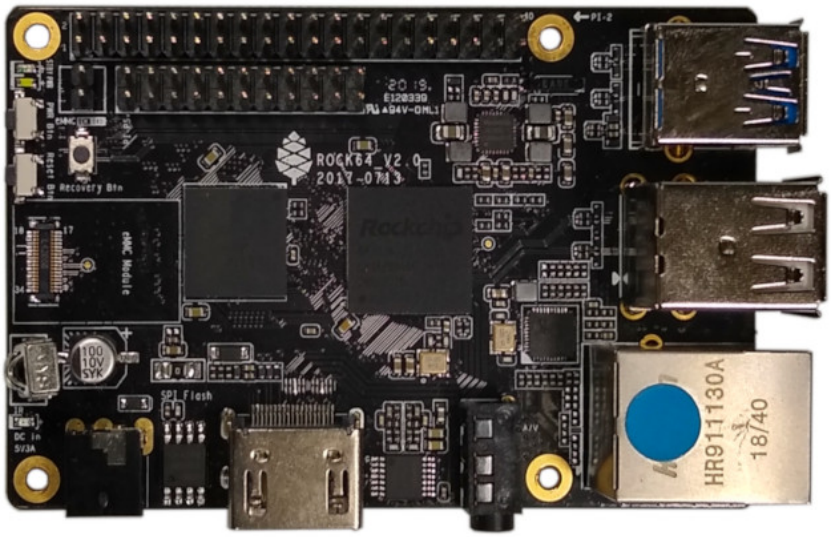
8.19.4 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available

8.19.5 Known Issues

- FreedomBox on NVMe disk has not been tested.

8.20 Rock64



Pine64's Rock64 is a powerful single board computer. It uses the Rockchip RK3328 Quad Core ARM64 processor. FreedomBox images are built and tested for this device.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.20.1 Download

Before downloading and using FreedomBox you need to ensure that latest u-boot based firmware is installed into the SPI flash chip. Download the latest u-boot to write to SPI flash and then see instructions on how to write u-boot firmware into SPI flash. The gist is that you download and write an image to an SD card. Boot with SD card and wait for white LED to blink continuously. After that power off remove SD card and proceed with FreedomBox download.

FreedomBox images meant for all "arm64" hardware work well for this device. However, u-boot firmware must present in SPI flash (or on a separate SD card, which is not explained here). Follow the instructions on the download page to create a FreedomBox SD card and boot the device. These images also work well for eMMC disk which an optional attachment to this board and disk drives in USB 2.0 ports (but not in the USB 3.0 port). The process for preparing them is same as for an SD card.

An alternative to downloading these images is to install Debian on the device and then install FreedomBox on it.

8.20.2 Availability

- Price: 25 USD (1GB)
- Price: 35 USD (2GB)
- Price: 45 USD (4GB)
- Pine64 Store

8.20.3 Hardware

- Open Source Hardware (OSHW): No
- CPU: Rockchip RK3328 Quad-Core SOC (4x Cortex A53 @ 1.5Ghz)
- GPU: Mali 450MP2
- RAM: 1 GiB or 2 GiB or 4 GiB LPDDR3
- Storage: eMMC module slot, microSD slot, 16 MiB SPI Flash
- USB: 2x USB 2.0, 1x USB 3.0
- Architecture: arm64
- Ethernet: 10/100/1000, RJ45
- WiFi: None, use a USB WiFi device

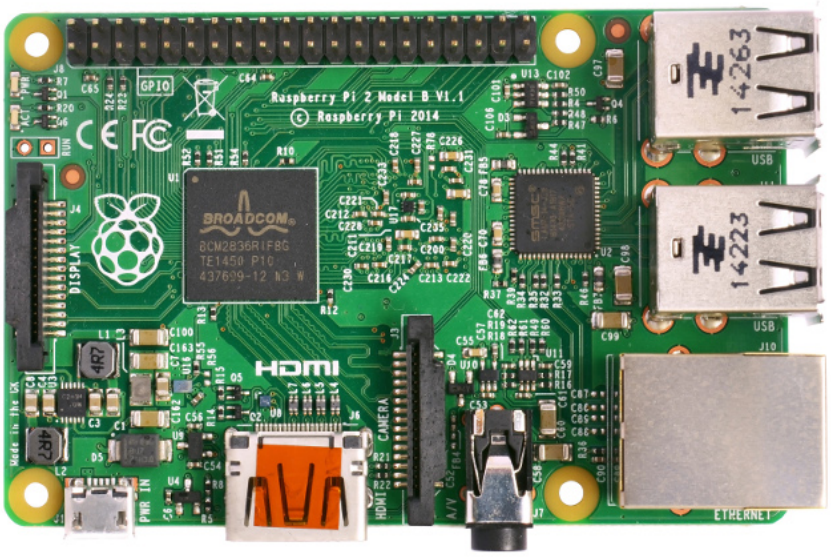
8.20.4 Non-Free Status

- Non-free blobs required: No
- WiFi: Not available

8.20.5 Known Issues

- FreedomBox does not work when booted from USB 3.0 port (but works from eMMC, SD card or USB 2.0 disk).
- FreedomBox does not work when booted form the top USB 2.0 port with some u-boot firmware versions (the one listed above). It only works with the bottom USB 2.0 port (the one closer to the board).

8.21 Raspberry Pi 2 Model B



Raspberry Pi 2 (Model B) is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi Model B+ with much faster processor and more RAM. FreedomBox images are built and tested for it.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.21.1 Download

FreedomBox SD card images for this hardware are available. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

8.21.2 Availability

- Price: 35 USD
- List of official distributors

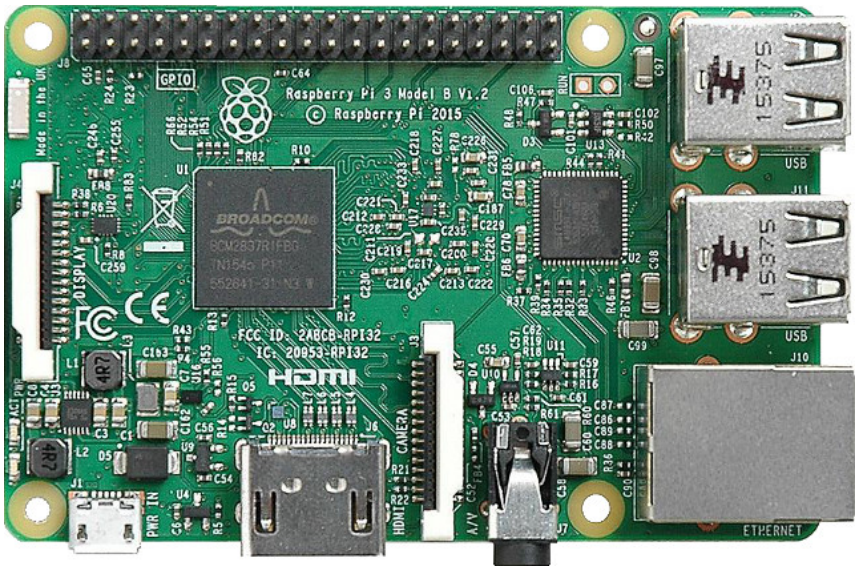
8.21.3 Hardware

- Open Hardware: No
- CPU: 900 MHz quad-core ARM Cortex-A7
- RAM: 1 GB
- Storage: MicroSD card slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: None, use a USB WiFi device
- SATA: None

8.21.4 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Not available

8.22 Raspberry Pi 3 Model B



Raspberry Pi 3 Model B is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi 2 Model B with a 64-bit processor and on-board Wi-Fi. FreedomBox "stable" and "testing" images are available for Raspberry Pi 3 Model B.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.22.1 Download

Raspberry Pi recommends creating a bootable SD card with the Raspberry Pi Imager (rpi-imager), their official cross-platform utility that downloads an image and writes it to removable media. FreedomBox "stable" and "testing" images can be selected in rpi-imager under the "Other specific-purpose OS" category.

FreedomBox SD card images for this hardware are also available online. Download the "stable" or "testing" image for Raspberry Pi 3 Model B. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

8.22.2 Availability

- Price: 35 USD
- List of official distributors

8.22.3 Hardware

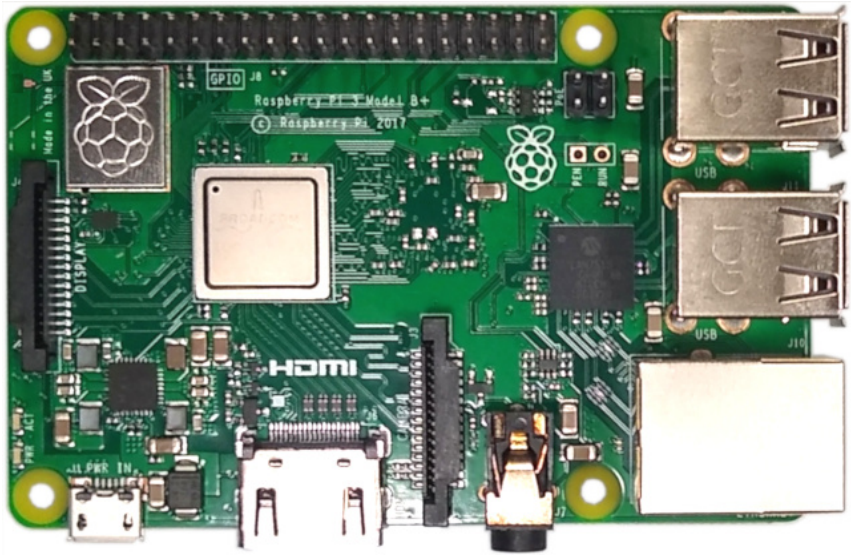
- Open Hardware: No
- CPU: 1.2GHz 64-bit quad-core ARMv8 CPU
- RAM: 1 GB

- Storage: MicroSD card slot
- Architecture: armhf
- Ethernet: 10/100, RJ45
- WiFi: 802.11n but requires non-free firmware, instead use a USB WiFi device
- SATA: None

8.22.4 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Requires non-free firmware

8.23 Raspberry Pi 3 Model B+



Raspberry Pi 3 Model B+ is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi 3 Model B with better Ethernet and a 5Ghz Wi-Fi. FreedomBox "stable" and "testing" images are available for Raspberry Pi 3 Model B+.

Please do not expect any output on a monitor connected via HDMI to this device as it does not display anything beyond the message 'Starting kernel...'. See the Quick Start page to access and control your FreedomBox from network.

Important: Read general advice about hardware before building a FreedomBox with this single board computer.

8.23.1 Download

Raspberry Pi recommends creating a bootable SD card with the Raspberry Pi Imager (rpi-imager), their official cross-platform utility that downloads an image and writes it to removable media. FreedomBox "stable" and "testing" images can be selected in rpi-imager under the "Other specific-purpose OS" category.

FreedomBox SD card images for this hardware are also available online. Download the "stable" or "testing" image for Raspberry Pi 3 Model B+. Follow the instructions on the download page to create a FreedomBox SD card and boot into FreedomBox.

8.23.2 Availability

- Price: 35 USD
- List of official distributors

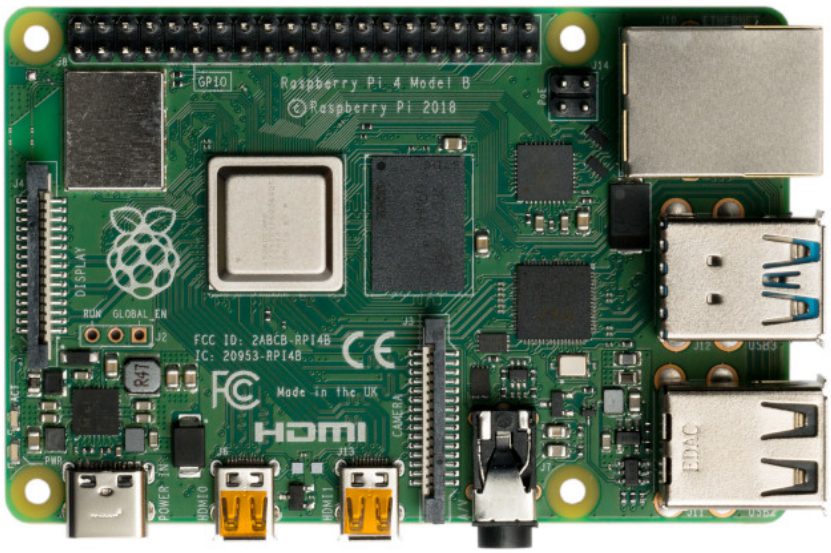
8.23.3 Hardware

- Open Hardware: No
- CPU: 1.4GHz 64-bit quad-core ARMv8 CPU
- RAM: 1 GB
- Storage: MicroSD card slot
- Architecture: armhf
- Ethernet: 10/100/1000, RJ45
- WiFi: 802.11ac but requires non-free firmware, instead use a USB WiFi device
- SATA: None

8.23.4 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Requires non-free firmware

8.24 Raspberry Pi 4 Model B



Raspberry Pi 4 Model B is a popular single board computer developed with the intention of promoting teaching of basic computer science in schools. It is a successor to Raspberry Pi 3 Model B+ with better processor and ability to drive multiple displays. Both stable and "testing" images are available for Raspberry Pi 4 Model B.

8.24.1 Download

Raspberry Pi recommends creating a bootable SD card with the Raspberry Pi Imager (rpi-imager), their official cross-platform utility that downloads an image and writes it to removable media. FreedomBox "stable" and "testing" images can be selected in rpi-imager under the "Other specific-purpose OS" category.

FreedomBox SD card images for this hardware are also available online. Download the "stable" or "testing" image for Raspberry Pi 4 Model B. Follow the instructions on the download page to create a FreedomBox SD card or USB drive and boot into FreedomBox.

8.24.2 Booting from USB

MicroSD cards are notouriously slow and prone to be corrupted. SD card corruption means that a high read/write load or a sudden power outage can make render your card unusable. It is therefore strongly recommended that you use a USB drive for your root partition. To be able to do that, you have to follow these preliminary steps:

1. Download and install the Raspberry Pi Imager.
2. Insert a microSD card to your computer.
3. Open the Raspberry Pi Imager 3.1 Select **CHOOSE OS >> Misc utility images >> Bootloader >> USB Boot**

   3.2 From **CHOOSE STORAGE** select your microSD card

   3.3 Click on **WRITE**
4. After the writing was succesful, you should eject the card.
5. Insert the newly created card into your Raspberry Pi and boot up the device. If it is connected to a monitor, you will see a green screen. Let your device rest for a minute, after which you can disconnect it from power and eject the card from the Raspberry Pi. It can now boot from USB.
6. Final step: write your FreedomBox image into your USB device, then connect it to your Raspberry Pi. Make sure you did not leave your microSD card in the Raspberry Pi.

Please note that steps 1. to 5. only need to be done once.

8.24.3 Build Image

FreedomBox images for this hardware can be built using Freedom Maker. Use the target 'raspberry64' to build the image for this board.

8.24.4 Availability

- Price: 35 USD (2GB RAM)
- Price: 50 USD (4GB RAM)
- Price: 75 USD (8GB RAM)
- List of official distributors

8.24.5 Hardware

- Open Hardware: No
- CPU: Broadcom BCM2711 SOC (4x Cortex-A72@1.5GHz)
- RAM: 2 GB or 4GB or 8 GB
- Storage: MicroSD card slot or USB drive
- USB: 2x USB 2.0, 2x USB 3.0, USB Type-C power supply
- Architecture: arm64
- Ethernet: 10/100/1000, RJ45
- WiFi: 802.11ac but requires non-free firmware, instead use a USB WiFi device
- SATA: None

8.24.6 Non-Free Status

- Non-free blobs required: boot firmware
- WiFi: Requires non-free firmware

8.25 USB Wi-Fi

FreedomBox works on many single board computers. However, many of these boards do not have built-in Wi-Fi capabilities. Even when Wi-Fi capability is available, non-free proprietary firmware is required to make them work.

A solution to the problem is to plug-in a USB Wi-Fi device into one of the available USB ports. There are many such devices available which do not require non-free firmware to work. The following is a list of such devices that work with FreedomBox devices. Some devices based on these chips have tested to work well with FreedomBox including functions such as access point mode.

- Devices with Atheros AR7010 chip
- Devices with Atheros AR9271 chip

8.25.1 Firmware Installation

The free firmware for these devices is not packaged in Debian yet. You can manually download and install the firmware as follows:

```
sudo su [enter password]
cd /lib/firmware
wget https://www.thinkpenguin.com/files/ath9k-htc/version-1.4-beta/htc_9271.fw
wget https://www.thinkpenguin.com/files/ath9k_firmware_free-version/htc_7010.fw
```

8.25.2 Resources

- Debian Wiki on WiFi drivers
- Wikipedia: Comparison of open-source Linux wireless network drivers
- WikiDevi: database of computer hardware

9 Debian Upgrade Notes

FreedomBox is built on the Debian operating system. Approximately every two years, there is a new stable Debian release. FreedomBox provides the option of automatically handling the upgrade to the new version of Debian after it is released. A distribution update will bring in major features and changes. Sometimes, old features will stop working. This page documents the expected changes and transitions during a distribution upgrade. If you dislike these changes, you can keep each distribution for at least 5 years before updating.

The current stable release is Debian 12 "bookworm". It was released on June 10, 2023. Below are additional considerations to have in mind when upgrading to a new release of Debian.

9.1 Debian 13 "trixie"

Estimated release date is around August 2025.

9.1.1 TT-RSS no longer supported

The tt-rss package has been removed from the next Debian release. It is recommended to move to one of the following options as a replacement:

- Miniflux
- Nextcloud News

9.1.2 Searx app no longer supported

The searx package has been removed from the next Debian release. There is currently no replacement for Searx app in FreedomBox. Searx app will no longer received security updates, so it is recommended to stop using the app, and to uninstall it.

10 Release Notes

The following are the release notes for each FreedomBox version.

10.1 FreedomBox 26.5 (2026-03-23)

10.1.1 Highlights

- action_utils: Don't restart web interface when installing an app
- apache: Use a Uwsgi native socket systemd unit for each app
- ui: Use inline SVG icons for all apps
- wireguard: Fix freedombox VPN IP for services

10.1.2 Other Changes

- action_utils: Stop associated service when stopping a socket unit
- apache: Increase OpenID Connect RP session timeout activity
- apache: Minor improvement to getting the request host
- app: Fix build issue with Django 5.x
- clients: Fix formatting of package row in table
- clients: Fix show empty clients in Desktop section
- clients: Use SVG icons when showing external links
- container: Add option to skip install
- container: Fix image extension to .raw for systemd v260
- doc: Reduce verbosity when building documentation
- html: Drop trailing slash from void elements
- html: Drop type attribute value of text/javascript
- janus: Drop unused reference to font-awesome

- letsencrypt: Don't perform operations on apps that are not installed
- locale: Update translations for German, Russian, Swedish
- pagekite: Fix issue with adding custom services
- tests: functional: Drop undefined 'sso' pytest mark
- ui: Add rest of the icons used from fork-awesome set
- ui: Better placement for dropdown indicator in dropdown button
- ui: Drop fonts-fork-awesome as dependency
- ui: Rename 'plinth_extras' template tags module to 'extras'
- ui: Simplify SVG app icons for using them inline in HTML
- ui: Use inline SVG icons for buttons, messages, spinners, etc.
- wireguard: Add button for direct APK download
- wireguard: Add entries for Homebrew and RPM packages
- wireguard: Remove client entry for F-Droid which is not available
- wireguard: Update windows client link

10.2 FreedomBox 26.4.2 (2026-03-08)

10.2.1 Highlights

- apache2: Disable pubtkt authentication module

10.2.2 Other Changes

- container: Hold freedombox packages during test setup
- d/control: Trim deps for nocheck build profile
- locale: Update translations for Albanian, Chinese (Simplified Han script), Czech, Russian, Turkish
- Vagrantfile: Enable public network for bridged networking

10.3 FreedomBox 26.4 (2026-03-02)

10.3.1 Highlights

- backups: Enable key-based SSH authentication for remote backups
- oidc: New app to implement OpenID Connect Provider
- apache: Implement protecting apps using OpenID Connect
- wireguard: Improve server section user experience flow

10.3.2 Other Changes

- : Remove some absolute file paths in SVGs
- : Update URL base from /plinth to /freedombox
- README/HACKING: Update weblate project path to /freedombox
- Vagrantfile: Drop unnecessary sudo configuration for actions
- action_utils: Drop support for link-local IPv6 addresses
- action_utils: Fix issue with type checking a generator
- action_utils: Implement utility to change umask temporarily
- actions, privileged_daemon: Drop some unused global statements
- apache: Fix diagnosing URLs protected by OpenID Connect
- apache: Preserve host header when proxying to service
- backups: Arrange form for adding remote location
- backups: Avoid some repeated text in form help text
- backups: Copy SSH client public key to remote
- backups: Create .ssh folder before creating SSH key
- backups: Create a better comment in the generated SSH key file
- backups: Display SSH public key when adding remote
- backups: Fix issue with Javascript in add remote location form
- backups: Fix showing proper error for incorrect passphrase
- backups: Fix type checking errors
- backups: Generate SSH client key if needed
- backups: Migrate to SSH key auth when mounting
- backups: Minor refactoring
- backups: Show/hide form elements instead of disabling for simplicity
- backups: Simplify handling of migration to SSH keys
- backups: Test adding/removing remote location
- backups: Tweak appearance of add remote location form
- backups: Use SSH key instead of password
- backups: Use selected SSH credential for remote
- backups: tests: Simplify functional test using more classes
- bin: Add tool to change FreedomBox password in Django database
- calibre: Use OpenID Connect instead of pubtkt based SSO
- cfg: Drop unused actions_dir option
- cfg: Drop unused config_dir option
- container: Align terminology in printed banner

- db: Create a utility to get credentials from dbconfig
- debian: Ensure that gbp creates a clean tarball prior to build
- deluge: Use OpenID Connect instead of pubtkt based SSO
- doc/dev: Set new theme for developer documentation
- doc/dev: Use OpenID Connect instead of pubtkt based SSO
- doc/dev: always have an up-to-date copyright year
- ejabberd: Fix setting up certificates for multiple domains
- email: Use OpenID Connect instead of pubtkt based SSO
- featherwiki: Use OpenID Connect instead of pubtkt based SSO
- gitweb: Fix issue with running post init due to missing method
- gitweb: Use OpenID Connect instead of pubtkt based SSO
- js: When page load fails during install, show it to user
- letsencrypt: When copying certificate reset the umask reliably
- locale/bg: Fix several translations with HTML links (Bulgarian)
- locale/de: Fix several translations with HTML links (German)
- locale: Update translations for Albanian, Catalan, Chinese (Simplified Han script), Czech, French, German, Greek, Italian, Swedish, Tamil, Turkish
- matrixsynapse: Update apache config to proxy Synapse client API
- miniflux: Get credentials from dbconfig-common directly
- miniflux: Revert workaround for a packaging bug with DB connection
- mumble: murmurd renamed to mumble-server
- oidc: Style the page for authorizing an OIDC app
- pyproject: Use new format to specify licenses
- quassel: Explicitly set permissions on the domain configuration file
- rssbridge: Use OpenID Connect instead of pubtkt based SSO
- searx: Use OpenID Connect instead of pubtkt based SSO
- sharing: Use OpenID Connect instead of pubtkt based SSO
- sso: Merge into users module, drop pubtkt related code
- syncthing: Use OpenID Connect instead of pubtkt based SSO
- syncthing: tests: Fix tests by allowing rapid restarts
- templates: Allow building pages without navigation bar and footer
- tests: functional: Fix expecting FreedomBox to be home page
- tests: functional: Fix reloading error page during install/uninstall
- tests: functional: Increase systemd rate limits for starting units
- tiddlywiki: Use OpenID Connect instead of pubtkt based SSO

- transmission: Use OpenID Connect instead of pubtkt based SSO
- ui: Add animation for notification dismissal
- ui: Dismiss notifications without page reload
- ui: Refactor notification delete buttons to avoid repeating code
- web_framework: Allow FreedomBox apps to override templates
- web_server: Log requests to WSGI app
- wireguard: Accept/use netmask with IP address for server connection
- wireguard: Fix format when showing multiple endpoints of the server
- wireguard: Fix showing default route setting in server edit form
- wireguard: Fix split tunneling
- wireguard: Show status of default route in server information page
- wireguard: filter .local addresses from showClient view
- wireguard: show server vpn ip in show client page
- wordpress: Use OpenID Connect instead of pubtkt based SSO when private

10.4 FreedomBox 26.3 (2026-02-02)

10.4.1 Highlights

- ui: Use HTMX to eliminate full page reloads
- wireguard: Add 'Start Server' button to help with client setup

10.4.2 Other Changes

- debian: Follows policy 4.7.3
- debian: Ignore lintian warning: service file missing Install section
- debian: Remove default Rules-Requires-Root
- debian: Remove preinst script
- debian: Update copyright years
- docs: Update container script usage
- lintian: Remove mismatched overrides
- locale: Update translations for Chinese (Simplified Han script), German, Italian, Turkish
- Makefile: Fix removing extra license file
- ui: Add HTMX as a dependency
- ui: Use HTMX to update notifications on partial page updates
- wireguard: Remove NM connections when app is uninstalled
- wireguard: Show next available client IP in Add Client form
- wireguard: Update functional tests to handle Start Server button
- wireguard: Show server endpoint on main app page

10.5 FreedomBox 26.2 (2026-01-20)

- gitweb: Fix deleting last repo disables app
- locale: Update translations for Estonian, Italian
- notifications: Close dropdown when clicking outside

10.6 FreedomBox 26.1 (2026-01-05)

- container: Add aliases for start/stop commands
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, Estonian, French, German, Italian, Russian, Turkish, Ukrainian

10.7 FreedomBox 25.17.1 (2025-12-15)

10.7.1 Highlights

- backups: Set proper permissions for backups-data directory
- minetest: Rename Minetest to Luanti

10.7.2 Other Changes

- locale: Update translations for Albanian, Bulgarian, Czech, Estonian, Hindi, Italian, Russian, Turkish, Ukrainian
- minetest: Remove legacy code, use new name, conf, etc.
- transmission: Deal with changes in latest forky package
- transmission: Remove obsolete apache redirects
- vagrant: Enable EFI firmware

10.8 FreedomBox 25.17 (2025-12-08)

10.8.1 Highlights

- ui: Implement a toggle menu for setting dark mode

10.8.2 Other Changes

- homeassistant: Fix spelling in tests
- locale: Fix python-brace-format error in Estonian
- locale: Update translations for Catalan, Estonian, French, Russian
- privileged: Don’t log exception that are expected
- ui: Dark theme color for tags text in an app card
- ui: Drop colors defined in Bootstrap 5.3
- ui: Fix dark theme color for disabled form elements
- ui: Fix dark theme color for form help text

- ui: Fix dark theme color for select-all button
- ui: Fix dark theme colors for app enable/disable toggle button
- ui: Fix dark theme colors for default button style
- ui: Fix dark theme colors for running status indicator
- ui: Minor CSS refactor to use variables
- ui: Use default button style for tag buttons
- upgrades: Use bootstrap spinner button instead of custom styling

10.9 FreedomBox 25.16 (2025-11-24)

10.9.1 Highlights

- dynamicdns: Use only IPv4 for GnuDIP protocol
- janus: Allow app to be installed from Debian unstable
- jsxc: Fix missing dependencies

10.9.2 Other Changes

- locale: Update translations for Albanian, Chinese (Simplified Han script), Czech, German, Italian, Russian, Turkish, Ukrainian
- janus: Relax content security policy for the video room
- janus: Update the video room code from latest upstream
- jsxc: Update content security policy to prevent style errors
- middleware: Implement middleware for common headers such as CSP
- package: Prevent freedombox's deps from removal during app uninstall

10.10 FreedomBox 25.15 (2025-11-10)

10.10.1 Highlights

- help: Fix serving images from help pages
- matrixsynapse: Explain federation and link to testing tool

10.10.2 Other Changes

- locale: Update translations for French, German, Italian
- main: Allow setting development mode from environment
- matrixsynapse: Clarify how to change domain name in status section
- matrixsynapse: Explicitly set the trusted key server to matrix.org
- README: Use the Weblate's language chart widget
- Run service using systemd even for development
- ttrss: Remove app not available in Trixie
- views: Implement an API to retrieve the readiness status in JSON

10.11 FreedomBox 25.14 (2025-10-27)

- Enable private tmp and join namespaces for the two daemons
- locale: Update translations for French, Japanese

10.12 FreedomBox 25.13.1 (2025-10-20)

- locale: Update translations for Albanian, German

10.13 FreedomBox 25.13 (2025-10-06)

10.13.1 Highlights

- backups: Fix robust handling of known errors
- debian: Stop privileged service during upgrade or removal
- miniflux: Fix DB connection issues during install/uninstall

10.13.2 Other Changes

- : Collect output for all privileged sub-processes
- : Use action_utils.run instead of subprocess.call
- : Use action_utils.run instead of subprocess.check_call
- : Use action_utils.run instead of subprocess.check_output
- : Use action_utils.run instead of subprocess.run
- action_utils: Handle capture_output argument in run wrapper
- actions: Fix lifetime of thread local storage
- actions: Log full exception from privileged daemon on error
- actions: Log method arguments in privileged daemon
- actions: Raise an exception if privileged server response is empty
- actions: Work with older privileged daemon
- actions_utils: Fix issue with collecting stdout/stderr
- backups: Don’t show enable/disable button as app can’t be disabled
- backups: Ignore a typing error with mypy
- ci: Switch backports test to trixie-backports
- config: Set home page to FreedomBox for invalid values
- d/rules: Drop a workaround for dh_installsytemd needed for /usr/lib
- daemon: When ensuring running state handle not-installed state
- diagnostics: In development mode, run diagnostics more rarely
- doc: Add manual page for freedombox-cmd
- gitweb: Use Git credential helper when cloning URLs with credentials

- glib: Add schedule parameter for setting interval in develop mode
- locale: Fix a string formatting issue in Italian translation
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, German, Italian, Russian, Turkish, Ukrainian
- Makefile: Move privileged daemon to /usr/lib/freedombox
- privileged_daemon: Fix showing errors for freedombox-cmd command
- privileged_daemon: Implement handling termination signal
- setup: Log full exception traceback when setup fails
- storage: Fix disk usage checking with disconnected SSH mounts
- zoph: Additional dbconfig configuration keys

10.14 FreedomBox 25.12 (2025-09-22)

10.14.1 Highlights

- views: Add a menu entry and view for showing logs of an app
- zoph: Fix cases when CLI user is set to "autodetect"

10.14.2 Other Changes

- api: Fix showing icons for custom shortcuts
- app: Add ability to retrieve logs from all systemd units of an app
- bepasty: tests: Fix functional test failure to missing / in URL
- bepasty: tests: Override backup/restore test properly
- ci: Don't include defunct extract-source job
- ci: Update container for functional tests from bookworm to trixie
- ci: functional: Provision the privileged daemon properly
- diagnostics, help: Link to diagnostics page logs in 500 error page
- index: Generalize showing custom shortcut icons
- locale: Update translations for Chinese (Simplified Han script), Czech, Dutch, German, Italian, Russian, Turkish, Ukrainian
- torproxy: When disabling apt over tor fails, report error properly
- upgrades: Multiple fixes for parsing Apt's Deb822 style sources
- upgrades: Overwrite apt sources lens with bug fixes
- views: Fix 'internal server error' when showing 404 page

10.15 FreedomBox 25.11 (2025-09-08)

10.15.1 Highlights

- email: Fix Thunderbird auto configuration failure
- matrixsynapse, upgrades: Install select packages from unstable

10.15.2 Other Changes

- action_utils: Implement a utility to run a command as different user
- actions, backup: Implement raw output for privileged daemon
- actions, backups: Fix tests depending on sudo based actions
- actions: Drop feature to run privileged action as another user
- actions: Drop sudo based privileged actions
- actions: Framework for capturing stdout/stderr in privileged daemon
- actions: Simplify raw output code in privileged methods
- backups: Trim translatable string
- container: Add support for Trixie as stable distribution
- container: Pass --nvram option to virsh undefine
- d/control: Remove libpam-abl as a recommendation
- diagnostics: Fix notification severity when skipping tests
- dynamicdns: Handle showing errors from GnuDIP
- gitweb: Don't use privileged action feature to run as different user
- gitweb: Use pathlib API more
- HACKING.md: Mention using virtual machines instead of containers
- homeassistant: Fix typo in description
- l10n: Fix a broken string in Russian translation
- l10n: Fix broken Italian translation
- l10n: Fix formatted strings for errors/exceptions
- locale: Update translations for Bulgarian, Chinese (Simplified Han script), Czech, Dutch, French, German, Italian, Russian, Turkish, Ukrainian
- Makefile, settings: Use full paths in pot files
- package: Don't consider uninstalled packages as available
- package: Refresh apt cache if sources list is newer
- power: logind config to ignore laptop lid close
- privileged: Don't isolate /var/tmp/ for privileged daemon
- privileged_daemon: Introduce a command line client for the API
- privileged_daemon: Log only to journal and not console
- samba: Update client list
- storage: Don't use privileged action feature to run as different user
- syncthing: Update Android clients to Syncthing-Fork
- tests: functional: Fix incorrect skipping of install tests
- ui: Fix missing semicolon in JS file
- upgrades: Don't allow needs-restart to restart privileged daemon
- upgrades: Overhaul detection of distribution
- web_server: Restart in development mode only for source code changes

10.16 FreedomBox 25.10 (2025-08-18)

10.16.1 Highlights

- email: Add full text search capability
- homeassistant: Add the most popular app for home automation

10.16.2 Other Changes

- : Update copyright year
- action_utils: Add support for device mappings in podman containers
- action_utils: Allow checking result for service operations
- actions: Add method to handle privileged JSON request to a server
- actions: Allow logging privileged calls to a server differently
- actions: Call to a privileged server if it is available
- actions: Raise a proper exception if privileged module is not found
- actions:privileged: Fix flake8 errors
- apache, letsencrypt: Create a site specific config for all domains
- apache: Add component to host an app on a site's root
- apache: Allow customization to the root of a site
- apache: Generalize TLS config to use certificates only if available
- apache: Make all directory listings more mobile friendly
- app: Don't load apps again
- apps: Fix regression in enabling install button
- config: Allow better Apache default home page
- container: Add component for managing containers using podman
- d/control: Drop dependency on python3-setuptools-git
- debian, pyproject.toml: Remove ancient python3-tomli workaround
- debian: Print warning if version mismatch
- diagnostics: Add collapsible sections for results
- doc: Update man page with option --version
- dovecot: Allow login using email address with dovecot version 2.4
- dovecot: Fix location of Inbox in dovecot 2.4
- dynamicdns: Add info about subdomains with Foundation's service
- dynamicdns: Fix a type check error
- dynamicdns: Switch update client to HTTP protocol
- email: Increment app version number to trigger dovecot restart

- featherwiki: Disable caching to avoid 412 errors
- featherwiki: Don’t allow index.html as a file name
- forms: Allow showing a None option during domain selection
- letsencrypt: Drop old app migration code
- locale: Update translations for Catalan, Dutch, Estonian, Tamil
- Makefile: Developer setup for privileged daemon
- matrixsynapse: Drop help text for domain selection dropdown
- module_loader: Don’t load modules again
- names: Add information about adding static domains/subdomains
- notifications: Add datetime to each notification
- notifications: Fix setting last_update_time
- notifications: Minor styling fix for operation notifications
- package: Refresh apt cache if old and some packages are not found
- package: Uninstall packages even if they are not in apt cache
- plinth: Add --version flag to print version and exit
- privileged: Add /bin file for privileged daemon
- privileged: Add systemd units for socket activating the daemon
- privileged: Introduce a UNIX socket daemon for privileged calls
- pyproject: Add missing pytest marker for "tags"
- setup: Perform a check for app availability after the page loads
- sogo: Allow adding IMAP accounts and changing full name
- templates: Show Launch web client button for all FreedomBox URLs
- tests: Ensure that privileged daemon is not used during tests
- tiddlywiki: Avoid "412: Precondition failed" error
- tiddlywiki: Don’t allow index.html as a file name
- ui: Fix user menu popup overflowing beyond the page
- ui: Use system’s UI fonts on all platforms instead of Lato
- uninstall: Use reverse order when uninstalling components
- views: Remove unused code related to refresh packages button

10.17 FreedomBox 25.9.4 (2025-08-02)

- distupgrade: Handle comments in sources.list file
- distupgrade: Update Trixie’s release date as announced

10.18 FreedomBox 25.9.3 (2025-07-21)

- config: Allow overriding target path in dropin config component
- email: Add support for Dovecot 2.4
- email: Start servers during re-setup if they are not running
- locale: Update translations for Albanian, Bulgarian, Catalan, Estonian, Tamil
- sogo: Fix typo in configuration for sieve server

10.19 FreedomBox 25.9.2 (2025-06-23)

- debian: Set the branch for trixie
- debian: Update debconf Portuguese translation
- locale: Add new translation for Estonian
- locale: Update translations for Dutch, German, Hungarian
- performance: Handle install for trixie

10.20 FreedomBox 25.9.1 (2025-06-02)

- locale: Update translations for Catalan, Telugu, Ukrainian

10.21 FreedomBox 25.9 (2025-05-05)

- locale: Add translation for Catalan
- locale: Update translations for French, Hindi, Telugu
- minetest: Create the configuration directory if necessary for Trixie
- minetest: Work with new luanti binary in Trixie

10.22 FreedomBox 25.8 (2025-04-21)

10.22.1 Highlights

- users: Fix unable to delete user
- zoph: Don’t fail while uninstalling

10.22.2 Other Changes

- bind: Keep configuration during distribution upgrades
- container: Fix waiting until plinth setup is finished when running tests
- distupgrade: Use new configuration file instead of halting upgrade
- locale: Update translations for Albanian, Arabic, Chinese (Simplified Han script), Czech, German, Hindi, Portuguese, Telugu, Turkish, Ukrainian
- sogo: Adjust apache configuration to work on Trixie
- upgrades: Allow packages to be upgraded from stable-updates
- upgrades: Cleanup use of return value from _apt_run
- zoph: Don’t use mod-php instead continue to use php-fpm

10.23 FreedomBox 25.7 (2025-04-07)

10.23.1 Highlights

- calibre: Remove existing data directory before a restore operation
- kiwix: Remove existing data directory before a restore operation
- mediawiki: Run database upgrade after package upgrade
- power: Disable sleep and hibernate on the system
- upgrades: Revamp distribution upgrade UI

10.23.2 Other Changes

- app: During startup, run post-init operation in background
- backups: Add ability to cleanup files before restoring a backup
- debian: Build depend on python3-systemd
- locale: Update translations for Bengali, French, Hindi, Russian, Telugu
- main: Run package operations immediately after startup
- mediawiki: Update configuration to work with version 1.4 (Trixie)
- mypy: Ignore missing type stubs for systemd.*
- package: Allow app to say it wants to rerun setup after updates
- packages: Don't run force upgrade hooks when freedombox performs ops
- radicale: Explicitly set the auth type to accommodate radicale 3.5
- radicale: Rerun setup when package updated
- service: Increase startup/shutdown timeout to 5 minutes
- service: Notify systemd when service starts up
- setup: Implement mechanism to rerun setup when apt is updated
- setup: Perform package operations immediately on trigger
- tests: Fix a warning with pytest setup
- tests: django: Fix a warning related to timezone settings
- upgrades: Add introduction to the distribution upgrade page
- upgrades: Don't ignore apt error during distribution upgrade
- upgrades: Increase distribution upgrade timeout to 3 days
- upgrades: Prevent installation of the Samba Active Directory service
- upgrades: Remove unused import
- upgrades: Run distribution upgrade around 06:00 everyday
- upgrades: Show notification before, during, and after a dist upgrade
- upgrades: Trigger special package operations in a simpler way
- upgrades: Use special desc. for snapshots take before dist upgrade

10.24 FreedomBox 25.6 (2025-03-25)

10.24.1 Highlight

- names: Store domains in kvstore instead of /etc/hosts

10.24.2 Other Changes

- container: Allow taking snapshots of VMs
- container: Don't remove qcow2 image when terminating VM
- doc: Fetch latest manual
- locale: Update translations for German
- operation: Drop type annotations on enum members
- setup: Fix a minor flake8 complaint
- SOGo: add dpkg-dev to the package list
- storage: tests: Fix a test failure for psutils >= 7.0
- syncthing: Extend setup process to recover from manual config errors
- tests: functional: Add enable-all-apps script
- upgrades: Run dpkg/apt fixes before dist upgrade

10.25 FreedomBox 25.5 (2025-03-10)

10.25.1 Highlights

- ui: tags: Add tag search/filter for system page

10.25.2 Other Changes

- action_utils: Ensure that package are unheld if dist upgrade fails
- locale: Update translations for Ukrainian
- setup: Fix issue with pending app update and force upgrade
- sogo: Fix a typo in tags
- ui: Minor change with renaming a variable
- ui: system: When canceling search stay on current page
- ui: tags: Minor refactoring in menu filtering and sorting
- ui: tags: Redirect to apps or system page appropriately
- ui: tags: Show tags on all cards pages if present
- upgrades: Drop special handling for searx
- upgrades: Log apt output to journal during dist upgrade
- upgrades: Log messages using python logging framework

- upgrades: Minor refactor to pre-dist upgrade checks
- upgrades: Perform easier checks first during dist upgrade
- upgrades: Perform sources file update more reliably
- upgrades: Refactor code for disabling Quassel during dist-upgrade
- upgrades: Refactor code to disable snapshots
- upgrades: Refactor code to hold packages
- upgrades: Refactor code to retrieve the new codename
- upgrades: Relax list of packages to hold during dist upgrade
- upgrades: Simplify dist upgrades checks using exceptions
- upgrades: Simplify some global names
- upgrades: Split dist upgrade into a separate module
- upgrades: Split the main dist upgrade code
- upgrades: Use bind mounts to edit sources file only upon completion
- upgrades: Use systemd service status instead of flag file
- upgrades: Use systemd-run to create transient service
- upgrades: tests: Add unit tests for dist upgrade methods

10.26 FreedomBox 25.4.1 (2025-03-02)

- debian: Move e2fsprogs to Recommends
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, German, Turkish

10.27 FreedomBox 25.4 (2025-02-24)

10.27.1 Highlights

- dynamicdns: Implement adding multiple domains
- names: Allow adding multiple static domain names

10.27.2 Other Changes

- container: Fix spelling of 'destroy'
- dynamicdns: Add/remove domains when app is enabled/disabled
- dynamicdns: Handle addition of duplicate domains
- dynamicdns: List domains that have not had status update yet
- dynamicdns: Show specific operations on domains in Names apps
- locale: Update translation for Telugu
- matrixsynapse: tests: Fix adding domain
- names: Add more URLs to the domain type component

- names: Handle addition of duplicate static domains
- names: Retrieve the most important domain in a more generic way
- names: Simplify showing current and available domains
- names: Simplify/clarify domain names types for static/dynamic
- names: Sort domains by priority of their domain types
- names: Specify priority order among domain types
- names: Use hostnamectl instead of socket API get the current hostname
- names: domain type: Minor refactor
- pyproject: Ignore missing types for numpy (needed by pytest)
- upgrades: Drop unattended-upgrade call from dist-upgrade
- upgrades: Inhibit shutdown during dist-upgrade

10.28 FreedomBox 25.3.1 (2025-02-15)

- email: fix fresh installation issue

10.29 FreedomBox 25.3 (2025-02-10)

- debian: Switch from dnsutils to bind9-dnsutils
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Chinese (Traditional Han script), Czech, Dutch, German, Turkish

10.30 FreedomBox 25.2 (2025-01-27)

10.30.1 Highlights

- gnome: Add app to provide a graphical desktop environment
- sogo: Add a new app for SOGo groupware

10.30.2 Other Changes

- action_utils: Implement methods to get/set the systemd boot target
- apache: Enable expires module by default
- d/control: Fix incorrect comment format
- d/copyright: Fix copyright entry for GNOME app icons
- db: Add more utilities for managing PostgreSQL databases
- email: Make rspamd learn spam/ham when the user marks mails as junk or not junk.
- email: Recommend Thunderbird Mobile instead of K-9 Mail
- firewall, networks, storage: Drop polkit pkla files
- gnome: Add changes missing from branch merge

- locale: Updated translations for Albanian, Bulgarian, Chinese (Simplified Han script), Chinese (Traditional Han script), Czech, Dutch, German, Turkish
- miniflux: Add management of postgresql daemon in a shared manner
- nextcloud: Drop some dead code
- nextcloud: tests: functional: Fix test for modified structure of UI
- nextcloud: tests: functional: Set override domain testing on VM
- ttrss: Add management of postgresql daemon in a shared manner
- ttrss: Fix issue with installing after uninstalling
- ui: Drop the temporary fix for missing popper.js 2.0
- ui: Fix missing arrow on dropdown form fields
- ui: Fix tag separator not showing on some machines
- wordpress: Order daemon enable/disable correctly

10.31 FreedomBox 25.1 (2025-01-13)

10.31.1 Highlights

- email: Show DNS entries for all domains instead of just primary
- privacy: Add option in UI to set lookup URL for public IPs

10.31.2 Other Changes

- app: Add tags to menu and frontpage components
- app: Allow apps to instantiate without Django initialization
- app: Stop showing short description on installation page
- apps: Only show app tags all the tags in apps page search box
- backups: Add warning that services may become unavailable
- backups: Handle error when there is not enough space on disk
- backups: Make all generated archive names consistent
- backups: Properly cleanup after downloading an archive
- deluge: tests: functional: Fix deluge client logged in detection
- doc: dev: Remove short description and add tags to all components
- dynamicdns: Use the public IP lookup URL from privacy app
- email: Create DKIM keys for all known domains
- email: Fix regression error when installing/operation app
- email: Show reverse DNS entries to be configured
- locale: Translated using Weblate (German, Russian)
- miniflux: Ignore an type check error with pexpect library

- privacy: Introduce utility to lookup external IP address
- privacy: Show notification for privacy settings again
- ui: Don't place JS file at the bottom of the page
- ui: Drop remnants of already removed background images
- ui: Fix missing variables in Bootstrap 5.2/Debian stable
- ui: Update section header style to increase size, remove underline
- ui: js: Load all JS files in deferred mode to speed up page load
- views: Use tags from menu or shortcut instead of the app
- web_framework: Disable caching templates files in development mode

10.32 FreedomBox 24.26.1 (2025-01-05)

10.32.1 Highlights

- email: Fix DKIM signing by setting correct ownership on private keys
- nextcloud: remove experimental warning

10.32.2 Other Changes

- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, French, German, Turkish
- mumble: Add diagnostic for setup config changes
- mumble: Support config file moved into /etc/mumble
- sharing: Drop jQuery code as the library dependency has been removed
- users: Drop jQuery code as the library dependency has been removed

10.33 FreedomBox 24.26 (2024-12-30)

10.33.1 Highlights

- apps: Replace short description with tags in apps list
- ui: Show breadcrumbs on deeper pages
- ui: Handle and show most page load errors as alerts

10.33.2 Other Changes

- actions: Allow privileged methods to be decorated again
- backups: Fix issue clicking on schedule buttons with Bootstrap 5
- backups: Fix issue with verifying SSH hosts with RSA key
- backups: Fix issue with verifying remote server identity
- backups: Format better when showing archive time delete page
- backups: Handle common errors during borg operations

- backups: Parse borg errors from all operations and not just some
- backups: Remove unused import contextlib
- backups: Require POST method for mount/unmount operations
- backups: Use ISO timestamp for auto-naming archives
- backups: tests: functional: Wait for pages to load after click
- container: Add support for VMs using libvirt
- container: Generalize language in output messages for VMs
- container: Minor refactoring to reduce repeated code
- container: Refactor nspawn specific operations into a separate class
- container: Update FSID inside the image file to keep it bootable
- context_processors: Use breadcrumbs to highlight current section
- debian: Remove dependency libjs-jquery
- diagnostics: Use generic handler to handle exceptions in diagnostics
- frontpage: Replace short description with tags
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, German, Tamil, Telugu, Turkish
- Makefile: Add i2p to list of apps to remove
- mediawiki: tests: functional: Use click function from functional library
- menu: Ensure that all menu items have names for use by breadcrumbs
- menu: Implement a helper method to lookup menu items using URL name
- middleware: Handle method not allowed errors and redirect
- middleware: Handle page not found errors specially
- minetest: Provide default gameid argument
- miniflux: tests: functional: Use helper functions from functional library
- networks: Fix error during creation of PPPoE connections
- samba: tests: functional: Wait for page update after enable/disable share
- sharing: tests: functional: Use click function from functional library
- system: Add tags to all remaining apps
- tags: Add button to clear all tags
- tags: Add tags to system apps
- tags: Remove short description from system apps
- tags: Replace short description with tags in app pages
- tests: functional: Add utility to click element wait for page update
- tests: functional: Fix typos in diagnostics checks
- tests: functional: Make first wizard run more robust
- tor, torproxy: Fix daemon services are running after reboot when app is disabled

- torproxy: Don't disable apt-transport-tor in setup
- ui: Don't show breadcrumbs in login and first wizard pages
- ui: Fix regression with margin above app title
- ui: Replace use of jQuery with plain JavaScript
- users: Restart nslcd service after configuration changes during setup
- users: tests: functional: Use click function from functional library
- views: Implement retrieving breadcrumbs of a page
- views: Show exception details with the utility to show errors
- zoph: Include tags from the manifest

10.34 FreedomBox 24.25 (2024-12-16)

10.34.1 Highlights

- deluge: Fix app installation on Debian testing
- notifications: Don't error when dismissing missing notifications
- ui: Switch to using bootstrap 5

10.34.2 Other Changes

- action_utils: Add utility to ensure that service is stopped
- apache: Relax content security policy to allow data: URL
- ci: Add flake8 to gitlabci container
- d/control: Drop unnecessary recommendations
- d/control: Drop version specification on dependencies for Bookworm
- debian: Added po-debconf Catalan translation
- deluge: Cleanup and simplify setup code
- diagnostics: Fix trimming an i18n string
- diagnostics: tests: functional: Pick errors more specifically
- firewalld: Reduce severity for alert about service on internal zone
- help: Add a class to the help index page
- help: Update the privacy notice on status log page
- i2p: Drop app as it has not been available in Debian for a while
- letsencrypt: Declare explicit dependency on openssl
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, French, German, Russian, Turkish
- matrixsynapse: Fix trimming an i18n string
- networks: Fix issue with loading create PPPoE form
- networks: Fix minor typo in template related to signal strength

- networks: Use new accordion component from Bootstrap 5
- openvpn: Declare explicit dependency on openssl
- power: Refactor display of package manager busy alerts
- radicale: Update link to supported clients
- sso: Switch using cryptography module instead of OpenSSL.crypto
- tags: Fix issue with JS init on a page without tags
- tags: tests: Fix deprecated webdriver attribute
- tests: functional: Disable smooth scrolling from Bootstrap 5
- tests: functional: Don't try disable app after tests if app was not installed
- tests: functional: Fix app installation test skipped on slow machines
- ui: Accept default styling of Bootstrap 5 for warning button
- ui: Align app icons in the center for home, apps, and help pages
- ui: Allow underlining for most links due to Bootstrap 5
- ui: Allow users to provide a CSS file to customize styling
- ui: Cleanup use of colors with CSS variables
- ui: Don't use nav-link inside card
- ui: Drop use of badge-* utility classes for newer replacements
- ui: Fix flash of notifications popdown during page load
- ui: Fix issue with notifications icon showing twice in mobile view
- ui: Fix overflow of exception text in message
- ui: Fix placement of tags menu under tags input with Bootstrap 5
- ui: Fix styling in dismiss button in form errors for Bootstrap 5
- ui: Fix toggle button styling for Bootstrap 5
- ui: Reduce the gap at the top of the pages
- ui: Rename data- attributes to data-bs- for Bootstrap 5
- ui: Restore spacing between form elements in Bootstrap 5
- ui: Restyle tags, remove underlining of text
- ui: Show disabled form elements as grey for Bootstrap 5
- ui: Style the 'Log out' item properly
- ui: Update custom button styles for Bootstrap 5
- ui: Update styling for navbar menu items
- ui: Use Bootstrap 5 styling for all alerts
- ui: Use collapse instead of dropdown for notification in Bootstrap 5
- ui: Workaround dropdowns not working with Bootstrap 5
- ui: app, system: Revert to earlier width for card lists

- ui: app: Fix an incorrect HTML tag nesting
- ui: backups: Drop unused styling in verify host key page
- ui: backups: Use Bootstrap color variables instead of static values
- ui: diagnostics: Fix gap between headings
- ui: diagnostics: Fix layout of repair buttons
- ui: firewall: Fix styling with Bootstrap 5
- ui: forms: Fix margins for form labels for Bootstrap 5
- ui: help: Fix alignment issue with footer links in about page
- ui: samba: Fix layout regressions with Bootstrap 5
- ui: snapshots: Fix horizontal form layout on mobiles for Bootstrap 5
- ui: snapshots: Fix horizontal form styling margins for Bootstrap 5
- ui: users: Fix close button for confirmation dialog for Bootstrap 5

10.35 FreedomBox 24.24 (2024-11-18)

10.35.1 Highlights

- nextcloud: Switch to using FreedomBox container registry
- networks: Overhaul Wi-Fi network scan page

10.35.2 Other Changes

- container: Quote arguments that contain spaces when restoring pytest args
- i18n: Fix translation of FreedomBox name in various places
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, German, Turkish
- networks: Fix display of mangled SSIDs when scanning Wi-Fi networks
- networks: Fix display of strength and channel for Wi-Fi connections
- networks: Fix editing wireless connections with SSID field
- networks: Improve styling of badges in the information tables
- networks: wifi: In new connection page set form defaults properly
- nextcloud: Drop diagnostic for checking availability of docker.com
- rssbridge: Mention miniflux in app description similar to tt-rss
- tests: functional: Don’t enable/disable app during tests
- tests: functional: Fix visit() returning prematurely before page load
- tests: functional: Wait for uninstall page load before uninstalling
- ui: Drop box shadow around the card in card lists
- ui: Drop logo backgrounds for index, apps, and system pages

10.36 FreedomBox 24.23 (2024-11-04)

- backups: Better explanation for the format of upload file
- backups: Sort list of apps in backup, restore, and schedules
- base: Add link to about for unauthenticated users
- debian: Drop python3-flake8 build dependency
- ejabberd: Set mod_mam default to always
- functional tests: Add pytest testinfra plugin
- help: Add all footer links to about page
- help: Make about page available to unauthenticated users
- index: Remove links and about text and link to about page
- index: css: Move the 'powered by' logo further down
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, French, German, Norwegian Bokmål, Spanish, Turkish
- middleware: tests: Drop some obsolete mock code
- system: Increase the size of items in listing page
- ui: Don't bold titles in card lists
- ui: Increase the width of app and system listings
- ui: Move app names below app icons
- ui: Remove the border around content container
- ui: Remove the noise background
- users: Delete or move home folder when user is deleted or renamed
- users: tests: functional: Check LDAP information is correct after renaming user

10.37 FreedomBox 24.22 (2024-10-21)

10.37.1 Highlights

- Add tags for apps

10.37.2 Other Changes

- .gitlab-ci.yml: Update for new infrastructure
- Remove unused imports to fix flake8 errors
- apache2: Allow popups to have different sandbox policy
- backups: Use new utility for handling file uploads
- calibre: tests: functional: Fix occasional failure in add book test
- ci: Add a custom driver for gitlab runner for podman
- ci: Add docker container for functional-tests:stable

- ci: Add gitlab runner configuration
- ci: Dockerfile: Drop obsolete dependency on pytest-bdd
- ci: Enable showing test failures immediately as they fail
- ci: Generalize script to update container, switch to podman
- ci: Rename Dockerfiles to Containerfiles
- ci: Update functional test timeout to 10h
- context_processors: Stop adding unused 'submenu' to context
- context_processors: Use active menu urls to decide what to highlight
- css: Fix height of navbar in mobile layout during first boot
- css: Navbar styling fixes in mobile layout
- first_boot: Add notification for next steps after first setup
- first_boot: Allow the next steps page to be revisited
- firstboot: Hide navigation toggler in mobile layouts
- firstboot: Improve the setup complete page with more setups
- firstboot: Make logo image responsive during first setup
- firstboot: Show spinner instead of message during first setup
- help, system: Stop using submenu.sorted_items
- help: tests: Fix tests failing due to tags related changes
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, Spanish, Turkish, Ukrainian
- middleware: Show translated error messages when operation completes
- networks: Remove first boot steps for connectivity/topology
- nextcloud: Fix install failure due to PrivateTmp=yes
- operation: Use safe formatter for translating messages
- setup: Translate errors when installing/updating/repairing apps
- ssh: Start server after nslcd service
- syncthing: Fix app setup in Debian testing
- tags: Adjust tags and style
- tags: Localization fixes
- tags: css: Minor styling cleanups
- tags: js: Minor fixes and refactoring
- tests: functional: Add package for printing test failures instantly
- tests: functional: Create utility to set user preferred locale
- upgrades: Remove step upgrade during first setup
- upgrades: Show notification to remind user to run updates manually
- utils: Improve safe formatter by handling more cases
- wordpress: tests: functional: Fix tests on Trixie

10.38 FreedomBox 24.21 (2024-10-07)

10.38.1 Highlights

- django: Improve handling of file uploads
- nextcloud: Fix existing installs to upgrade properly

10.38.2 Other Changes

- action_utils: Fix missing parent folder when creating apt hold flag
- action_utils: Introduce utility to move uploaded files
- actions: Handle exceptions with Path-like objects
- featherwiki: Use new utility for handling uploads
- functional tests: Upgrade dependencies selenium and splinter
- kiwix: Don't leave invalid .zim in library after a failed attempt
- kiwix: Use new utility for handling uploads
- locale: Albanian: Fix build issue by correcting typo
- locale: Update translations for Albanian, Czech
- names: Don't control resolved daemon when package is not installed
- names: Don't hard depend on systemd-resolved
- names: Don't show resolver status if package is not installed
- names: Introduce method to check if resolved is installed
- names: Perform resolve diagnostics only if resolved is installed
- names: Schedule a task to install systemd-resolved when possible
- names: Try to install systemd-resolved during app setup
- networks: Disable DNS-over-TLS option if resolved is not installed
- networks: Don't show DNS-over-TLS when resolved is not installed
- nextcloud: During upgrade wait properly for upgrade to complete
- privacy: Disable DNS fallback option if resolved is not installed
- samba: Remove option to backup app
- storage: Skip tests that involve loopback device in a container
- tiddlywiki: Avoid writing duplicate temporary files
- transmission: tests: Fix functional test failures in Debian testing

10.39 FreedomBox 24.20.1 (2024-09-25)

10.39.1 Highlights

- samba: Fix nmb systemd service is in erroneous state
- users: Inactivate users in LDAP user database

10.39.2 Other Changes

- Revert "debian: tests: Wait for systemd-resolved to be started"
- ci: Run autopkgtest but allow failure
- d/tests: Add breaks-testbed restriction
- locale: Update translations for Bulgarian, Chinese (Simplified Han script), Spanish, Turkish
- security: Remove PAM configuration for 'access' module
- users: Increment app version for changes w.r.t. inactive users
- users: Set proper class on default password policy object

10.40 FreedomBox 24.20 (2024-09-23)

10.40.1 Highlights

- nextcloud: Fix issue with upgrading to next version

10.40.2 Other Changes

- action_utils: Add a method to reset services in 'failed' state
- action_utils: Update outdated docstrings
- apache: Don't restart daemon when changing certificates
- config, names: Move domain name configuration to names app
- config, names: Move setting hostname from config to names
- debian: tests: Wait for systemd-resolved to be started
- letsencrypt: Allow reloading daemons after cert changes
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified Han script), Czech, Dutch, Spanish, Turkish
- miniflux: Workaround a packaging bug with DB connection
- service: Add privileged utility for 'try-reload-or-restart' action
- tests: functional: Don't timeout when web server restarts
- upgrades: Treat n/a release as testing
- users: Don't cache NSS user identity information
- users: Invalidate nscd cache after nslcd service startup

10.41 FreedomBox 24.19 (2024-09-09)

10.41.1 Highlights

- mediawiki: Increase PHP maximum execution time to 100 seconds
- names: Use systemd-resolved for DNS resolution

10.41.2 Other Changes

- bind: Don't start a stopped daemon during changes/upgrades
- bind: Fix port number clash with 'shared' network connections
- bind: Set default forwarder as systemd-resolved
- container: Re-run failed provisioning even if container is running
- letsencrypt: Handle both standard and custom repairs
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified), Dutch, German, Spanish, Telugu, Turkish, Ukrainian
- makefile: Workaround problems with systemd-resolved package
- names, network: Re-feed DNS known to network-manager to resolved
- names: Add option for setting global DNS-over-TLS preference
- names: Add option for setting global DNSSEC preference
- names: Implement a diagnostic check for checking name resolution
- names: Restart instead of reload for systemd-resolved changes
- names: Show systemd-resolved status in the names page
- networks: Add more options for IPv6 configuration method
- networks: Add support for DNS-over-TLS for individual connections
- networks: Declare a need for DHCP/DNS ports to be open in firewall
- networks: Fix focusing on network interface field on error
- networks: Groups fields in create/edit connection form
- networks: Set 'auto' as default IPv6 method in new connection form
- networks: Show current global value of DNS-over-TLS and link to it
- privacy: Implement a way to disable fallback DNS servers
- privacy: Show notification again so that users see the new setting
- storage: Handle grub-pc package not available
- upgrades: Add repair step for held packages

10.42 FreedomBox 24.18 (2024-08-26)

- .md, pyproject.toml: Update default branch from 'master' to 'main'
- d/control: Don't recommend libnss-gw-name
- d/control: Remove haveged as it no longer relevant on latest kernels
- debian: Set gbp default branch to main
- doc/dev: Fix Django related errors with auto-documentation
- doc/dev: Limit table of contents depth to 2 for clarity
- doc/dev: Update copyright year
- featherwiki, tiddlywiki: Remove redundant </p> in template

- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified), Czech, Dutch, Norwegian Bokmål, Spanish, Telugu, Turkish, Ukrainian
- plinth: Fix translating app operations
- ttrss: Remove unavailable Android client - org.fox.tttrss
- upgrades: Add diagnostic for held packages

10.43 FreedomBox 24.17 (2024-08-12)

10.43.1 Highlights

- featherwiki: Add new app for Personal Notebooks
- tiddlywiki: Add new app for Non-linear Notebooks

10.43.2 Other Changes

- COPYING.md: Update copyright years
- actions: Add error when likely parameters are not marked as secret
- actions: Define and allow a new alias for str; secret_str
- actions: Log arguments without secret strings in privileged calls
- apache: Enable dav and dav_fs modules
- backups: Mark secret strings in privileged actions
- base.html: Drop dependency on modernizr.js library
- bepasty: Mark secret strings in privileged actions
- django: settings: Don't set USE_L10N on newer versions
- featherwiki: Fix a type check failure
- featherwiki: Proxy download through freedombox.org
- firewall: Setup inter-zone forwarding
- gitweb: Don't backup/restore a drop-in configuration file
- ikiwiki: Mark secret strings in privileged actions
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified), Czech, Spanish, Turkish
- makefile: Don't fail while building and installing multiple versions
- mediawiki: Mark secret strings in privileged actions
- miniflux: Mark secret strings in privileged actions
- mumble: Mark secret strings in privileged actions
- nextcloud: Mark secret strings in privileged actions
- nextcloud: Prevent process restart when nextcloud is uninstalled
- nextcloud: Redirect to URL nextcloud/ if ending slash is not given
- nextcloud: tests: functional: Add base and interface tests

- pagekite: Mark secret strings in privileged actions
- rssbridge: tests: Add missing `__init__.py`
- shadowsocks: Mark secret strings in privileged actions
- shadowsocksserver: Mark secret strings in privileged actions
- ssh: Mark secret strings in privileged actions
- storage: Add diagnostic for grub config issue
- templates: Fix warning about using default.html for form template
- tests: functional: Allow submitting form buttons along with inputs
- tests: functional: Name the background fixture
- users: Mark secret strings in privileged actions

10.44 FreedomBox 24.16 (2024-07-29)

10.44.1 Highlights

- miniflux: Add new app
  - Note that miniflux is currently only available in Debian testing and unstable.

10.44.2 Other Changes

- locale: Update translations for Bulgarian, Chinese (Traditional), German, Swedish
- miniflux, ttrss: Factor out duplicated postgres utility methods
- miniflux: Add list of client apps from upstream clients list
- miniflux: Drop postgres-contrib package as it seem redundant
- miniflux: Fix error handling for reset password
- miniflux: Fix issues with running the CLI in a pseudo-terminal
- miniflux: Fix regression in creating admin user
- miniflux: Remove a spurious error message after resetting password
- miniflux: Trim error messages when creating user
- miniflux: tests: functional: Fix failing tests when run out of order
- readme: Mention the AGPLv3+ license
- tests: functional: Handle click failure when waiting for page update
- tests: functional: Handle obscured elements when submitting forms

10.45 FreedomBox 24.15 (2024-07-16)

- locale: Update translations for Albanian, Chinese (Simplified), French, German, Polish, Spanish, Turkish, Ukrainian
- make: Ignore .mypy_cache folders while copying files

10.46 FreedomBox 24.14 (2024-07-01)

- container: Allow podman containers to run inside the container
- diagnostics: Add option for automatic repair
- locale: Update translations for French

10.47 FreedomBox 24.13 (2024-06-17)

- locale: Update translations for Czech, Dutch, Swedish, Ukrainian

10.48 FreedomBox 24.12 (2024-06-03)

- locale: Update translations for Albanian, Chinese (Traditional), German, Russian

10.49 FreedomBox 24.11 (2024-05-20)

- locale: Update translations for Chinese (Simplified), Chinese (Traditional), Spanish, Swedish, Turkish

10.50 FreedomBox 24.10 (2024-05-06)

10.50.1 Highlights

- nextcloud: Enable app with experimental warning
- minidlna: Add media directory selection form

10.50.2 Other Changes

- app, component: Add repair method
- diagnostics: Add optional component_id to DiagnosticCheck
- diagnostics: Change "Re-run setup" to "Try to repair"
- letsencrypt: Re-obtain certificates during repair
- letsencrypt: Remove unused imports
- locale: Update translations for Chinese (Simplified), Chinese (Traditional), Czech, Spanish, Turkish
- minidlna: Do not proxy minidlna web interface over Apache
- minidlna: Explicitly include ssdp service to firewall configuration
- minidlna: Restart app when upgrading to reconfigure firewall
- nextcloud: Add fallback for when quadlet is not available
- nextcloud: Allow re-running setup
- nextcloud: Allow re-running setup when app is disabled
- nextcloud: Create network using systemd generator
- nextcloud: Drop network namespacing in container, use host network
- nextcloud: Enable pretty URLs without /index.php in them

- nextcloud: Implement enable/disable container
- nextcloud: Populated and maintain a list of trusted domains
- nextcloud: Pull the image separately before starting systemd unit
- nextcloud: Restart container when dependent services are restarted
- nextcloud: Ship instead of create cron timer related units
- nextcloud: Use php-fpm container instead of apache container
- nextcloud: Use systemd generator for creating container service
- nextcloud: Wait on init sync lock
- nextcloud: Warn that community provides the container not team
- notification: Handle more formatting errors
- setup: Add method to run app repair
- storage: Add an option to include help text to directory selection form

10.51 FreedomBox 24.9 (2024-04-22)

- action_utils, nextcloud: Make podman util more generic
- config: Handle dropin config files with limited permissions
- locale: Update translations for Albanian, Chinese (Simplified), Chinese (Traditional), Russian, Spanish, Turkish
- nextcloud: (Note that this app is currently disabled, and not available to install.)
  - Add network interface to firewall zone after creating it
  - Add warning that the app is experimental
  - Allow backup/restore when app is disabled
  - Connect to mysql using socket instead of TCP
  - Connect to redis using Unix socket
  - Connect to slapd for LDAP using Unix socket
  - Don’t fail uninstall if DB or user do not exist
  - Don’t set password on redis server
  - Don’t show incorrect phone region when it is not set
  - Drop a repeated creation of systemd service file
  - Ensure that database is running when running setup
  - Expand on the warning about container usage
  - Fail on errors when configuring the app
  - Fix configuring trusted proxies setting
  - Improve check used to test if installation wizard is done
  - Improve database permission granting
  - Improve setting up LDAP configuration
  - Minor changes to cron timer units
  - Minor refactoring
  - Minor update to description regarding users’ usage
  - Refactor container creation code

  - Refactor setting admin password
  - Remove fail2ban jail, brute force protection present
  - Rename the systemd service names
  - Retrieve database password in a more robust way
  - Use a separate DB for redis
  - Use secrets module generate passwords
  - When backup fails, unset the maintenance mode
- tests: functional: Disable functional tests

10.52 FreedomBox 24.8 (2024-04-08)

10.52.1 Highlights

- gitweb: Fix an issue when cloning existing repository
- ttrss: Improve backup and restore of the database
- ttrss: Include dbconfig configuration file in backup
- zoph: Include dbconfig configuration file in backup

10.52.2 Other Changes

- action_utils: Add generic utils for managing podman containers
- actions: Don't repeat action traceback in stderr
- diagnostics: Don't store list of app objects with results
- locale: Update translations for Albanian, Chinese (Simplified), Dutch, French, German, Norwegian Bokmål, Spanish, Turkish
- network: Skip bridge interfaces in connections list
- nextcloud: Add new app based on podman container
  - Note: The Nextcloud app is currently disabled (not available to install). It will be made available after there has been more testing and review.
  - Add backup/restore
  - Add option to configure the default phone region
  - Configure redis caching, create static PHP file
- ttrss: Ensure that database is removed after uninstall
- ttrss: tests: functional: Uninstall during backup/restore test
- upgrades: Re-enable unattended-upgrade during dist-upgrade
- zoph: Re-add a safety check when reading the setup state of the app

10.53 FreedomBox 24.7 (2024-03-25)

10.53.1 Highlights

- package: Don't remove packages of other apps on uninstall
- samba: Fix Samba not accessible from IPv6 localhost ::1 address
- system: Organize items into sections
- users: Add email address field when creating/updating user accounts

10.53.2 Other Changes

- actions: Minor refactor to action error logging
- actions: Move most of the privileged action code to main directory
- actions: Provide HTML error message with action error
- backups: Adjust to changes in privileged errors
- letsencrypt: Remove unnecessary processing of the error messages
- letsencrypt: Show better error messages
- letsencrypt: Simplify error warning when certificate revoke fails
- matrixsynapse: Prevent setup page from being shown during uninstall
- middleware: Show HTML exception message as extra detail in messages
- package: Drop special error message handling for package errors
- samba: Disable nmbd NetBIOS service
- snapshot: Show better error messages
- storage: Adjust to changes in privileged errors
- storage: Show better error message
- tests: Automatically create pytest marks for apps
- tests: Merge actions related test files
- tests: Move test configuration to plinth directory
- tests: Remove unused fixture for testing actions
- upgrades: Show better error messages
- users: Add email address field during first boot
- views: Fix alignment of close button in error messages
- views: Implement a utility to easily show error message

10.54 FreedomBox 24.6 (2024-03-11)

10.54.1 Highlights

- gitweb: Fix modifying git repositories when gitweb app is disabled
- users: Fix creating users with initial set of groups
- wordpress:
  - Don't fail setup if mysql installed but not running
  - Drop database user when app is uninstalled
  - Fix backup, restore and uninstall when db is not running
- zoph:
  - Don't fail setup if mysql installed but not running
  - Don't fail with backup/restore if app is disabled

  - Don't redirect to setup page when app is disabled
  - Ensure that database server is running when setting up app
  - Hide configuration form when app is disabled
  - Restore database password to old value after restore operation
  - Uninstall fully so that reinstall works

10.54.2 Other Changes

- : Add type hints for app init methods
- : Add type hints for diagnose method
- action_utils: Implement method for starting a service temporarily
- actions: Fix log message when action return can't be decoded
- actions: When action errors out, log a better message
- app: Add ability to hide configuration form when app is disabled
- app: views: Expose method to get enabled/disabled state and cache it
- daemon: Added method to ensure a daemon is running in component
- diagnostics: Add tests for get_results
- diagnostics: Handle TypeError when copying results
- locale: Update translations for Dutch
- log: Don't log with in color inside actions scripts
- samba: Ignore non-existent users who are in freedombox-share group
- tests: functional: Refactor install/setup fixture for apps
- tests: functional: Uninstall app after backup and before restore
- users: Minor refactor when creating django groups
- users: tests: Do not remove LDAP user when testing views
- wordpress: Fix minor issue in restoring database
- wordpress: tests: Uninstall app after backup and before restore

10.55 FreedomBox 24.5 (2024-02-26)

- backups: tests: Don't use pytest marks on fixtures
- container: Fix issue with missing make command on stable image
- daemon: Add new component for daemons shared across apps
- firewalld: Implement force upgrading to any 2.x versions
- gitweb, users: Minor fixes for newer pycodestyle
- locale: Update translations for German
- pyproject.toml: Exclude the build directory from mypy checks
- setup: Ensure that apt is updated before checking force upgrade

- setup: Ensure that force upgrade won't run when app is not installed
- setup: Minor refactoring of force upgrader class instantiation
- setup: Try force upgrade before running app setup
- tests: Patch apps_init for enable/disable daemon test
- tor: tests: Convert to pytest style tests from class based tests
- tor: tests: Fix issue with pytest 8.x versions
- wordpress: Add shared daemon component for mariadb/mysql
- zoph: Add shared daemon component for mariadb/mysql

10.56 FreedomBox 24.4 (2024-02-12)

10.56.1 Highlights

- locale: Update translations for Albanian, Chinese (Simplified), German, Spanish, Swedish, Telugu, Turkish, Ukrainian

10.56.2 Other Changes

- : Introduce make file based build, eliminate setup.py
- Makefile: Move most of the provision process into build system
- Makefile: Move various tests into build system
- actions: Drop legacy placeholders for unused actions
- d/copyright: Update copyright year
- debian: Remove lintian override for init script
- doc: Install man1 page using Makefile
- doc: dev: Update all references to setup.py
- help: tests: Run tests using doc in current dir instead of /usr
- pyproject.toml: Move project meta data from setup.py

10.57 FreedomBox 24.3 (2024-01-29)

10.57.1 Highlights

- diagnostics: Add option to toggle daily run
- zoph: Fix failing PHP configuration requirements

10.57.2 Other Changes

- diagnostics: Add parameters to DiagnosticCheck
- diagnostics: Add method to translate checks
- diagnostics: Translate descriptions only in view
- diagnostics: Store results of full run in database
- diagnostics: Simplify getting translated description in results
- diagnostics: Safely access results when showing notification
- diagnostics: Fix a potential iteration of None value in error cases
- glib: Change API for repeating an in-thread scheduled task

10.58 FreedomBox 24.2 (2024-01-15)

- locale: Update translations for Chinese (Simplified), French, Spanish, Swedish, Turkish, Ukrainian

10.59 FreedomBox 24.1 (2024-01-01)

10.59.1 Highlights

- locale: Add translation for Belarusian
- storage: Show notification when rootfs is read-only

10.59.2 Other Changes

- locale: Update translation for Dutch
- tests: operation: Fix mock has_calls assertion

10.60 FreedomBox 23.21 (2023-11-20)

- locale: Update translations for Chinese (Simplified), Czech, Spanish, Swedish, Turkish, Ukrainian

10.61 FreedomBox 23.20 (2023-11-06)

10.61.1 Highlights

- backups: Don’t leave services stopped if backup fails
- coturn: Fix incorrectly passing transport argument to STUN URIs

10.61.2 Other Changes

- apache: tests: Update to use DiagnosticCheck class
- app: Update diagnose() docstring
- datetime: Fix diagnostic test for checking NTP server sync
- diagnostics: Add shortcut to re-run setup for app
- ejabberd: Update old STUN URIs to remove 'transport' parameter
- email: Increase the size of the message to 100MiB
- locale: Update translations for Albanian, German, Spanish, Turkish, Ukrainian
- matrixsynapse: Update old STUN URIs to remove 'transport' parameter
- operation: Fix issue with re-running setup when it fails first time
- tests: functional: Run tests on two app servers

10.62 FreedomBox 23.19 (2023-10-23)

10.62.1 Highlights

- backup: Fix bug in adding existing unencrypted backup location
- diagnostics: Run daily check and notify on failures
- email: Fix issue with install caused by missing drop-in config file
- kiwix: Add app for Kiwix offline Wikipedia reader
- upgrades: Allow matrix-synapse to be installed from bookworm

10.62.2 Other Changes

- db: Serialize most of the database queries using locks
- diagnostics: Add DiagnosticCheck dataclass
- diagnostics: Refactor background diagnostics task
- diagnostics: Refactor check IDs, tests and background checks
- glib: Add a jitter to the interval by default when scheduling tasks
- glib: Refactor schedule debugging in a central place
- kiwix: Do not require login to access the app
- kiwix: Drop unnecessary file in /etc/plinth/modules-enabled
- kiwix: Fix various issues after review
- locale: Update translations for Arabic, Czech, Dutch, German, Hindi, Spanish, Swedish, Telugu, Turkish, Ukrainian
- matrix-synapse: Update warning on how to change domain name
- operation: Add unique ID for each operation

10.63 FreedomBox 23.18 (2023-09-25)

- : Fix all typing hint related errors
- development: Make passing mypy checks mandatory
- development: Perform backports tests on bookworm instead of bullseye
- ikiwiki: Disable discussion pages by default for new wiki/blog
- locale: Update translations for Bulgarian
- middleware: Add new middleware to handle common errors like DB busy
- upgrades: Fix detecting apt over tor during upgrade
- wordpress: Use absolute path in service file

10.64 FreedomBox 23.17 (2023-09-11)

- locale: Update translations for Czech, Dutch, Spanish, Swedish, Turkish, Ukrainian

10.65 FreedomBox 23.16 (2023-08-28)

10.65.1 Highlights

- openvpn: Renew server/client certificates
- openvpn: Correctly set expiry of server/client certs to 10 years

10.65.2 Other Changes

- backups: Remove use of length_is template function
- django: Remove use of X-XSS-Protection header
- locale: Update translations for Czech, Norwegian Bokmål, Swedish
- networks, samba: Fix tests setting firewall zone
- openvpn: Ensure that re-running setup works as expected
- openvpn: Fix app not installing on Debian testing
- openvpn: Minor refactoring in setting up easy-rsa
- openvpn: Use config file instead of env vars for easy-rsa
- sso: Switch to django-axes >= 5.0
- sso: Use POST method for logout
- users, networks: Use the autofocus HTML attribute sparingly

10.66 FreedomBox 23.15 (2023-08-14)

- debian: Add Swedish translation for debconf
- locale: Update translations for Dutch, German, Spanish, Turkish, Ukrainian

10.67 FreedomBox 23.14 (2023-07-31)

10.67.1 Highlights

- app: Implement advanced option to rerun app setup
- torproxy: Add separate app for Tor Proxy
- upgrades: Use codename in apt preferences

10.67.2 Other Changes

- HACKING: Add instructions for container on Raspberry Pi
- bepasty: Don't enable app when setup is rerun
- bind: Don't enable app when setup is rerun
- ci: Add mypy static type check
- container: Add support for retrieving GPG keys using wget
- container: Update for bookworm images
- deluge: Don't enable app when setup is rerun
- ejabberd: Don't enable app when setup is rerun
- firewall: Add diagnostic check for backend
- firewall: Add diagnostic check for passthroughs
- firewall: Add diagnostic for default zone
- gitweb: Don't enable app when setup is rerun
- ikiwiki: Don't enable app when setup is rerun
- infinoted: Don't enable app when setup is rerun
- janus: Don't enable app when setup is rerun
- jsxc: Don't enable app when setup is rerun
- kvstore: Optionally, don't throw exception when deleting key
- locale: Update translations for Dutch
- mediawiki: Don't enable app when setup is rerun
- minetest: Don't enable app when setup is rerun
- openvpn: Don't enable app when setup is rerun
- performance: Don't enable app when setup is rerun
- privoxy: Don't enable app when setup is rerun
- quassel: Don't enable app when setup is rerun
- radicale: Don't enable app when setup is rerun
- rssbridge: Don't enable app when setup is rerun
- shaarli: Don't enable app when setup is rerun

- sharing: Don't enable app when setup is rerun
- tor, torproxy: Export settings from old to new app
- tor, torproxy: Update description for info on services provided
- tor: tests: Make functional test check for running service
- ttrss: Don't enable app when setup is rerun
- upgrades: Use codename for unattended-upgrades origin pattern
- users: Add diagnostic checks for nsswitch config
- users: Add diagnostics check for nslcd config
- wireguard: Don't enable app when setup is rerun
- zoph: Don't enable app when setup is rerun

10.68 FreedomBox 23.13 (2023-07-17)

- container: Add support for ARM64 containers
- HACKING: Instructions for macOS on Apple Silicon
- locale: Update translations for German, Spanish, Turkish, Ukrainian

10.69 FreedomBox 23.12 (2023-06-19)

10.69.1 Highlights

- packages: Purge packages on uninstall
- gitweb: Fix issue with service startup when gitweb is not enabled
- mediawiki: Increment version to run update.php automatically

10.69.2 Other Changes

- deluge: Utilize purging of packages and don't remove explicitly
- locale: Update translations for Czech, Dutch, Spanish, Turkish, Ukrainian
- mediawiki: Utilize purging of packages and don't remove explicitly
- roundcube: Clarify description for local mail only option
- samba: Remove additional configuration files on uninstall
- searx: Fix typo in method name
- shaarli: Utilize purging of packages and don't remove explicitly
- uninstall: Remove experimental warning

10.70 FreedomBox 23.11 (2023-06-05)

10.70.1 Highlights

- : Fix icons not present in the generated .deb
- shadowsocksserver: Add separate app for Shadowsocks server

10.70.2 Other Changes

- apache: Fix failure during app update
- apache: Use drop-in config component for /etc files
- bepasty: Use drop-in config component for /etc files
- calibre: Use drop-in config component for /etc files
- cockpit: Use drop-in config component for /etc files
- config: Add new component for managing drop-in /etc/ config files
- debian/install: Add new place in /usr to keep drop-in config files
- deluge: Use drop-in config comonents for /etc files
- ejabberd: Use drop-in config component for /etc files
- email: Use drop-in config component for /etc files
- gitweb: Use drop-in config component for /etc files
- i2p: Use drop-in config component for /etc files
- ikiwiki: Use drop-in config component for /etc files
- janus: Use drop-in config component for /etc files
- letsencrypt: Use drop-in config component for /etc files
- matrixsynapse: Use drop-in config component for /etc files
- mediawiki: Use drop-in config component for /etc files
- minidlna: Use drop-in config component for /etc files
- networks: Use drop-in config component for /etc files
- pagekite: Drop the config file for forcing use of Debian certs
- privacy: Use drop-in config component for /etc files
- radicale: Use drop-in config component for /etc files
- roundcube: Use drop-in config component for /etc files
- rssbridge: Use drop-in config component for /etc files
- searx: Use drop-in config component for /etc files
- security: Use drop-in config component for /etc files
- sharing: Use drop-in config component for /etc files
- ssh: Use drop-in config component for /etc files
- sso: Use drop-in config component for /etc files
- syncthing: Use drop-in config component for /etc files
- transmission: Use drop-in config component for /etc files
- ttrss: Use drop-in config component for /etc files
- upgrades: Use drop-in config component for /etc files
- users: Use drop-in config component for /etc files
- wordpress: Use drop-in config component for /etc files

10.71 FreedomBox 23.10 (2023-05-22)

- : Move modules-enabled files to /usr/share
- locale: Update translations for Dutch, Portuguese, Spanish, Turkish, Ukrainian
- doc/dev: Set language code explicitly in Sphinx configuration
- gitweb: Disable gpg signing in tests

10.72 FreedomBox 23.9 (2023-05-08)

10.72.1 Highlights

- transmission: Allow remote UIs to connect
- ttrss: Allow apps to use /tt-rss URL instead of separate one

10.72.2 Other Changes

- apache: Reload apache using component if config changes
- debian: Follows policy v4.6.2
- debian: Update copyright years
- doc/dev: Update copyright year
- help: Add information on obtaining source code
- locale: Update translations for Japanese
- mediawiki: Make a utility method public
- mediawiki: Make retrieving list of supported languages robust
- mediawiki: Simplify retrieving the default language
- minidlna: Resize icon and export to PNG also
- service: Remove reference to managed_services in a message
- storage: Handle mount error properly
- theme: Move icons to app folders
- tor: Check if Hidden service is version 3
- tor: Only diagnose relay ports if feature enabled
- tor: Rename Hidden service to Onion service
- transmission: Add Tremotesf to list of client apps
- ttrss: Don’t show app in enabled list of apps if install fails
- ttrss: Update list of clients
- ttrss: Use the apache component to restart apache on config change
- uninstall: Fix issue with uninstall of apps that have no backup
- zoph: Don’t fail at showing app view during uninstall

10.73 FreedomBox 23.6.2 (2023-05-01)

- upgrades: Check apt result during dist-upgrade
- locale: Update translations for Bulgarian, Ukrainian

10.74 FreedomBox 23.8 (2023-04-24)

10.74.1 Highlights

- coturn: Prevent package removal when roundcube is uninstalled
- datetime: Re-implement backup/restore for timezone
- gitweb: Disable snapshot feature

10.74.2 Other Changes

- HACKING: Force pip to install packages to system environment
- bepasty: When uninstalling, remove all data and configuration
- calibre: Remove libraries during uninstallation
- ci: Force pip install for functional tests
- coturn: When uninstalling, remove all data and configuration
- datetime: Use unique component ID for related daemon
- deluge: When uninstalling, remove all data and configuration
- gitweb: When uninstalling, remove all data and configuration, remove repositories
- gitweb: Make globally configured features overridable per-repository
- gitweb: Simplify handling shortcut for front page
- ikiwiki: When uninstalling, remove all data and configuration
- infinoted: When uninstalling, remove all data and configuration
- locale: Update translations for Bulgarian, French, Ukrainian
- matrixsynapse: When uninstalling, remove all data and configuration
- mediawiki: When uninstalling, remove all data and configuration
- mediawiki: Fix broken view on Bullseye due to language selection
- openvpn: When uninstalling, remove all data and configuration
- roundcube: When uninstalling, remove all data and configuration
- rssbridge: When uninstalling, remove all data and configuration
- samba: When uninstalling, remove all data and configuration
- searx: When uninstalling, remove all data and configuration
- searx: Simplify handling shortcut for front page
- shaarli: When uninstalling, remove all data and configuration

- shadowsocks: When uninstalling, remove all data and configuration
- sharing: When uninstalling, remove all data and configuration
- syncthing: When uninstalling, remove all data and configuration
- syncthing: Remove unused pathlib import so job code-quality can pass
- tests: Don't error during collection if selenium is not installed
- tests: functional: Make install script work for Bullseye
- tests: functional: Remove handling for custom enable/disable buttons
- tests: functional: Update detecting page changes
- tor: When uninstalling, remove all data and configuration
- ttrss: When uninstalling, remove all data and configuration
- upgrades: Check apt result during dist-upgrade
- wordpress: When uninstalling, remove all data and configuration

10.75 FreedomBox 23.6.1 (2023-04-10)

- coturn: Prevent package removal when roundcube is uninstalled
- datetime: Re-implement backup/restore for timezone
- gitweb: Disable snapshot feature
- gitweb: Make globally configured features overridable per-repository
- locale: Update translations for Bulgarian, French

10.76 FreedomBox 23.7 (2023-03-27)

10.76.1 Highlights

- matrixsynapse: Add token based registration verification
- mediawiki: Allow setting site language code

10.76.2 Other Changes

- container: Fix resizing disk image containing multiple partitions
- container: Force pip to install packages to system environment
- container: Increase wait time to accommodate slower architectures
- locale: Update translations for Bulgarian
- tests: functional: Fix setting first ethernet connection as internal

10.77 FreedomBox 23.6 (2023-03-13)

- /etc/issue: Update message to reflect that all users can login
- ci: Force pip to install packages to system environment
- datetime: Use timedatectl to read current timezone
- samba: Make sure shares are not accessible from the internet
- ttrss: Fix failing backup

10.78 FreedomBox 23.5 (2023-02-27)

- mediawiki: Fix app view error
- locale: Update translations for Albanian, Bulgarian, Czech, Dutch, German, Spanish, Swedish, Turkish, Ukrainian
- samba: tests: Fix enable share view test

10.79 FreedomBox 23.4 (2023-02-13)

10.79.1 Highlights

- ejabberd: Fix making call connections when using TURN
- matrixsynapse: Disable verification to fix public registrations
- snapshot: Fix issue with snapshot rollbacks

10.79.2 Other Changes

- app: Add backup and restore menu items to toolbar menu
- backups: Allow selecting a single app from URL when creating backup
- config: Drop RuntimeMaxUse=5% for journal logging
- dynamicdns: Skip uninstall test
- ejabberd: Add Monal and Siskin for iOS and remove ChatSecure
- email: Redirect to the app page if roundcube isn't installed
- ikiwiki: Re-run setup for each site after restore
- locale: Update translations for Bulgarian, Spanish
- matrixsynapse: Minor refactor in getting/setting public registrations
- matrixsynapse: Add python3-psycopg2 to packages
- matrixsynapse: Use yaml.safe_load
- searx: Add libjs-bootstrap to packages
- snapshot: Fix mounting /.snapshots subvolume and use automounting
- templates: Show better title for 404 page
- uninstall: Fix spelling in warning message
- vagrant: Drop unnecessary script that deletes sqlite file

- vagrant: Hide the vagrant-script directory
- vagrant: Mount source in /freedombox instead of /vagrant
- vagrant: Switch to /freedombox before running service with alias

10.80 FreedomBox 23.3 (2023-01-30)

- config: Fix showing the value of the default home page
- email: Revert workaround for error on finishing uninstall
- firewalld: Allow upgrade to version 2*
- gitweb: tests: Skip tests using git when git is not installed
- locale: Update translations for Bulgarian
- tests: functional: Fix submitting forms with notifications present
- tor: Also use Aptsources822 augeas lens
- tor: Remove workaround for old Augeas bug
- upgrades: Add augeas lens for Deb822 apt sources
- views: Use dedicated view when showing an app with operations

10.81 FreedomBox 23.2 (2023-01-16)

- locale: Update translations for Albanian, Bulgarian
- ssh: Add sudo to allowed groups
- upgrades: Stop quassel during dist upgrade

10.82 FreedomBox 23.1 (2023-01-03)

10.82.1 Highlights

- package: Don’t uninstall packages that are in use by other apps
- tor: Add onion location to apache

10.82.2 Other Changes

- email: Workaround an issue with error on finishing uninstall
- gitweb: Run git commands as a web user
- janus: Allow upgrade to 1.1
- locale: Update translations for Galician, Spanish
- operation: tests: Fix warning when test helpers start with ’Test’
- zoph: Add explicit dependency on default-mysql-server

10.83 FreedomBox 22.27 (2022-12-19)

10.83.1 Highlights

- minidlna: Fix incorrect marking for firewall local protection
- zoph, wordpress: Add conflicts on libpam-tmpdir

10.83.2 Other Changes

- container: Drop free tag from image URLs
- d/control: Don't recommend libpam-tmpdir
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, German, Turkish, Ukrainian
- package, email: Move conflicting package removal to framework
- snapshot: Fix showing unsupported message on non-btrfs filesystems
- tests: functional: Set timeout to 3 hours
- upgrades: dist-upgrade: Don't change apt security line
- users: tests: Fix privileged tests
- wordpress: Redirect Webfinger queries

10.84 FreedomBox 22.26 (2022-12-05)

10.84.1 Highlights

- ejabberd: Enable mod_http_upload
- security: Remove restricted access setting and configuration
- ssh: Restrict logins to groups root, admin and freedombox-ssh

10.84.2 Other Changes

- calibre: Add protection to local service using firewall
- deluge: Add protection to local service using firewall
- email: Add protection to local service using firewall
- firewall: Create a mechanism for protecting local services
- firewall: Introduce component for local service protection
- i2p: Add protection to local service using firewall
- i2p: Remove donation URL that is no longer available
- minidlna: Add protection to local service using firewall
- searx: Ensure that socket is only reachable by Apache and root
- ssh: Add checkbox to remove login group restrictions
- syncthing: Add protection to local service using firewall
- transmission: Add protection to local service using firewall

10.85 FreedomBox 22.25 (2022-11-21)

- email: Add fail2ban jail for dovecot
- email: Fix creation of aliases for security@ and usenet@

10.86 FreedomBox 22.24 (2022-11-07)

10.86.1 Highlights

- locale: Update translations for Bulgarian, French, German, Norwegian Bokmål

10.86.2 Other Changes

- debian/lintian-overrides: Fix mismatch patterns and new messages
- minetest: Handle upgrade from 5.3.0 to 5.6.1
- storage: Drop skip_recommends
- upgrades: Add documentation link to upgrades service file
- upgrades: Update list of holds during dist upgrade

10.87 FreedomBox 22.23 (2022-10-24)

10.87.1 Highlights

- letsencrypt: Fix regression with comparing certificate
- rssbridge: Add option to allow public access

10.87.2 Other Changes

- locale: Update translations for Bulgarian, Hungarian, Swedish
- storage: Handle file systems on non-physical devices
- upgrades: Allow FreedomBox vendor when adding backports
- upgrades: Skip unattended-upgrade in dist-upgrade

10.88 FreedomBox 22.22.1 (2022-10-16)

- debian: tests: Fix PYTHONPATH
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, Turkish, Ukrainian
- privacy: Remove unused import, fix pipeline

10.89 FreedomBox 22.22 (2022-10-10)

10.89.1 Highlights

- privacy: Add new system app for popularity-contest
- matrix: Add fail2ban jail

10.89.2 Other Changes

- : Use privileged decorator for actions
- action_utils: Drop support for non-systemd environments
- action_utils: Drop unused progress requests from apt-get
- actions: Allow actions to be called by other users
- actions: Allow nested and top-level actions
- actions: Drop unused superuser_run and related methods
- actions: Implement getting raw output from the process
- actions: Use separate IPC for communicating results
- apache: Fix logs still going into /var/log files
- bind: Drop enabling DNSSEC (deprecated) as it is always enabled
- config: Drop ability to set hostname on systems without systemd
- config: Drop legacy migration of Apache homepage settings
- fail2ban: Make fail2ban log to journald
- firewall: Drop showing running status
- locale: Update translations for Albanian, Czech, Norwegian Bokmål, Russian, Swedish, Ukrainian
- minidlna: Use the exposed URL for diagnostic test
- openvpn: Drop RSA to ECC migration code and two-step setup
- privacy: Set vendor as FreedomBox for dpkg and popularity-contest
- searx: Show status of public access irrespective of enabled state
- templates: Update HTML meta tags for better description and app-name
- tests: Add fixture to help in testing privileged actions
- wordpress: Update fail2ban filter

10.90 FreedomBox 22.21.1 (2022-10-01)

- locale: Update translations for Bulgarian, Ukrainian
- notification: Don’t fail when formatting message strings

10.91 FreedomBox 22.21 (2022-09-26)

- janus: Enable systemd sandboxing
- locale: Update translations for Albanian, Bulgarian, Czech, Danish, Dutch, French, German, Greek, Hungarian, Indonesian, Italian, Latvian, Lithuanian, Norwegian Bokmål, Persian, Polish, Portuguese, Russian, Serbian, Slovenian, Spanish, Swedish, Turkish, Ukrainian
- mediawiki: Add powered by freedombox logo
- wordpress: Add fail2ban filter and jail
- wordpress: Disable readme.html, xmlrpc.php, wp-cron.php

10.92 FreedomBox 22.20 (2022-09-12)

10.92.1 Highlights

- matrixsynapse: Allow matrix-synapse >= 1.65 to install successfully

10.92.2 Other Changes

- backups: Use generic form template for create and schedule views
- backups: tests: functional: Find forms more accurately
- bepasty: Use generic form template for add password view
- bepasty: tests: functional: Minor refactor for form submission
- calibre: tests: functional: Find forms more specifically
- d/maintscript: remove tahoe and mldonkey apache conf files
- debian: Add Italian debconf translation
- ejabberd: tests: functional: Ensure jsxc is installed
- firewall: Allow upgrade from any version to 1.2.*
- first_boot: tests: functional: Find form more specifically
- gitweb: Fix issue with page not refreshing during uninstall
- gitweb: Use generic form template for create/edit repository
- gitweb: tests: functional: Find forms more accurately
- ikiwiki: tests: functional: Find forms more accurately
- locale: Update translations for Chinese (Simplified), Czech, French, Italian, Turkish
- samba: Ignore mounted files when listing mounts
- samba: Update client apps information
- shaarli: tests: functional: Specify setup form submission button
- sharing: tests: functional: Find forms more accurately
- snapshot: tests: functional: Minor refactoring for form submission
- sso: tests: functional: Find forms more accurately
- templates: form: Specify a form class for use with functional tests
- tests: functional: Assert app is not installed after uninstallation
- tests: functional: Force specifying form to submit more accurately
- tests: functional: Wait for installation to complete fully
- users: tests: functional: Find forms more accurately
- version: Compare Debian package version numbers
- wordpress: tests: functional: Find forms more specifically
- zoph: tests: functional: Simplify finding the form to submit

10.93 FreedomBox 22.19 (2022-08-29)

10.93.1 Highlights

- jsxc: Allow disabling the app
- app: Add a menu item to trigger uninstallation

10.93.2 Other Changes

- app: Add API to uninstall an app
- avahi: Don't disable after tests
- backups: Use AppView for the main app page
- container: Display help message when no args are passed
- container: Show default values in command help
- d/control: Break ufw as we use firewalld
- debian: Update Spanish translation template
- diagnostics: Use AppView for app page
- ejabberd: Set hostname for test that relies on it
- forms: Implement form for uninstallation
- janus: Convert action to privileged
- janus: Handle upgrades to 1.0.*
- letsencrypt: Use AppView for app page
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, French, German, Spanish, Turkish, Ukrainian
- names: Use AppView for app page
- networks: Use AppView for app page
- operation: Factor out template code into a separate file
- operation: Show operations on app page in addition to setup page
- package: Implement low-level methods for uninstalling
- package: Implement uninstall in Package component
- power: Use AppView for app page
- security: Use AppView for app page
- setup: Drop check for already running operation
- setup: Implement operation to uninstall an app
- snapshot: Use AppView for app page
- tests: Make functional.is_available check faster
- tests: functional: Add install/uninstall test for all apps
- tor: Use AppView and Operation for app page

- ttrss: Add donation url
- upgrades: Add button to test dist-upgrade in development mode
- upgrades: Hold janus during dist-upgrade
- views: Implement a view to uninstall an app

10.94 FreedomBox 22.18 (2022-08-15)

10.94.1 Highlights

- networks: Remove DNSSEC diagnostics
- setup: Allow starting installation when package manager is busy
- setup: Fix issue with immediate refresh after installation

10.94.2 Other Changes

- : Add setup method on all apps that don't have it
- : Drop module level app property
- : Make force upgrading part of app rather than a module
- : Make setup method part of App class for all apps
- app: Drop optimization that skips setup process
- backups: tests: Mark need for Django database during API tests
- container: Add IdentitiesOnly option to SSH
- container: Ignore flake8 error 'line too long' in bash script text
- coturn: Fix link to ejabberd in description
- doc: dev: Document previously undocumented components
- ejabberd: Fix showing the status messages
- locale: Update translations for Bulgarian, Czech, Dutch, French, German, Turkish, Ukrainian
- matrixsynapse: Fix showing the status messages
- notification: Pass full context when rendering body template
- operation: Add module to manage threaded operations
- package: Run installation operation using app_id instead of module
- setup: Drop setup_helper and use the new Operation API
- sharing: Add installing and enable/disable like other apps
- sharing: tests: functional: Fix a flaky test by waiting
- ssh: tests: functional: Keep service enabled after tests
- storage: Fix enumerating partitions without mount points

10.95 FreedomBox 22.17 (2022-08-01)

10.95.1 Highlights

- help: Add "How can I help?" section to Contribute page

10.95.2 Other Changes

- locale: Update translations for Chinese (Simplified), Dutch, French, German, Turkish
- wordpress: Don't install php-ssh2

10.96 FreedomBox 22.16 (2022-07-18)

10.96.1 Highlights

- cockpit: Reconfigure to allow any origin
- rssbridge: New app to generate RSS feeds for websites

10.96.2 Other Changes

- apache: Also configure to serve on /freedombox
- apache: Merge old configuration files into a better location
- apache: Redirect all logs to systemd journal
- cockpit: Depend on apache and setup after it
- cockpit: Use decorator for privileged actions
- config: Add option to set logging mode: none/volatile/persistent
- config: Set volatile logging by default
- debian: Follows policy version 4.6.1
- debian: Update copyright year
- gitweb: Switch default branch name to main for new repositories
- janus: Change short description to "Video Room"
- locale: Update translations for Bulgarian, Chinese (Simplified), French, Russian, Ukrainian
- privoxy: Restrict to private IPs, prevent access over the internet
- privoxy: Use privileged decorator for actions
- roundcube: Add fail2ban jail
- roundcube: Configure to log to journald
- roundcube: Use privileged to simplify actions
- rssbridge: Add functional tests
- rssbridge: Fix flake8 errors
- rssbridge: Whitelist all bridges by default

10.97 FreedomBox 22.15 (2022-07-04)

10.97.1 Highlights

- backups: Add options to keep sshfs shares responsive
- backups: Unmount repositories before and after backup
- users: create home directories for newly created users

10.97.2 Other Changes

- : pylint: Avoid calling super() with arguments
- : pylint: Don't inherit from 'object'
- : pylint: Drop unnecessary 'pass' statements
- : pylint: Explicitly specify encoding when open a file
- : pylint: Suppress unused argument warnings
- ci: Use compatible versions of Selenium and Splinter
- locale: Update translations for Bulgarian, Russian, Ukrainian
- mediawiki: Add regex validator to the domain field
- mediawiki: Remove Buster specific code not needed in Bullseye
- mediawiki: Remove wgLogo as it is not needed in Bullseye
- pyproject.toml: Ignore some refactoring messages with pylint
- static: js: css: Make multiple select fields work with Django 4.0
- tests: functional: Simplify GitLabCI configuration
- upgrades: Hold packages one at a time
- upgrades: Re-add workaround for grub
- views: Add a comment about change in Django 4.0

10.98 FreedomBox 22.14.1 (2022-06-27)

10.98.1 Highlights

- matrixsynapse: Allow new dependency to be installed from backports

10.98.2 Other Changes

- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, German, Turkish
- actions: Note that privileged actions can't output to stdout
- mumble: Backup/restore the configuration file
- mumble: Don't set the root channel name unless it is changed
- mumble: Use privileged decorator for superuser actions
- mumble: tests: Add functional tests for setting the passwords

10.99 FreedomBox 22.14 (2022-06-20)

10.99.1 Highlights

- ejabberd: Allow domains to be added or removed
- mumble: Allow setting a password that is required to join the server
- mediawiki: Add option to change the site name

10.99.2 Other Changes

- actions: Add a decorator for marking superuser actions
- doc: dev: Use and recommend new privileged actions
- ejabberd: Automatically use coturn
- janus: Improve description about coturn
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, French, German, Russian, Turkish
- tests: Add a dummy parameter for middlewares for Django 4.0
- translation: Don't use session for storing lang pref in Django 4.0
- transmission: Simplify actions using the privileged decorator
- users: Fix deleting user LDAP entry with Django 4.0

10.100 FreedomBox 22.13 (2022-06-06)

10.100.1 Highlights

- email: Make app available for all users (even without advanced flag)
- janus: Add new app for lightweight WebRTC server
- wordpress: Allow installing/updating plugins and themes

10.100.2 Other Changes

- email: Add description about ISP and domain limitations
- locale: Added Latvian translation
- locale: Update translation for Bulgarian, Chinese (Simplified), Czech, Dutch, German, Russian, Swedish, Turkish
- mumble: Allow changing root channel name
- tests: functional: Add jobs for bullseye-backports
- tests: functional: Integrate into Salsa CI
- transmission: Add redirects to avoid 409 conflict
- wordpress: tests: Continue past language selection screen
- wordpress: tests: Fix writing title for new post in newer versions

10.101 FreedomBox 22.12 (2022-05-23)

10.101.1 Highlights

- : Show Learn More... links in frontpage with description
- mediawiki: Serve hidden service over http for .onion domains

10.101.2 Other Changes

- apache: Allow URL diagnostics to work with redirects
- firewall: Show service name in port forwarding info table
- frontpage: Allow showing links to manual pages
- frontpage: Reuse app header template for showing app description
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, French, German, Norwegian Bokmål, Turkish
- mediawiki: Add stricter sandbox rules for jobrunner service
- mediawiki: Fix URL diagnostics with redirects involved
- ssh, bind: Show 'Learn More...' links
- tor: Show port forwarding information in consistent way
- tt-rss: Fix description about user access

10.102 FreedomBox 22.11 (2022-05-09)

10.102.1 Highlights

- email: Fix userdb lookups with LDAP
- matrixsynapse: Allow new dependencies to be installed from backports

10.102.2 Other Changes

- HACKING: Improve documentation on how to run tests
- container: Show executed commands when setting up/running tests
- locale: Update translations for Bulgarian, Danish, French, Hungarian, Norwegian Bokmål, Polish, Russian, Ukrainian
- mediawiki: Check if admin password is at least 10 characters long
- mediawiki: Handle password rejection from MediaWiki
- samba: Fix functional tests when user is not logged in at start
- tests: functional: Get rid of dependency on xvfb
- transmission: Improve description

10.103 FreedomBox 22.10 (2022-04-25)

- locale: Update translations for Bulgarian, Czech, Dutch, German, Greek, Russian, Swedish, Turkish
- sharing: Allow spaces in path strings

10.104 FreedomBox 22.9 (2022-04-11)

10.104.1 Highlights

- minetest: Allow alternate name for 3d armor mod
- upgrades: Use python3-typing-extensions from bullseye-backports

10.104.2 Other Changes

- calibre: Fix description of allowable library names
- locale: Add new Arabic translation
- locale: Update translations for Arabic, Bulgarian, Czech, Dutch, French, German, Hungarian, Turkish
- plinth: Add forum link to footer

10.105 FreedomBox 22.8 (2022-03-28)

10.105.1 Highlights

- network: Fix showing wifi connection

10.105.2 Other Changes

- calibre: explain correct name format for new library
- ikiwiki: add packages that are necessary
- locale: Updated translations for Chinese (Simplified), French, Russian
- upgrades: Allow backports from src:freedombox

10.106 FreedomBox 22.7 (2022-03-14)

- locale: Update translations for French, Hungarian, Spanish

10.107 FreedomBox 22.6.1 (2022-03-06)

- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, German, Turkish

10.108 FreedomBox 22.6 (2022-03-02)

10.108.1 Highlights

- email: Enable as an advanced app

10.108.2 Other Changes

- dynamicdns: Fix adding null domain into configuration
- email: Add backup/restore component
- email: Add basic functional tests
- email: Add front page shortcut, update name and description
- email: Add more special-use IMAP folders, set autoexpunge to 60days
- email: Add shortcut for non-admin users to manage their aliases
- email: Add various documentation links for future readability
- email: Allow re-running setup
- email: Backup/restore aliases and mailboxes
- email: Depend on and run redis server
- email: Implement adding common aliases for first admin user
- email: List all listening ports of the daemons
- email: Narrowly match just rspamd's spam header
- email: Open firewall port for managesieve protocol
- email: Revert to LDAP auth as pam does not allow non-admin users
- email: Set an icon from Tango project
- email: Setup rspamd configuration to include FreedomBox config
- email: Tweak client auto-configuration file
- email: Update donation URL to rspamd donation URL
- email: aliases: Drop ability to enable/disable aliases
- email: clients: Make Thunderbird URLs language independent
- email: dkim: Implement setting up DKIM signing keys
- email: dns: Show table for desired DNS entries
- email: postfix: Fix priority for authentication directives
- email: postfix: use inline map for TLS SNI maps
- email: rspamd: Log to journald via syslog
- email: rspamd: Simplify installing configuration
- locale: Update translations for Bulgarian, Chinese (Simplified), Czech, Dutch, German, Hungarian, Russian, Swedish, Telugu, Turkish, Ukrainian
- minidlna: add iOS VLC client
- samba: add iOS VLC client

10.109 FreedomBox 22.5 (2022-02-14)

10.109.1 Highlights

- dynamicdns: Replace ez-ipupdate
  - Drop NAT detection as it is no longer used
  - Drop about page and merge into description
  - Drop tabs and use single page
  - Rewrite configuration handling and update using URL

10.109.2 Other Changes

- app: Add component to store enabled state of an app in kvstore
- backups: Implement backup/restore of key/value settings
- locale: Update translations for Albanian, Bulgarian, Chinese (Simplified), Czech, Dutch, German, Hungarian, Russian, Spanish, Swedish, Turkish, Ukrainian
- minetest: Reduce the number of configuration update messages
- tests: functional: Add plugin for HTML reports
- tt-rss: Restrict access to `feed-reader` group in "/tt-rss-app"
- users: Fix typo in description

10.110 FreedomBox 22.4 (2022-01-31)

10.110.1 Highlights

- coturn: Use wildcard listening address to fix startup issues
- sso, users: Redirect to home page after logout

10.110.2 Other Changes

- apache: Don’t redirect to HTTPS for .onion domains
- apache: Don’t set HSTS for .onion domain
- cockpit: Explicitly redirect to HTTPS as needed for WebSockets
- doc: Fail when downloading images from Debian wiki fails
- email_server: Drop showing diagnostics/repair and roundcube config
- email_server: Drop some unused code
- locale: Update translations for Bulgarian, Dutch, French, German, Hungarian, Russian, Swedish, Ukrainian
- matrixsynapse: Add FluffyChat to client list
- mldonkey: Drop app not available in Debian Bullseye and Bookworm
- power: Add a link to power app in the system menu
- roundcube: Add setting for local connection only

- shaarli: Add android app to description
- shaarli: Add backup component
- shaarli: Add functional tests
- snapshots: Clarify that snapshots are take during updates too
- tests: functional: Implement a workaround for issue with screenshots
- users: Clarify help message for authorization password
- wireguard: tests: Add functional tests

10.111 FreedomBox 22.3 (2022-01-17)

10.111.1 Highlights

- upgrades: Allow matrix's new dependency to be installed
- sso: Adjust URL to CAPTCHA page needed by Django security fix

10.111.2 Other Changes

- container: Avoid a warning that interactive mode is intended
- help: tests: Fix functional test to check for status logs
- locale: Update translations for Chinese (Simplified), Czech, Dutch, German, Hungarian, Turkish
- sso: Add missing captcha/rate limiting on SSO login
- tests: functional: Fix setting domain name with active notifications
- tt-rss: Allow published articles to be publicly available

10.112 FreedomBox 22.2 (2022-01-11)

10.112.1 Highlights

- debian, setup.py: Add dependency on python3-tomli
- help: Fix failing setup when manual directory is not available

10.112.2 Other Changes

- backups: Correct spelling of encryption protocols
- i2p: Fix grammar in description
- ikiwiki: Initialize shortcuts during post-init setup
- locale: Update translations for Chinese (Simplified), Czech, Dutch, German, Hungarian, Italian, Swedish, Turkish
- mumble: Change description to include iOS client app
- networks: Fix reference to an option
- openvpn: Add link to IOS app
- radicale: Update Thunderbird URLs
- transmission: Fix capitalization
- wireguard: Fix spelling

10.113 FreedomBox 22.1 (2022-01-03)

10.113.1 Highlights

- package: Add diagnostic to check if a package is the latest version

10.113.2 Other Changes

- backups: Capitalize 'SSH' in template
- config, upgrades: Specify submit button for tests
- datetime: Explicitly list systemd-timesyncd as a dependency
- locale: Update translations for Chinese (Traditional), Czech, Dutch, Hungarian, Norwegian Bokmål, Spanish, Swedish
- storage: Skip tests if not enough disk space is available
- upgrades: Relabel from 'Update' to 'Software Update'

10.114 FreedomBox 21.16 (2021-12-20)

10.114.1 Highlights

- datetime: Fix checking when timesyncd will run on a system

10.114.2 Other Changes

- cockpit, ejabberd: Make 'name' optional in Signal handlers
- diaspora: Drop app that was never finished
- email_server:
  - Adjust TLS configuration parameters
  - Fix issue with handling domain removal
  - Include postfix package in packages list
  - Re-implement TLS configuration
  - Rename dovecot TLS configuration file for consistency
- letsencrypt: Handle cert setup when an app wants all domains
- locale: Update translations for Chinese (Simplified), Chinese (Traditional), Czech, French, German, Swedish, Turkish
- monkeysphere, tahoe-lafs: Drop unused apps
- roundcube: Allow upgrades by avoiding configuration file change
- tests: Fix app name in pytest.skip statement
- tests: functional: Skip MLDonkey app
- upgrades: Cleanup dist upgrade steps specific to bullseye release
- upgrades: Refactor dist upgrade process

10.115 FreedomBox 21.15 (2021-12-06)

10.115.1 Highlights

- dynamicdns: Update URLs to the new dynamic DNS server
- firewall: Allow configuration upgrade to version 1.0.x
- shaarli: Enable app (only available in testing and unstable)

10.115.2 Other Changes

- : Drop module level depends declaration
- : Drop module level package_conflicts and use component API
- : Drop unused manual_page at module level
- : Drop use of managed_packages and rely on Packages component
- : Drop use of managed_services, rely on Daemon component
- : Drop use of module level is_essential flag
- : Drop use of module level version
- : Drop use of unnecessary managed_paths
- : Use the App's state management API
- actions/letsencrypt: Drop use of managed_paths and use LE component
- actions/service: Drop unused list action
- actions/service: Drop use of managed_services for Daemon component
- actions: Get list of packages from Packages components
- app: Introduce API for managing setup state of the app
- app: Introduce API to setup an app
- bind: Drop alias handling unnecessary in >= Bullseye
- daemon: Add new component to hold information about related daemons
- doc/dev: Drop discussion on managed_paths
- doc/dev: Drop reference to module level depends declaration
- doc/dev: Remove mention of managed_services
- doc/dev: Remove outdated reference to init() at module level
- doc/dev: Update documentation to not refer to managed_packages
- email_server: Merge domain configuration with app view
- email_server: Simplify domain configuration form
- first_boot: Drop use of loaded_modules and use App.list
- forms: Fix regression with TLS domain form in quassel and tt-rss
- letsencrypt: On domain removal, don't revoke certificate, keep it

- locale: Update translations for Czech, German, Norwegian Bokmål
- main: List apps instead of modules
- middleware, views: Reduce use of setup_helper
- module_loader, app: Move app init to app module
- package: Add parameter to specify skipping package recommendations
- package: Implement installing packages in the component
- package: Introduce component API for package conflicts
- packages: Move checking for unavailable packages to component
- security: Drop use of loaded_modules and use App.list
- security: Drop use of managed_services in security report
- security: Get the list of packages from Packages component
- setup: Drop unused API for app's state management
- setup: List dependencies for apps instead of modules
- setup: Run setup on apps instead of modules
- setup: Use apps instead of modules to determine running first setup
- setup: Work on apps instead of modules for force upgrade
- tests: Add 'domain' mark for apps that add/remove domains
- web_server: Drop use of loaded_modules and use App.list

10.116 FreedomBox 21.14.1 (2021-11-24)

- config: Add packages component to a re-add zram-tools dependency

10.117 FreedomBox 21.14 (2021-11-22)

10.117.1 Highlights

- tt-rss: Allow selection of a domain name

10.117.2 Other Changes

- : Split app initialization from app construction
- app: Introduce separate method for post initialization operations
- datetime: Avoid error when systemctl is not available
- debian: Fail build if no module dependencies found
- locale: Update translations for Swedish, Turkish, Ukrainian
- main: Drop initializing Django when listing dependencies

10.118 FreedomBox 21.13 (2021-11-08)

10.118.1 Highlights

- avahi, samba: Use systemd sandboxing
- components: Introduce new component - Packages
- security: Properly handle sandbox analysis of timer units

10.118.2 Other Changes

- email_server (not enabled yet):
  - Add buttons for managing aliases, domains, spam
  - Authenticate using PAM instead of LDAP
  - Delivery mail to /var/mail instead of home directory
  - Don't use user IDs when performing lookups
  - Drop hash DB and use sqlite3 directly
  - Use Django forms and views
- locale: Update translations for German, Swedish, Turkish, Ukrainian
- tests: Use BaseAppTests for functional tests of most apps
- utils: Fix ruamel.yaml deprecation warnings

10.119 FreedomBox 21.12 (2021-10-25)

- locale: Update translations for Bulgarian, Czech, French, German, Turkish, Ukrainian
- middleware: Don't show setup view to non-admin users
- performance: Add backup support (no data)
- storage: Pass optional mount point to partition expansion
- storage: tests: Fix tests for expanding disk partitions
- tests: Add BaseAppTests class for common functional tests

10.120 FreedomBox 21.11 (2021-10-11)

10.120.1 Highlights

- ttrss: Fix daemon not running sometimes on startup

10.120.2 Other Changes

- : Always pass check= argument to subprocess.run()
- : Convert all functional tests to python format
- : Move all systemd service files from /lib to /usr
- calibre: Run service only if when installed

- d/control: Allow building with python interpreter of any arch
- d/rules: Don't install and enable other systemd service files
- d/rules: Don't use setup.py to invoke tests, invoke directly instead
- email: Manage known installation conflicts
- locale: Update translation for Bulgarian, Ukrainian
- package: Add functions for removing packages
- performance: Cleanup code meant for cockpit version < 235
- pyproject.toml: Merge contents of .converagerc
- pyproject.toml: Merge contents of pytest.ini
- settings: Choose password hashing complexity suitable for SBCs
- setup: Show and remove conflicts before installation
- sso, translation: Help set language cookie when user logins in
- storage: tests: functional: Fix tests always getting skipped
- tests: Add some missed marks for functional tests
- tests: Add tests for action utilities
- tests: Improve handling of tests skipped by default
- tests: help: Add help view tests
- translation: Always set language cookie when switching language
- ttrss: Add systemd security hardening to daemon
- ttrss: tests: functional: Make subscription faster
- user: Accommodate Django 3.1 change for model choice iteration
- users: Help set language cookie when user profile is edited
- wordpress: Run service only if when installed and configured

10.121 FreedomBox 21.10 (2021-09-27)

10.121.1 Highlights

- locale: Update translations for German, Italian, Swedish, Turkish, Ukrainian

10.121.2 Other Changes

- Use Django gettext functions instead of ugettext
- Use allow/denylist instead white/blacklist in comments
- Use django.urls.re_path() instead of its alias url()
- Various isort fixes
- pyproject: Make isort consistent across execution environments
- settings: Set Django auto field type explicitly

- signals: Drop provider args when creating Signal object
- sso: Update usage of OpenSSL crypt signing API
- tests: Convert functional tests to python format
- tests: Introduce fixtures to make it easy to test actions
- tests: Show warning when app not available
- tests: Use common fixtures for testing actions module
- tests: Use newer splinter API for finding links
- views: Update utility for checking URL safety

10.122 FreedomBox 21.9 (2021-09-18)

10.122.1 Highlights

- mediawiki: Backup and restore uploaded files
- mediawiki: Enable a subset of default extensions

10.122.2 Other Changes

- apache: Update security settings
  - Drop support for GnuTLS
  - Drop support for SSLv3, TLSv1 and TLSv1.1
  - Enable and prioritize HTTP/2 protocol
  - Setup Mozilla recommended configuration
- locale: Update translations for Bulgarian, Chinese (Simplified), Dutch, Persian, Russian, Swedish, Turkish, Ukrainian
- mediawiki: Handle upgrade for 1.35
- mediawiki: Switch to MediaWiki 2020 logo
- plinth: remove diagnose command
- Add workaround for Django 3.2 with captcha 0.5.6

10.123 FreedomBox 21.8 (2021-08-30)

10.123.1 Highlights

- wordpress: New app to manage a WordPress site/blog

10.123.2 Other Changes

- d/control: Drop wireless-tools as recommends
- email: Basic app to manage an email server - Email server app is currently disabled by default, so it is not yet visible in the interface.
- locale: Update translations for Norwegian Bokmål, Ukrainian
- security: Remove display of past vulnerabilities

10.124 FreedomBox 21.7 (2021-08-16)

10.124.1 Highlights

- ttrss: Allow upgrade to version 21

10.124.2 Other Changes

- locale: Update translations for Albanian, Bengali, Chinese (Simplified), German, Indonesian, Norwegian Bokmål, Ukrainian, Vietnamese
- action_utils: Use flag to indicate freedombox package has been held
- debian: Ensure fuse gets replaced by fuse3

10.125 FreedomBox 21.6 (2021-05-31)

10.125.1 Highlights

- locale: Add Sinhala language
- locale: Add Vietnamese language
- backups: Change submit button to fix translation issues

10.125.2 Other Changes

- locale: Update translations for Chinese (Simplified), Chinese (Traditional), Dutch, French, German, Hungarian, Indonesian, Japanese, Portuguese, Sinhala, Swedish, Telugu, Turkish, Vietnamese

10.126 FreedomBox 21.5 (2021-04-19)

10.126.1 Highlights

- ejabberd: Add STUN/TURN configuration
- locale: Add Albanian language

10.126.2 Other Changes

- Update copyright year
- action_utils: Introduce utility for masking services
- ci: Merge with Salsa CI pipeline
- config, dynamicdns, pagekite: Remove incorrect use of str
- config: Convert entered domain name to lower case
- config: Disable rsyslog and syslog forwarding
- config: Fix tests related to user home directory
- config: Install and configure zram for swap
- container script: Must convert env. var. string to a Path object

- container: Work in the absence of systemd in PATH
- container: distribution as environment variable
- coturn: Mention ejabberd in app description
- coturn: Validate TURN URIs if provided in form
- debian: Add coverage to autopkgtest
- deluge, mldonkey, syncthing, transmission: Depend on nslcd.service
- deluge: Fix daemon user not in freedombox-share group after installation
- diagnostics: Use lock to protect results
- docs: Add filename to code snippets in tutorial
- docs: Add missing imports in tutorial
- docs: Add some troubleshooting information
- docs: Generate developer documentation
- docs: Improve Developer Documentation index page
- docs: Set the `version` attribute as required instead of optional
- dynamicdns: Convert entered domain name to lower case
- dynamicdns: Wait after changing domain name in tests
- first_boot: Use session to verify first boot welcome step
- letsencrypt: Always return a diagnostics result
- locale: Update translations for Albanian, Chinese (Simplified), Dutch, German, Greek, Indonesian, Polish, Spanish, Swedish, Turkish
- pagekite: Convert entered kite name to lower case
- security: Clarify vulnerability count and provide link to more info
- security: Ensure that fail2ban is not re-enabled on version increment
- security: Increment app version to reload fail2ban
- security: Move fail2ban default configuration to this app
- ssh, apache: Make fail2ban use systemd journald backend by default
- users: Fix unit test failures when LDAP is empty

10.127 FreedomBox 21.4.2 (2021-03-28)

10.127.1 Highlights

- firstboot: Use session to verify first boot welcome step

10.127.2 Other Changes

- locale: Update translations for German, Greek, Indonesian, Turkish
- manual: Update Contributing and Matrix Synapse pages

10.128 FreedomBox 21.4.1 (2021-03-13)

10.128.1 Highlights

- deluge, mldonkey, syncthing, transmission: Ensure nslcd is running before the service is started
- deluge: Fix daemon user not in freedombox-share group after installation

10.128.2 Other Changes

- config: Fix tests related to user home directory
- locale: Update translations for Dutch, German, Greek, Polish, Spanish, Swedish, Turkish

10.129 FreedomBox 21.4 (2021-02-28)

10.129.1 Highlights

- matrix-synapse: Auto configure STUN/TURN using coturn server

10.129.2 Other Changes

- coturn: Add new component for usage of coturn by other apps
- coturn: Minor refactor view to use utility to generate URIs
- coturn: Remove advanced flag, make app visible to all
- locale: Update translations for Dutch, French, German, Hungarian, Italian, Lithuanian, Norwegian Bokmål, Swedish, Turkish
- matrix-synapse: Update description to talk about TURN configuration
- plinth: Disable start rate limiting for service
- ui: Fix buttons jumping on click in snapshots page
- upgrades: Disable searx during dist-upgrade

10.130 FreedomBox 21.3 (2021-02-11)

10.130.1 Highlights

- zoph: Add new app to organize photos
  - Only available in Debian testing (bullseye) due to issues in buster.

10.130.2 Other Changes

- locale: Update translations for Dutch, Greek, Spanish, Swedish, Turkish
- sharing: Improve shares group access description
- upgrades: Add 10 minute delay before apt update
- upgrades: Disable apt snapshots during dist upgrade
- upgrades: Only check free space bytes before dist upgrade

10.131 FreedomBox 21.2 (2021-02-05)

10.131.1 Highlights

- calibre: Fix freedombox.local inaccessible after enabling app
- matrix-synapse: Install python3-psycopg2 from backports

10.131.2 Other Changes

- backups: schedule: tests: Fix failures due to long test run
- jsxc: Fix issues with jQuery >= 3.5.0
- locale: Update translations for Bengali, Dutch, French, German, Hungarian, Italian, Polish, Russian, Spanish, Swedish, Turkish
- mediawiki: Fix app installation process doesn't display status information
- mediawiki: Set default logo to mediawiki.png
- minidlna: Implement force upgrading from older version
- minidlna: Minor refactor of media directory handling
- plinth: Show running spinner when app installation is in progress
- radicale: Allow older 2.x release to upgrade to 3.x
- roundcube: Allow upgrade to 1.4.*
- tests: Update functional tests default config
- upgrades: Add notifications for dist upgrade
- upgrades: Increment version for MatrixSynapse 1.26

10.132 FreedomBox 21.1 (2021-01-25)

10.132.1 Highlights

- backups: Add scheduled backups for each location

10.132.2 Other Changes

- container script: Various improvements
- locale: Update translations for Bulgarian, Chinese (Simplified), Chinese (Traditional), Czech, Danish, Dutch, French, Galician, German, Greek, Gujarati, Hindi, Hungarian, Italian, Lithuanian, Norwegian Bokmål, Persian, Polish, Portuguese, Russian, Serbian, Slovenian, Spanish, Swedish, Turkish, Ukrainian
- networks: Change connection type to a radio button
- networks: Hide deactivate/remove buttons for primary connections
- networks: Prevent unintended changes to primary connection.
- networks: Separate the delete button and color it differently
- networks: Use radio buttons for network modes
- performance: Fix web client link to Cockpit

- plinth: Fix disable daemon when service alias is provided
- setup: Enable essential apps that use firewall
- syncthing: Create LDAP group name different from system group
- syncthing: Hide unnecessary security warning
- tahoe: Disable app
- ui: New style for select all checkbox
- upgrades: Require at least 5 GB free space for dist upgrade

10.133 FreedomBox 21.0 (2021-01-11)

10.133.1 Highlights

- apache2: Allow downloads in openvpn and backups with latest browsers

10.133.2 Other Changes

- locale: Update translations for Dutch, French, German, Hungarian, Polish, Spanish, Swedish, Turkish
- app: Add locked flag
- app: component: Add app_id and app properties
- app: info: Move client validation to info component
- backups: Add new component for backup and restore
- backups: Don't open a new window for downloading backups
- dev-container: 'up' command: Show banner also when container is already running
- dev-container: Add command to print container IP address
- dev-container: Add subcommand to run tests
- doc: dev: Update the tutorial to reflect latest API/code
- ejabberd: functional tests: Wait until the jsxc buddy list is loaded
- functional tests: Make tests compatible with pytest-bdd v4.0
- functional-tests: Fix installation errors in install.sh script
- gitweb: Add functional tests for git-access group
- gitweb: tests: functional: Fix test failures in localized environment
- mumble: Updated mumla and removed plumble from clients list
- openvpn: Don't show running status on download profile button
- plinth: Fix daemon is enabled check when service alias is provided
- radicale: Fix backup and restore of configuration
- tests: functional: Improve creating users in tests
- transmission: Show port forwarding information
- transmission: Update description

- upgrades: Add service for dist upgrade
- upgrades: Ensure freedombox package is upgraded during dist upgrade
- upgrades: Hold tt-rss during dist upgrade, if available
- upgrades: Install python3-systemd for unattended-upgrades
- upgrades: Restart FreedomBox service at end of dist-upgrade
- upgrades: Use full path to searx action script
- users: Skip action script tests if LDAP is not set up

10.134 FreedomBox 20.21 (2020-12-28)

10.134.1 Highlights

- apache: Create snake oil certificate if not exists
  - Fixes an issue when installing FreedomBox on Hetzner Cloud's Debian image
- calibre: Fix link to manual page

10.134.2 Other Changes

- deluge: Require user to be in bit-torrent group to access
- locale: Update translations for German, Hungarian, Polish, Russian, Spanish, Swedish
- security: Fix access denied for user daemon from cron
- upgrades: Allow grub-pc upgrade without reinstalling grub
- upgrades: Update searx search engines during dist upgrade
- users: Remove timeout when creating Samba user

10.135 FreedomBox 20.20.1 (2020-12-19)

10.135.1 Highlights

- config: Skip homepage test on buildd
- ui: Migrate from bootstrap 3 to bootstrap 4

10.135.2 Other Changes

- apache: Disallow all inline styling in sandbox settings
- gitweb: Make functional tests compatible with pytest-bdd v4.0
- javascript: Fix disabled submit buttons when navigating back to a page
- locale: Update translations for Dutch, German, Turkish
- ui: Adopt a consistent and new table style

10.136 FreedomBox 20.20 (2020-12-14)

10.136.1 Highlights

- config: Add user websites as choices for homepage config
- templates: Make toggle button responsive

10.136.2 Other Changes

- apache: Add app name for diagnostics
- diagnostics: Improve exception handling in app diagnostics
- diagnostics: Show app name and fallback to app id if not exist
- locale: Update translations for Dutch, French, German, Portuguese, Spanish, Swedish, Turkish
- mumble: Implement force upgrade for 1.3.*
- snapshot: Check that / is a btrfs subvolume before setup
- upgrades: Hold mumble-server during dist upgrade

10.137 FreedomBox 20.19 (2020-11-30)

10.137.1 Highlights

- openvpn: Create user group "vpn"
- upgrades: Add first boot step to run initial update

10.137.2 Other Changes

- bepasty: Apply translation to autogenerated comments
- locale: Update translations for Bengali, Dutch, German, Spanish
- networks: Apply translation to a tooltip
- samba: Show toggle buttons and share names
- snapshots: Translate snapshot types (field description)
- upgrades: Fix sources list for dist upgrade from buster
- upgrades: Hold freedombox package during dist upgrade

10.138 FreedomBox 20.18.1 (2020-11-23)

- locale: Update translations for Dutch, French, German, Italian, Norwegian Bokmål, Spanish, Swedish, Turkish
- sso: Fix regression in auth-pubtkt configuration

10.139 FreedomBox 20.18 (2020-11-16)

10.139.1 Highlights

- openvpn: Support Elliptic Curve Cryptography (ECC)
  - If you are already using OpenVPN, you can migrate to ECC to improve speed and security. Visit the OpenVPN page in the FreedomBox interface to perform the one-time migration, and to re-download the client profiles.

10.139.2 Other Changes

- dynamicdns: Handle IPv6
- locale: Update translations for Dutch, French, German, Italian, Spanish
- openvpn: Cleanup easyrsa 2 to 3 upgrade code
- openvpn: Remove explicit setup step

10.140 FreedomBox 20.17.1 (2020-11-07)

- ci: Fix flake8 errors
- debian: Rename source package to freedombox
- locale: Update translations for German, Italian, Turkish
- pubtkt: Fix Python format language errors

10.141 FreedomBox 20.17 (2020-11-02)

10.141.1 Highlights

- locale: Add Chinese (Traditional) translation
- mediawiki: Add action to set domain name
- upgrades: Add a setting to enable dist upgrade

10.141.2 Other Changes

- apache: setup uwsgi by default
- backups: i18n: Mark form success messages for translation
- locale: Update translations for Danish, French, German, Italian, Norwegian Bokmål, Polish, Russian, Spanish, Swedish, Telugu, Turkish
- mediawiki: Ensure password file is not empty
- networks: css: Make button wider in network list
- networks: i18n: Mark string for translation on delete page
- networks: i18n: Mark various strings for translation
- notifications: i18n: Mark app names and extra data for translation
- package: i18n: Mark progress status strings for translation
- upgrades: Disable the option when not able to dist upgrade

10.142 FreedomBox 20.16 (2020-10-19)

10.142.1 Highlights

- app: Add donation buttons on app pages
- updates: Eliminate delay and better status for manual upgrade

10.142.2 Other Changes

- calibre: Update group description to reflect 'using' app
- diagnostics: Lazy format all diagnostic test strings properly
- diagnostics: Show low system memory notifications
- help: Link to updates page when new version is available
- locale: Update translations for Chinese (Simplified), French, Greek, Norwegian Bokmål, Russian, Slovenian, Spanish, Swedish, Turkish
- notifications: Show severity level on every notification
- upgrades: Add status section showing version and upgrade status

10.143 FreedomBox 20.15 (2020-10-05)

10.143.1 Highlights

- calibre: Add new e-book library app
- mumble: configure letsencrypt component
- upgrades: Detect and upgrade to next stable release

10.143.2 Other Changes

- bepasty: Change default permissions to 'read'
- container: Assign virtual network interface to trusted firewall zone
- container: Handle edge cases with container update
- coturn: Don't handle certificates if not installed
- debian/control: Add sshpass as build dependency
- doc: Before fetching, drop all old to cleanup deleted pages/images
- doc: dev: Link to list of potential apps from tutorial
- dynamicdns: Drop unnecessary code to set app as enabled
- locale: Update translations for French, Norwegian Bokmål, Portuguese, Spanish, Swedish, Turkish
- module_loader, web_framework: Update console log messages
- mumble: Store and use a single domain for TLS certificate setup
- pagekite: Don't announce unconfigured kite as a valid domain
- pagekite: Don't update names module if not installed

- quassel: Don't handle certificates if not installed
- ssh: action script: Require user credentials when editing ssh keys
- tests: functional: Simplify calling the login helper
- tor: Don't check if enabled when not installed
- upgrades: Check free space before dist-upgrade
- upgrades: Extend function to check for normal dist availability
- upgrades: Set a flag so interrupted dist-upgrade can be continued
- users: Deal with admin user already existing during first boot
- users: Require admin credentials when creating or editing a user

10.144 FreedomBox 20.14.1 (2020-09-23)

- cockpit: Don't show home page icon to non-admin users
- locale: Update translations for French, German, Norwegian Bokmål, Russian, Turkish
- minidlna: Fix typo DNLA -> DLNA
- module_loader: Load/process all essential modules before others

10.145 FreedomBox 20.14 (2020-09-15)

10.145.1 Highlights

- apache: Disable mod_status (CVE-2020-25073)
- bepasty: New app for file upload and sharing
- matrixsynapse: Allow upgrade to version 1.19

10.145.2 Other Changes

- apps: Remove Coquelicot
- backups: Make app available by default
- debian: Add newline to end of /var/lib/plinth/firstboot-wizard-secret
- debian: Don't show first wizard secret on command line
- debian: Temporarily revert source package rename
- diagnostics: Prevent showing running status on diagnostics menu item
- doc: Add moinmoin wiki parser
- doc: Fix wiki links in manual
- ejabberd, mumble, wireguard: Update Apple app links
- ejabberd: Use new ruamel.yaml API and allow duplicate keys
- firewall: Show port forwarding info contextually
- firewall: Show port forwarding info in tabular format

- gitweb: Add ability to change default branch
- gitweb: Fix enable auth webserver component on app init
- help, networks: Clarify i18n different contexts for "Manual"
- i18n: Mark strings missed for translation
- ikiwiki: Validate a path when deleting wiki or blog
- js: Don’t show running status on buttons pulled to right
- jsxc, sharing, wireguard: Add ’Learn more...’ link for help pages
- locale: Update translations for Danish, Dutch, Galician, German, Hungarian, Italian, Spanish, Swedish, Russian, Turkish
- matrixsynapse: Perform a one time conversion to new config format
- matrixsynapse: Rename Riot to Element
- matrixsynapse: Use conf.d snippets
- radicale: Remove code to handle 1.x
- radicale: Stop service during backup and restore
- samba: Hide common system partitions
- snapshots: Clarify description for disabling yearly snapshots
- ssh: Disallow managing keys for the root user
- storage: Fix expanding partitions on GPT partition tables
- upgrades, security: Update the messages describing backports
- upgrades: Add first boot step to configure backports
- upgrades: Change backports activation message wording
- upgrades: Display correct backports info for unstable
- upgrades: security: Don’t use technical term ’backports’ in UI
- wireguard: Remove hardcoded Windows client version

10.146 FreedomBox 20.13 (2020-07-18)

10.146.1 Highlights

- upgrades: Update apt cache before manual update
- minidlna: Do not expose statistics over public web

10.146.2 Other Changes

- backups: Allow remote repository usernames to start with numbers
- locale: Update translations for Chinese (Simplified), Hungarian, Kannada, Norwegian Bokmål, Spanish, Swedish
- security: Move backports notice to security page
- upgrades: Add button to activate backports if needed for current release
- debian: Rename source package from plinth to freedombox

10.147 FreedomBox 20.12.1 (2020-07-05)

- cfg, frontpage: Ignore errors while reading config and shortcuts
- locale: Update translations for French, German, and Norwegian Bokmål

10.148 FreedomBox 20.12 (2020-06-29)

10.148.1 Highlights

- apt: Recover from errors before installing apps or updating system
- apache: Add strict content security policy, sandbox and other security headers
- storage: Allow ejecting SATA disks
- configuration: Allow changes using .d drop-in files

10.148.2 Other Changes

- configuration: Move default configuration into source code
- configuration: Read from multiple locations in /etc/ and /usr/share/
- debian: Add ssl-cert and nscd as proper dependencies
- frontpage: Allow adding shotcuts using .d drop-in files
- frontpage: Read shortcuts from multiple locations in /etc/, /usr/share and /var/lib
- locale: Update translations for Czech, Danish, French, German, Russian, Spanish, Swedish, Telugu, Turkish
- storage: Automount system disks without partition table but ignore all loopback devices
- storage: Allow ejecting SATA disks
- storage: Show only physical disks and not all mount points
- upgrades: Skip enabling backports on testing and unstable
- upgrades: Show more logs
- ui: Show a spinner and disable button on form submit

10.149 FreedomBox 20.11 (2020-06-15)

10.149.1 Top Highlight

- locale: Add new translation for Arabic (Saudi Arabia)

10.149.2 Other Changes

- javascript: Remove use of Turbolinks library
- locale: Update translations for French, Norwegian Bokmål, German, Swedish, Polish, and Spanish
- matrixsynapse: Handle upgrade to versions 1.15.x
- upgrades: Avoid manual update interruption when upgrading freedombox package
- upgrades: Don’t enable backports on Debian derivatives

10.150 FreedomBox 20.10 (2020-06-01)

10.150.1 Top Highlights

- pagekite: Fix expired certificates causing connection failures
- tor: Fix problems with running a relay

10.150.2 Other Changes

- backups: Add optional field - Name
- cockpit: Promote for advanced storage/firewalld/networking ops
- firewall: Don't show tun interface in internal zone warning
- firewall: Mention that internal services are available over VPN
- ikiwiki: Enable 'attachment' plugin by default
- locale: Update translations for Spanish, French, Russian, Norwegian Bokmål, Czech, Hungarian, and Greek
- minidlna: Add link to manual page
- minidlna: Fix internationalization for name of the app
- mldonkey: Add app to freedombox-share group
- openvpn: Use app toggle button and common app view
- radicale: Fix link in description to clients
- samba: Add clients information
- templates: Fix setup state check
- users: Avoid error when user's groups cannot be parsed

10.151 FreedomBox 20.9 (2020-05-18)

10.151.1 Top Highlights

- performance: Add app for system monitoring
- upgrades: Restart services and system when needed after upgrades
  - System restart will happen at 02:00 local time

10.151.2 Other Changes

- bind: Add service alias for bind9 -> named
- firewall: Reload firewalld so it works with newly installed services
- first_setup: Fix regression with logo not showing
- locale: Update translations for Norwegian Bokmål, German, Swedish, Spanish, and Russian
- mediawiki: Stop jobrunner during backup/restore
- minidlna: Stop service during backup/restore

- mumble: Stop service during backup/restore
- package: Fix error log when checking if package manager is busy
- performance: Launch the Cockpit graphs directly if possible
- quassel: Fix stopping service during backup/restore
- quassel: Use systemd sandboxing features
- samba: Change description to Network File Storage
- snapshot: Fix issues with restore and delete
- snapshot: Set as essential module
- storage: Auto-mount disks, notify of failing disks
- tor: Fix stopping service during backup/restore

10.152 FreedomBox 20.8 (2020-05-04)

- syncthing: Add service to freedombox-share group
- users: When adding service to sharing group, only restart if already running
- datetime: Ignore time synchronization service in containers and virtual machines
- minidlna: Make app installable inside unprivileged container
- web_server: Suppress warnings that static directories don't exist
- debian: Remove unused timer
- static: Use SVG logo during first wizard welcome step
- static: Reduce the size of the background noise image
- setup.py: Don't install/ship .po files
- static: Don't ship visual design file and unused images
- all: Update links to repository and project page
- coturn: Add app to manage Coturn TURN/STUN server
- mediawiki: Partial fix for installing on testing
- datetime: Disable diagnostics when no tests are available
- data: Print hostname and IP addresses before console login
- snapshot: Fix message when not available
- snapshot: Fix title
- mumble: Add Mumla to the list of clients
- locale: Update translations for Spanish, Telugu, Russian, German, French, and Swedish

10.153 FreedomBox 20.7 (2020-04-20)

- matrixsynapse: Fix initial installation and upgrade from backports
- gitweb: Improve error handling when creating repository
- locale: Update translations for French, Serbian, and Telugu

10.154 FreedomBox 20.6.1 (2020-04-11)

- users: Restore line of help text that was accidentally dropped
- debian: Add firmware-ath9k-htc to Recommends
- gitweb: Use proper ellipsis char when showing clone progress
- locale: Update translations for Norwegian Bokmål, German, French, Portuguese, Italian, Russian, and Serbian

10.155 FreedomBox 20.6 (2020-04-06)

- app: Ensure toggle buttons work independently of configuration form
- networks, monkeysphere: Make styling more specific to avoid interference
- syncthing: Update description to mention 'syncthing' group
- radicale: Support upgrade up to any 2.x version
- packages: Hold freedombox package during package installs
- users: Add component for managing users and groups
- app: Fix grammar in developer documentation string
- ikiwiki: Disable public edits of blog pages
- ikiwiki: Add moderation of blog comments
- firewalld: Support upgrade up to any 0.8.x version
- infinoted: Fix permissions of sync directory
- locale: Added Serbian translation
- locale: Update translations for Russian, French, German, Czech, Italian, Hindi, Telugu, and Spanish

10.156 FreedomBox 20.5.1 (2020-03-26)

- networks: Update label wording in topology form
- jsxc: Fix issue with serving static files
- debian: Separate binary packages for each language manual
- locale: Update translations for Norwegian Bokmål and German

10.157 FreedomBox 20.5 (2020-03-23)

- app: Fix description block in app header
- pagekite: Don't signal new domain on init if app is disabled
- pagekite: Don't attempt to notify about domain if app is disabled
- pagekite: Remove app enabled checking from getting configuration
- pagekite: On enable/disable, add/remove domain from names module
- pagekite: Fix an error message in custom services form
- matrixsynapse: Handle release of matrix-synapse 1.11

- setup: Fix regression to force-upgrade caused by Info changes
- pagekite: Don’t allow non-unique custom services
- index: Reintroduce clients button in front page
- upgrades: Don’t ship apt backport preferences file
- upgrades: Use internal scheduler instead of systemd timer
- shadowsocks: Change default configuration
- shadowsocks: Fix incorrect setting of state directory
- shadowsocks: When editing configuration, don’t re-enable
- mediawiki: Don’t allow anonymous edits
- names: Fix Local Network Domain is not shown
- shadowshocks: Fix setting configuration on Buster
- locale: Update translations for Swedish, Spanish, and French

10.158 FreedomBox 20.4 (2020-03-09)

- apache: Handle transition to php 7.4
- app: Fix showing app name in port forwarding information
- apps: Do not show status block if service is running
- i2p: New style app page layout
- locale: Update translations for French, Telugu, Spanish, and Swedish
- networks: Add first boot step for network topology wizard
- networks: Add form for network topology
- networks: Don’t show router wizard if not behind a router
- networks, firewall: Support newer version of policykit
- networks: Fixes for networks wizards access and user experience
- networks: If topology wizard is skipped, skip router wizard too
- networks: Show router wizard before Internet connection type wizard
- plinth: Increase sqlite busy timeout from default 5s to 30s
- quassel: Fix unable to disable application without choosing a domain name
- shadowsocks: Move user settings to state directory
- storage: Directory selection form improvements
- transmission: Allow to submit download directory if it is creatable
- upgrades: Clean apt cache every week
- views: Improve template security

10.159 FreedomBox 20.3 (2020-02-24)

- apps: Update style for toggle button
- apps: Drop border shadow for app icon in mobile view
- apps: Show short description as secondary title
- apps: Remove css filters and glow from app icons
- cards: Remove the transition delay on hover effect
- system: Implement new style for cards
- framework: Generate secret key (existing sessions will get logged out)
- framework: Cleanup expired sessions every week
- networks: Add setting for internet connection type
- networks: Ask about internet connection type during setup
- shadowsocks: Fix shadowsocks not able to start
- jsxc: Bypass issue with stronghold to get the app working again
- monkeysphere: Fix regression with reading Apache configuration
- help: Fix attribute on download manual button
- firewall: Improve speed of some operations using DBus API
- css: Add missing license identifier on some CSS files
- deluge: Use safer method for editing configuration
- deluge: More reliable initial configuration setup
- samba: Add link to manual page
- searx: Update search engines for 0.16.0
- openvpn: Fix spelling for Tunnelblick
- bind: Show served domains
- Update translations for German, Swedish, Italian, Spanish, Norwegian Bokmål, Hungarian, Polish, and French

10.160 FreedomBox 20.2 (2020-02-10)

- networks: Support virtual Ethernet (veth) devices
- diagnostics: Show firewall service status
- storage: Show disks if FreedomBox is running in an unprivileged container
- service: Stop service not before but after disabling it
- users: Use more precise username validation
- sso, users: Turn off autocapitalization on the username field
- help: Fix anchor hidden under navbar
- searx: Fix installation issue for 0.16.0
- firewall: Show Run Diagnostics button in app

- glib: Introduce method to schedule an operation at regular intervals
- notification: Show a drop down from main navbar for notifications
- storage: Show low disk space warning using notifications API
- upgrades: Show notification when FreedomBox is updated
- security: Add Sandbox Coverage to report page
- matrixsynapse: Enable systemd sandboxing
- locale: Update translations for Telugu, French, Norwegian Bokmål, German, Spanish, and Swedish

10.161 FreedomBox 20.1 (2020-01-27)

- deluge: Allow to set a download directory
- deluge: Fix installation failure on slow machine
- storage: Make external disk mounts accessible to other users
- gitweb: Add link to the manual page
- style: Fix incorrect margins for containers in mobile view
- style: Fix responsiveness for app header
- network: Fix activating connections that don't have real devices
- wireguard: Add WireGuard VPN app
- networks: Add router configuration page
- networks: Add first boot step for router config helper
- bind: Enable sandboxing for bind service
- locale: Updated translations for Dutch, Norwegian Bokmål, German, Spanish, Swedish, French, and Greek

10.162 FreedomBox 20.0 (2020-01-13)

- samba: Improve speed of actions
- deluge: Manage deluged service and connect automatically from web interface
- openvpn: Enable support for communication among all clients
- storage: Ignore errors resizing partition during initial setup
- storage: Make partition resizing work with parted 3.3
- debian: Add powermgmt-base as recommended package
- openvpn: Enable IPv6 for server and client outside the tunnel
- networks: Fix crashing when accessing network manager D-Bus API
- mediawiki: Use a mobile-friendly skin by default
- mediawiki: Allow admin to set default skin
- matrixsynapse: Allow upgrade to 1.8.*
- security: Add explanation of sandboxing
- Update translations for Greek, German, Swedish, Hungarian, Norwegian Bokmål, and French

10.163 FreedomBox 19.24 (2019-12-30)

- app: Fix JavaScript doesn't run on first visit
- samba: Add private shares
- firewall: Support upgrading firewalld to 0.8
- deluge: Add systemd sandboxing features
- infinoted: Add systemd sandboxing features
- storage: Add systemd sandboxing features to udiskie service
- upgrades: Add systemd sandboxing features to repository setup service
- security: List whether each app is sandboxed
- mediawiki: Avoid delay in update script
- diagnostics: Use new component based API for all diagnostic tests
- minidlna: Fix showing clients information
- mediawiki: Fix problem with session cache failing logins
- locale: Update translations for French, German, Swedish, Greek, Hungarian, Norwegian Bokmål, and Dutch

10.164 FreedomBox 19.23 (2019-12-16)

- minidlna: New app for MiniDLNA (Simple Media Server)
- apps: Show app icons in app pages
- apps: Implement responsive layout for app pages
- samba: Recursively set open share directory permissions
- transmission: Add directory selection form
- mumble: Add option to set SuperUser password
- cockpit: Extend apps description with access info
- cockpit: Add list of valid urls to access the app
- Update translations for French, German, Spanish, Portuguese, and Swedish

10.165 FreedomBox 19.22 (2019-12-02)

- samba: Add new app for Samba file sharing
- pagekite: Remove tabs in the configuration page
- openvpn: Fix text with manual link
- pagekite: Show existing services only if there are any
- pagekite: Move Custom Services under Configuration
- pagekite: Use the new app toggle button
- openvpn: Add client apps
- backups: Fix title not appearing

- diagnostics: Don’t run on disabled modules
- apps: Remove link to webapps in app descriptions
- interface: Fix error with app toggle input
- templates: Add toolbar for apps
- toolbar: Move diagnostics button into dropdown menu
- ssh: Fix Avahi SFTP service file
- diagnostics: Fix IPv6 failures
- matrix-synapse: Fix installation of 1.5 from buster-backports
- app: Fix javascript constant redeclaration error
- ikiwiki: Move the create button to manage section
- gitweb: Move create button into manage section
- networks: Move actions button into connection section
- users: Move create button into users section
- locale: Update translations for French, German, and Swedish

10.166 FreedomBox 19.21 (2019-11-18)

- gitweb: Allow to import from a remote repository
- interface: Disable turbolinks on links that don’t point to /plinth/...
- backups: Show proper error when SSH server is not reachable
- tor: Rename "Hidden Service" to "Onion Service"
- ejabberd: Handle case where domain name is not set
- tahoe: Mark Tahoe-LAFS as an advanced app
- searx: Set safe_search to Moderate by default
- backups: Make verify ssh host page string translatable
- backups: Simplify SSH fingerprint verification command
- doc: Fix unavailability of manual images
- tor: Fix port diagnostics by correcting port data type
- tor: Expect obfs service to be also available on IPv6
- tor: Listen on IPv6 for OrPort
- clients: implement launch button feature
- apps: Implement toggle button in apps pages
- Update translations for German, Hungarian, Swedish, Norwegian Bokmål, French, Polish

10.167 FreedomBox 19.20 (2019-11-04)

- doc: Add Spanish manual
- ssh: Add option to disable password authentication
- sharing: Fix wrong links on Apache2 directory index page
- gitweb: Set correct access rights after enabling application
- gitweb: Fix links leading to blank page
- gitweb: Set proper access after restoration of a backup
- snapshot: Sort snapshot list from newest to oldest
- infinoted: Add missing manual page link
- backups: Fix typo
- Update translations for German, Spanish, Swedish, Czech, French, Norwegian Bokmål, Hungarian

10.168 FreedomBox 19.19 (2019-10-21)

- gitweb: New app for simple git hosting
- ikiwiki: Allow full Unicode text in wiki/blog title names
- users: reload Apache2 to flush LDAP cache after user operations
- ssh: Show server fingerprints in SSH page
- frontpage: Show public shortcuts to all users regardless of group
- ikiwiki: Remove extra create button when no wiki/blog is present
- quassel: Add Let's Encrypt component for certificates
- Update translations for Czech, French, Bulgarian, Dutch, German, and Norwegian Bokmål

10.169 FreedomBox 19.18 (2019-10-07)

- diagnostics: Ensure that exceptions are reported as failures
- users: Rearrange UI to match with other apps
- upgrades, ikiwiki, networks, backups: Replace page tabs with buttons
- dynamicdns, i2p, pagekite, snapshot: Cleanup page templates
- deluge: Support deluge 2 by starting it properly
- minetest: Remove mod-torches no longer available in testing/unstable
- security: Add past vulnerabilities count, move report to new page
- Update translations for Spanish, Norwegian Bokmål, German

10.170 FreedomBox 19.17 (2019-09-23)

- firstboot: Add new help menu to firstboot navbar
- firstboot: Hide left menu during first boot as intended
- Update translations for Chinese (Simplified) and Czech
- Fix tests for letsencrypt and tor

10.171 FreedomBox 19.16 (2019-09-09)

- backups: Allow adding backup repositories on multiple disks
- help: Add buttons for contribute, support, and feedback
- action_utils: Workaround problem with setting debconf answers
- views: Fix failure in redirecting from language selection page
- manual: Move PDF download link to HTML manual page
- help: Convert help icon in the navbar to dropdown
- ejabberd: Fix listen port configuration for ejabberd 19.x
- cockpit, ejabberd: Prevent restart on freedombox startup
- ejabberd: Perform host/domain name operations only when installed
- logging: Improve formatting and reduce noise
- translations: Update Hungarian, German, Italian, French, and Norwegian Bokmål

10.172 FreedomBox 19.15 (2019-08-26)

- security: Hide vulnerability table by default
- names: Perform better layout of domain names table on small screens
- cockpit: Apply domain name changes immediately
- ejabberd: Prevent processing empty domain name
- config: Send hostname change signal only after fully processing it
- letsencrypt: Don't try to obtain certificates for .local domains
- avahi: Expose .local domain as a proper domain
- cockpit: Make essential and install by default
- tt-rss: Force upgrade to 18.12-1.1 and beyond
- updates: Allow matrix-synapse 1.3 to be installed for buster users
- javascript: Don't resubmit when refreshing the page
- storage: Fix regression with restoring backups with storage
- matrix-synapse: Use recommended reverse proxy configuration
- Update translations for German, Hungarian, and Norwegian Bokmål

10.173 FreedomBox 19.14 (2019-08-12)

- storage: Handle all device paths during eject
- storage: Fix incorrect internationalization when throwing an error
- upgrades: Use collapsible-button style for logs
- firewall: Allow automatic upgrade to 0.7.x
- upgrades: Handle release info change
- frontpage: Fix regression with loading custom shortcuts
- names: Add dynamic domain name
- names: Add button to configure each type of name
- names: Update page layout for clearer presentation
- names: Introduce new API for domain name handling
- api: Fix regression with listing only enabled apps in mobile app
- Update translations for Czech, Hungarian, French, Chinese (Simplified), Turkish, Polish, and Norwegian Bokmål

10.174 FreedomBox 19.13 (2019-07-29)

- backups: Make UI more consistent with other apps
- backups: Make backup location tables collapsible
- Updated translations for Chinese (Simplified), German, and Norwegian Bokmål
- help: Show security notice when backports are in use
- security: Show vulnerability counts

10.175 FreedomBox 19.12 (2019-07-22)

- sharing: Allow directories to be publicly shared
- backups: Add option to select/deselect all apps for backup or restore
- dbus: Allow plinth user to own FreedomBox DBus service
- letsencrypt: Simplify renewal hooks implementation
- cockpit: Don't handle domains if app is not installed
- dynamicdns: Send domain added signal properly during init
- ejabberd: Backup and restore TLS certificates
- Started new Galician translation on Weblate
- Updated translations for Czech, Norwegian Bokmål, Hungarian, Spanish, Telugu, Chinese (Simplified), German, Turkish, and Russian

10.176 FreedomBox 19.2.2 (2019-07-17)

This release does not contain any functional changes, but fixes test failures when building the package.

10.177 FreedomBox 19.2.1 (2019-07-09)

This is a bugfix release for 19.2.

- dbus: Allow plinth user to own FreedomBox DBus service

10.178 FreedomBox 19.11 (2019-07-08)

- backups: Fixes to issues while adding SSH remotes:
  - Improve UX of adding ssh remote
  - Avoid creating duplicate SSH remotes
  - Fix issue with repository not being initialized
  - Verify SSH hostkey before mounting
  - Allow SSH directory paths with : in them
  - Require passphrase for encryption in add repository form
  - Don't send passphrase on the command line
  - Un-mount SSH repositories before deleting them
- matrixsynapse: Fix missing translation mark
- Started new Greek translation on Weblate
- Updated translations for Chinese (Simplified), Hungarian, Spanish, and Russian

10.179 FreedomBox 19.10 (2019-06-24)

- syncthing: Open firewall ports for listening and discovery
- radicale: Workaround issue with creating log directory
- Update translations for Turkish, German, Czech, Norwegian Bokmål, and Portuguese
- Introduce components for firewall, webserver, uwsgi, and daemons

10.180 FreedomBox 19.9 (2019-06-10)

- config: Add option to show advanced apps, which are hidden by default
- monkeysphere: Hide by default
- searx: Add option to allow public access to the application
- Introduce component architecture for apps, with components for menus and shortcuts
- Start new translation for Bulgarian
- Update translations for Turkish and Norwegian Bokmål

10.181 FreedomBox 19.8 (2019-05-27)

- Switch to using SVG icons for all apps.
- Updated translations for Czech, Norwegian Bokmål, Hungarian, German, Turkish, and Spanish.

10.182 FreedomBox 19.7 (2019-05-13)

- i2p: Include default favorites.
- Separate enabled and disabled apps.
- Display port forwarding info for apps.
- Added Slovenian translation.
- Updated translations for Dutch, German, Hungarian, Norwegian Bokmål, Polish, Portuguese, Telugu.

10.183 FreedomBox 19.6 (2019-04-29)

- i2p: Enable new application for I2P Anonymity Network.
- Updated translations for Czech, German, Norwegian Bokmål, and Turkish.
- letsencrypt: Provide link to configure domain if not configured.
- firewall: Show port numbers and types.

10.184 FreedomBox 19.5 (2019-04-15)

- storage: Use more reliable method to list disks and disk space usage.
- Updated translations for Russian and German.

10.185 FreedomBox 19.4 (2019-04-01)

- clients: Open web app in a new browser tab
- matrix-synapse: Change client diagnostics url
- minetest: Fix duplicate domain names being displayed in UI
- storage: Do not show an eject button on /boot partitions
- letsencrypt: Call letsencrypt manage_hooks with correct arguments
- dynamicdns: Install module by default
- storage: Don't check type of the disk for / and /boot
- storage: Don't log error when checking if partition is expandable
- Updated translations for Norwegian Bokmål, Czech, German, Hungarian, Spanish, German, and Russian.

10.186 FreedomBox 19.3 (2019-03-18)

- UI: Move tabs below descriptions.
- firewall: Style heading
- names: Add description
- pagekite: Change heading text
- ikiwiki: Consistent styling for delete warning page
- main: Show service version in logs
- setup: Organize data files into various apps
- Updated translations for Czech, Hungarian, Norwegian Bokmål, Spanish, German, French, Italian, and Turkish.

10.187 FreedomBox 19.2 (2019-03-02)

- config: Fix Ikiwiki entries not showing up as default apps
- config: Migrate default app configuration to new conf file
- config: Rename Default App to Webserver Home Page
- config: Add option to use Apache's default home page as home page
- config: Fix error when setting JSXC as the home page
- Disable Coquelicot for Buster release
- matrix-synapse: Fix LDAP login issue
- config: Revert changes in freedombox.conf to avoid conffile prompt
- openvpn: Migration from easy-rsa 2 to 3 for existing installations
- tor: Use fixed 9001 port for relaying
- package: Implement identifying packages that need conffile prompts
- setup: Trigger force upgrade for app that implement it
- bind: Handle conffile prompt during upgrade
- apache: Pre-enable necessary apache modules
- apache: Use cgid module instead of cgi
- openvpn: Make frontpage shortcut appear after an upgrade
- openvpn: Work around firewalld bug 919517
- firewalld: Implement upgrading from 0.4.x to 0.6.x
- ttrss: Implement upgrade from 17.4 to 18.12
- radicale: Add description of web interface
- ttrss: Add backup support
- security: Migrate access config to new file
- Updated translations for Czech, Hungarian, Norwegian Bokmål, Spanish, German, Telugu.

10.188 FreedomBox 19.1 (2019-02-14)

- radicale: Increment module version to trigger upgrade handling
- radicale: Remove obsolete diagnostics
- radicale: Fix server URLs in client info
- Updated translations for Czech, Norwegian Bokmål, and Spanish.
- setup: Add option to handle configuration prompts during install
- radicale: Simplify upgrading to newer packages
- matrixsynapse: Use Let's Encrypt certificates

10.189 FreedomBox 19.0 (2019-02-09)

- mldonkey: Add some more clients to the module page
- mldonkey: Add to the description the three available front-ends
- monkeysphere: Fix handling of multiple domains and keys
- monkeysphere: Fix regression with reading new apache domain config
- apache: Switch to mod_ssl from mod_gnutls
- mldonkey: Enable app
- upgrades: Fix priority for buster-backports version
- upgrades: Fix premature adding of buster-backports sources
- Updated translations for Czech, German, and Spanish
- Switched to a new version number scheme: YY.N
  - YY is the year of release.
  - N is the release number within that year.

10.190 Version 0.49.1 (2019-02-07)

- ui: Fix regression with configure button in home page.
- backups: Rename 'Abort' buttons to 'Cancel'.
- backups: Use icon for add repository button.
- backups: Move subsubmenu below description.
- backups: Add title and description to other pages.
- backups: Add link to manual page.
- backups: Fix styling for upload size warning.
- backups: Increase timeout for SSH operations to 30 seconds.
- letsencrypt: UI: Fix checkbox disabling.
- datetime: Switch from chrony to systemd-timesyncd.
- Updated translations for Czech, Norwegian Bokmål, and Spanish.

10.191 Version 0.49.0 (2019-02-05)

- security: Update javascript for Content Security Policy.
- help: Use correct package to determine available version.
- repro: Disable app due to issues with Debian package.
- ui: Fix regression with card icon style in front page.
- js: Support full librejs compatibility.
- js: Remove javascript license link from footer.
- backups: Remove incorrectly set buffer size during download.

- backups: Fix incomplete download archives.
- backups: Improve performance of backup download.
- radicale: Handle migration from 1.x to 2.x.
- datetime: Switch from ntp to chrony.
- backports: Add buster-backports to apt sources list.
- Updated translations for Czech, Norwegian Bokmål, and Hungarian.

10.192 Version 0.48.0 (2019-01-28)

- Updated translations for Czech, Hungarian, German, and Norwegian Bokmål.
- UI improvements:
  - Fix top margin for content containers.
  - Fix setting width of card-list at various page sizes.
  - Show help nav item text when navbar is collapsed.
  - Hide restart/shutdown items when navbar is collapsed.
  - Compact pages on extra small screen sizes.
- Backups improvements:
  - Add backup/restore support for syncthing and openvpn.
  - Upgrade apps before restoring them
  - Fix showing not-installed apps in create backup page
  - Automatically install required apps before restore.
  - Add a loader to the restore button to indicate progress.
- Serve default favicon for apps that don't provide one.
- radicale: Fix issue with configuration changes not applying.
- storage: Fix false error message in log when visiting home page.
- infinoted: Handle timeout issue when stopping daemon during setup.
- matrix-synapse: Fix startup error caused by bind_address setting.
- radicale: Avoid changes to conffile for radicale 2.x.
- help: Fix showing status logs when an error occurs.
- fail2ban: Enable bans for apache auth failures.
- mldonkey: Initial work on new module for the eDonkey network.
  - Not available yet, due to bug in package.

10.193 Version 0.47.0 (2019-01-14)

- Show Gujarati in the list of languages.
- Replace glyphicons with forkawesome icons.
- Snapshots:
  - Change configuration to avoid filling up disk.
  - Handle "Config in use" error.
  - Update descriptions and configuration options.
- Firewall: Fix issue with transition from iptables.
- Security: Switch to Argon2 password hash.
- Cockpit: Add link to manual page and update description.
- Radicale: Add initial support for radicale 2.x.
- Setup:
  - Handle showing setup page after app completes installation.
  - Optimize installation in-progress checks and refresh time.

10.194 Version 0.46.0 (2018-12-31)

- Updated translations for Czech, German, Spanish, Ukrainian, and Norwegian Bokmål.
- Use systemd journal for logging.
- Rename plinth binary package to "freedombox", and merge freedombox-setup package into it.

10.195 Version 0.45.0 (2018-12-17)

- Storage: Merge list of removable media into existing table.
- Backups: Allow remote backups to SSH servers using sshfs.
- Backups: Removed asking for backup archive name.
- Automatically handle future versions of PHP.
- Updated translations for Hungarian, Czech, Spanish, Chinese (Simplified), Italian, Norwegian Bokmål, French, and German.

10.196 Version 0.44.0 (2018-12-03)

- UI: Add card style and gray noise background to apps pages.
- UI: Fix distortion of the client apps buttons.
- ejabberd: Handle BOSH port change from TCP 5280 to 5443.
- Minetest: Update mods list to available Debian packages.
- Firewall: Use nftables instead of iptables.
- Snapshots: Fix default snapshot listing.
- Snapshots: Show description above either tab.
- Snapshots: Allow snapshots to be selected for deletion.
- Translations: Updated Czech, Norwegian Bokmål, Spanish, German, and Portuguese.

10.197 Version 0.43.0 (2018-11-19)

- Backups improvements:
  - Allow backups to be downloaded directly, without export step.
  - Restore directly from uploaded backup.
  - Avoid error for apps with no data to backup.
  - Show free disk space on upload and restore page.
  - Do not limit maximum upload size.
- openvpn: Migrate to easy-rsa 3 and fix setup issues.
- Make single sign-on tickets valid for 12 hours.
- Use consistent terminology for updates.
- Updated translations for Czech and Portuguese.

10.198 Version 0.42.0 (2018-11-05)

- Fix wrong color in mobile menu
- snapshot: Fix broken snapshot management after snapper update
- Enable backup/restore for tor, upgrades, monkeysphere, letsencrypt, tahoe
- monkeysphere: Handle importing new OpenSSH format keys
- udiskie: unmount drive as superuser
- Updated translations for Telugu, Indonesian, and Italian

10.199 Version 0.41.0 (2018-10-22)

- Enable backup/restore for datetime, deluge, avahi, backups, bind, security, snapshot, ssh, firewall, diagnostics, names, power, and storage.
- snapshot: Fix issue with setting configuration.
- backups: Fix backup archives ownership issue.
- backups: Fix issue with showing exports from disks without labels.
- backups: Don't rely on disk labels during export/restore.
- backups: Fix downloading extracted archive files.
- Updated translations for Norwegian Bokmål, French, Russian, and Spanish.

10.200 Version 0.40.0 (2018-10-08)

- Backups
  - Enable backup/restore for mumble, privoxy, roundcube, searx, jsxc, coquelicot, transmission, quassel, shadowsocks, sharing, pagekite, and cockpit.
  - Allow backup archives to be downloaded/uploaded through browser.
  - mediawiki: Backup/restore settings as well as data.
- User Interface

  - Change card text style and position.
  - Change maximum cards per row.
  - Add tint effect on card icons under "Apps".
- mediawiki: Run update script for 1.31 upgrade.
- customization: Show custom shortcuts on frontpage.
- Updated translations for Norwegian Bokmål, Portuguese, Spanish, Czech, German, French, and Italian.

10.201 Version 0.39.0 (2018-09-24)

- Updated translations for Hungarian and Norwegian Bokmål.
- Merge Removable Media (udiskie) into Storage module.
- Add Backups module for backing up apps data.

10.202 Version 0.38.0 (2018-09-10)

- mediawiki: Enable SVG support for MediaWiki
- upgrades: Clean up old kernel packages during automatic upgrades
- Make the progress bar at the top of the page more visible.
- Updated translations for Norwegian Bokmål, Czech, Russian, German, Hungarian, and Spanish.

10.203 Version 0.37.0 (2018-08-27)

- Updated translations for Czech, Norwegian Bokmål, Russian, Spanish, Hungarian, and Dutch.
- install: Use Post/Response/Get pattern for reloads.

10.204 Version 0.36.0 (2018-08-13)

- Updated translations for Hindi, Spanish, Russian, Telugu, German, Hungarian, Czech, and French
- ejabberd: Remove deprecated settings from already existing config files
- mediawiki: Fix issue with re-installation
- mediawiki: Enable Instant Commons
- mediawiki: Fix images throwing 403s
- turbolinks: Reload page using JavaScript
- Add Lato woff2 fonts
- Disable launch button for web client when not installed

10.205 Version 0.35.0 (2018-07-30)

- configuration: Add an option to set a default app for FreedomBox. The root URL path (`https://domainname/`) will redirect to the selected app.
- ejabberd: Remove deprecated `iqdisc` setting. To apply this fix, disable and then re-enable the Message Archive Management setting.
- ejabberd: Replace logo with original version.
- mediawiki: Enable short URLs, which look like `https://domainname/mediawiki/ArticleName`.
- radicale: Clarify description for shared calendar/addressbook.
- storage: Handle mount points with spaces.
- udiskie: Add button to eject drives.
- udiskie: Also show read-only filesystems.
- udiskie: Remove internal networks warning.
- udiskie: Show special message when no storage device available.
- Add turbolinks library for smoother navigation.
- Removed extra text from icons for mediawiki, radicale, and tahoe-lafs.
- Updated translations for Russian, Spanish, Dutch, Hungarian, Hindi, Italian, Telugu, German, and Norwegian Bokmål.

10.206 Version 0.34.0 (2018-07-16)

- Prompt for secret during firstboot welcome
  - (Does not apply to downloadable FreedomBox images, but only when installed using freedombox-setup package.)
- Updated translations for Italian, Dutch, Hindi, Hungarian

10.207 Version 0.33.1 (2018-07-04)

- Fix issue where editing a user would remove them from admin group
- Updated translations for Hungarian, Czech, Spanish, Russian, Hindi

10.208 Version 0.33.0 (2018-07-02)

- Updated translations for Hungarian, Norwegian Bokmål, Spanish, Russian, Czech, Hindi, Dutch, Italian
- firewall: Display information that a service is internal only
- users: Don't show Create User link to non-admin users
- users: Redirect to users list on successful user creation
- packages: Show button to refresh package lists when a package is not available for install
- Only show front page shortcuts that a user is allowed to access
- Restrict removal of last admin user
- Use logos instead of icons in the apps page
- udiskie: New module for automatic mounting of removable media

10.209 Version 0.32.0 (2018-06-18)

- Apply new card based design
- Fix client info table size and flickering
- first-setup: Automatically expand root partition
- mediawiki: Enable image uploads
- mediawiki: Make private mode and public registrations mutually exclusive
- mediawiki: Hide frontpage shortcut when private mode is enabled
- Updated translations for Norwegian Bokmål, Czech, Spanish, Russian, Hindi, Telugu, Italian, Dutch, German, and Hungarian

10.210 Version 0.31.0 (2018-06-04)

- Updated translations for Czech, Spanish, Russian, German, Italian, Hindi, Telugu, and Norwegian Bokmål
- mediawiki: Added private mode option
- users: Fix user permissions not being saved
- users: internationalize a string
- mediawiki: Run update script for 1.30 upgrade
- shortcuts: Fix urls for ikiwiki shortcuts

10.211 Version 0.30.0 (2018-05-21)

- Updated translations for Russian, Italian, Norwegian Bokmål, Hungarian, and Hindi
- setup: Remove unavailable as a state in setup_helper

10.212 Version 0.29.1 (2018-05-08)

- security: Fix issue with Plinth locked out from sudo
- Updated translations for Czech and Spanish

10.213 Version 0.29.0 (2018-05-07)

- security: Allow console login access to user plinth
- Add an option to enable/disable public registrations in mediawiki
- tt-rss: Skip the check for SELF_URL_PATH
- searx: Fix issue with uwsgi crashing
- Updated translations for Czech, Spanish, German, Norwegian Bokmål, and Italian

10.214 Version 0.28.0 (2018-04-23)

- setup: disable install button for currently unavailable apps
- Add locale for Lithuanian (lt)
- Translation updates for Italian, Czech, Russian, Spanish, German, Norwegian Bokmål, Telugu, and Dutch

10.215 Version 0.27.0 (2018-04-09)

- middleware: Skip 'installed' message for essential apps
- users: Fix admin group appearing twice in permissions
- apps: Fix app names and short descriptions not being translated
- snapshots: Move manual page link to the index page
- UI: Fix progress bar not appearing
- snapshots: Fix for permissions issue when updating configuration
- snapshots: Add option to enable/disable software installation snapshots
- Translation updates for Italian, Czech, Russian, Spanish, Dutch, German, Norwegian Bokmål, and Ukrainian

10.216 Version 0.26.0 (2018-03-26)

- snapshots: Update description
- searx: Rewrite url from /searx to /searx/
- manual: Link to manual from each service
- Workaround security issues in django-axes
- apache: Only regenerate snake oil cert when needed
- apache: Explicitly enable the latest version of PHP module
- apache: Increase module version number to fix php7.2
- Update translations for Chinese (Simplified), Russian, Czech, German, Norwegian Bokmål, Hungarian, Spanish, and Italian

10.217 Version 0.25.0 (2018-03-12)

- sharing: Add app for sharing disk folders.
- ttrss: Update list of client apps.
- infinoted: Allow setup to recover after timeout issue.
- snapshots: Add configuration tab with settings for time-based snapshots.

10.218 Plinth v0.24.0 (2018-02-26)

- Add file-sharing application Coquelicot.
- Add metasearch engine application Searx.
- Add locale for Hungarian (hu).
- mediawiki: Allow shortcut to be publicly visible on front page.
- clients: Add and correct Client Apps.
- locale: Preferred language can be set in each user's profile.
- locale: Anonymous users can select preferred language.
- config: Remove language selection from config page.
- matrixsynapse: Fix mail attribute for ldap login.

10.219 Plinth v0.23.0 (2018-02-12)

- snapshots: Modify configurations to reduce disk usage.
- snapshots: Skip currently active snapshot when deleting all snapshots.
- jsxc: Use consistent url format.
- sso: Increase timeout to 60 minutes.
- theme: Change font from Helvetica to Lato.
- Translation updates for Czech, German, Gujarati, and Telugu.

10.220 Plinth v0.22.0 (2018-01-30)

- matrix-synapse: Make sure configuration file does not get corrupted.
- tor: Show enabled status properly.
- first_setup: Fix not showing admin user creation step.
- Migrate from GitHub to Salsa
- Migrate from CircleCI to GitLab CI on Salsa.
- Translation updates for Czech, Dutch, Gujarati, Hindi, Russian and Telugu.
- Started new translation for Ukrainian.

10.221 Plinth v0.21.0 (2018-01-15)

- navigation bar: Change label from 'Configuration' to 'System'.
- storage: Removed beta warning for expanding partition.
- groups: Consistently show available user groups, even before applications are installed.
- syncthing: Restrict administration to users in "syncthing" group.
- help: Show menu on smaller screens also.
- diagnostics: Enable the "Run Diagnostics" button when applications are enabled but not running.

10.222 Plinth v0.20.0 (2018-01-01)

- bind: Don't use forwarders by default
- ejabberd: Remove redundant button Client Apps
- mediawiki: Add wiki application
- users: Make sure first run actually works
- bind: Add information about current utility

10.223 Plinth v0.19.0 (2017-12-18)

- ejabberd: Use dynamic reload instead of restart when changing configuration.
- manual: Make manual available as a PDF download.
- minetest: Show domain information for users to connect to minetest.
- snapshots: Add button to delete all snapshots.
- snapshots: Add option to enable/disable automatic timeline snapshots.
- users: Add groups for bit-torrent and feed-reader, available when these applications are installed.

10.224 Plinth v0.18.0 (2017-12-04)

- Add Shadowsocks client with socks5 proxy.
- Fix SSO regressions and conflict with captcha.
- transmission: Fix sso not being enabled on upgrade.
- avahi: Add service for FreedomBox discovery.
- Add client information for modules.

10.225 Plinth v0.17.0 (2017-11-20)

- transmission: Enable Single Sign On.
- cockpit: Add short description to frontpage shortcut.
- fail2ban: Fix spelling and sentence structure.

10.226 Plinth v0.16.0 (2017-11-06)

10.226.1 Added

- Add mobile, web and desktop client info for modules.
- Enable django SecurityMiddleware to improve security ratings.
- cockpit: New module for server administration and web terminal.

10.226.2 Fixed

- letsencrypt: Fix internal server error when obtaining a certificate.
- ejabberd: Fix LDAP server entry in config file during setup.
- jsxc: Fix outdated URLs for connecting to local ejabberd server.

10.227 Plinth v0.15.3 (2017-10-20)

10.227.1 Changed

- Rename Disks to Storage.
- Rename Snapshot to Storage Snapshots.
- tt-rss: Enable API access by default.
- Allow access to Plinth from outside the LAN.
- matrix-synapse: Disable public registration by default.
- power: Merge actions into the user dropdown.

10.227.2 Added

- Add locales for Kannada (kn) and for Bengali (bn).
- ejabberd: Use Let's Encrypt certificate, also across renewals.
- matrix-synapse: Add enable/disable public registrations.
- Add captcha validation on 3 failed attempts.
- matrix-synapse: Enable LDAP integration.
- letsencrypt: Automatically obtain and revoke SSL certificates.

10.227.3 Fixed

- Fix front page label names.
- Fix vertical alignment of shortcut icons.
- storage: Fix issue with locales that use other decimal separators.
- Make tt-rss api accessible using Apache basic auth.
- letsencrypt: Handle case where current domain is empty.
- Handle both admin and non-admin user names in update user template.

10.228 Plinth v0.15.2 (2017-09-24)

10.228.1 Added

- letsencrypt: Show more info on cert validity status.
- letsencrypt: Add option to delete certificates.
- letsencrypt: Add option to let Plinth manage certbot's renewal hooks.
- power: Warn if a package manager is running before shutdown/restart.
- security: Install and manage fail2ban.
- names: Include domain and services from dynamicdns.
- disks: Add low disk space warning to system and disks page.

- ssh: New application to manage SSH server.
- Add api module to get enabled services and access info.
- Add Django password validators.
- ejabberd, ikiwiki, ttrss: Add user login descriptions.

10.228.2 Removed

- diaspora: Disable for this release due to issues affecting package.
- Remove help from navbar before firstboot complete.

10.228.3 Fixed

- i18n: Don't use backslash-newline for wrapping long lines.
- radicale: Update link to documentation.
- sso: Upgrade crypto to 4096-bit RSA and SHA-512.
- Users: Allow non-admin users to log out.

10.228.4 Changed

- letsencrypt: Make Let's Encrypt an essential module.
- UI: Make apps and configure pages responsive on small screens.
- Make help accessible for logged-in non-admin users.

10.229 Plinth v0.15.0 (2017-07-01)

- Added Tahoe-LAFS module for distributed file storage.
- Added Diaspora* module for federated social networking.
  - Currently only available in "contrib" repository.
- New Locales for Czech (cs) and Tamil (ta).
- Added SSO using auth_pubtkt for Syncthing, TT-RSS, and the Repro admin panel.
  - If you are logged in to Plinth, you will be automatically logged in to these web apps.
- ejabberd: Added option to enable/disable Message Archive Management.
- help: Added Debian release name to about page.
- firstboot: De-bloat first welcome screen.
- Pinned footer to the bottom of the viewport.
- disks: Restrict precision of reported available space on root partition.
- diagnostics: Disable button if app/service is not running.
- help: Only show help pages if user is logged in.
- navbar: Moved logout to user drop-down and added a new power drop-down.

- disks: Show disabled partition resize option if no space is available.
- Added line break to titles to fix frontpage layout.
- syncthing: Fixed typos and clarity in description.
- firewall: Fix 500 error when firewalld is not running.
- setup: Disable install/upgrade when dpkg/apt is running.
- disks: Use information from lsblk for more accuracy.
- datetime: Show timezone properly when it not in expected list.

10.230 Plinth v0.14.0 (2017-04)

- tor: Added option to use upstream bridges.
- openvpn: Added shortcut to front page, shown only when logged-in.
- openvpn: Non-admin users can download their own profiles.
- Added new locales for Hindi (hi) and Gujarati (gu).
- Added Syncthing module for file synchronization.
- Added Matrix Synapse as chat server with groups, audio and video.
- Require admin access for all system configuration pages.
- Changed appearance of topbar and footer.
- openvpn: Regenerate user key or certificate if empty.
- disks: Workaround issue in parted during resize.

10.231 Plinth v0.13.1 (2017-01-22)

- Two new apps were added:
  - Gobby Server (infinoted) for collaborative editing of text documents
  - Domain Name Server (BIND), in system menu
- Added JavaScript license web labels to provide partial support for LibreJS.
- Added basic configuration form for Minetest server.
- Added indicator to Help->About page if new Plinth version is available.
- Show app logos on front page instead of generic icons.
- Prevent anonymous users from accessing setup pages.
- Split Chat Server (XMPP) app into Chat Server (ejabberd) and Chat Client (jsxc).

10.232 Plinth v0.12.0 (2016-12-08)

- Open up RTP ports in the firewall for repro (SIP server).
- Front page shortcuts for services show a Configure button in the details box for logged-in users.
- Add mods packages to be installed with Minetest server.
- Fix issue with reading Dynamic DNS status as non-root user.
- After the hostname is changed, ensure the domain name is still set correctly.
- Allow the domain name to be cleared, and properly set the configuration in this case.
- On the Certificates (Let's Encrypt) page, show a more informative message when no domains are configured.
- On the Chat Server (XMPP) page, show more clearly if domain is not set.
- Apps that require login will not be shown on the front page, unless the user is logged in.
- Show status block for News Feed Reader (Tiny Tiny RSS).
- Change appearance of front page with larger icons and repositioned text.
- Firewall page only lists services that have been setup. The port lists are collapsible under each service.
- Support configuring IPv6 networks.
- Make it less likely to accidentally delete the only Plinth user.
- Updated to work with JSXC 3.0.0 (XMPP web client).

10.233 Plinth v0.11.0 (2016-09-29)

- Added loading icon for additional busy operations.
- Added basic front page with shortcuts to web apps, and information about enabled services.
- networks: Add batctl as dependency, required for batman-adv mesh networking.
- users:
  - Fixed checking restricted usernames.
  - Display error message if unable to set SSH keys.
  - Flush nscd cache after user operations to avoid some types of errors.
- monkeysphere:
  - Adopted to using SHA256 fingerprints.
  - Sort items for consistent display.
  - Handle new uid format of gpg2.
  - Fixed handling of unavailable imported domains.
- minetest: Fixed showing status block and diagnostics.
- Fixed stretched favicon.
- Switched base template from container-fluid to container. This will narrow the content area for larger displays.
- Plinth is now able to run as "plinth" user instead of root user.
- xmpp: Replaced jwchat with jsxc.
- ikiwiki: Allow only alphanumerics in wiki/blog name to avoid invalid paths.

10.234 Plinth v0.10.0 (2016-08-21)

- Updated Plinth to support Django 1.10.
- Added a page to display recent status log from Plinth. It is accessible from the 500 error page.
- Tor: Added options to toggle relay and bridge relay modes.
- Radicale: Added access rights control.
- Ikiwiki: Updated suggested packages.
- Users and Groups: Fixed editing users without SSH keys.
- Networks: Added basic support for configuring batman-adv mesh networking.
- Networks: Fixed incorrect access for retrieving DNS entries.
- New languages:
  - Persian (50% translated)
  - Indonesian (not started, contributions needed)
- New modules added to Plinth:
  - Disks: Shows free space of mounted partitions, and allows expanding the root partition.
  - Security: Controls login restrictions.
  - Snapshots: Manages Btrfs snapshots.

10.235 Version 0.9.4 (2016-06-24)

- Added Polish translation.
- Fixed issue preventing access to Plinth on a non-standard port.
- Dealt with ownCloud removal from Debian. The ownCloud page in Plinth will be hidden if it has not been setup. Otherwise, a warning is shown.
- Fixed issue in Privoxy configuration. Two overlapping listen-addresses were configured, which prevented privoxy service from starting.
- Fixed issue that could allow someone to start a module setup process without being logged in to Plinth.
- Fixed issues with some diagnostic tests that would show false positive results.
- Added check to Diagnostics to skip tests for modules that have not been setup.
- Fixed some username checks that could cause errors when editing the user.
- Added sorting of menu items per locale.
- Moved Dynamic DNS and Pagekite from Applications to System Configuration.
- Allowed setting IP for shared network connections.
- Switched Dreamplug image from "non-free" to "free". This means that we no longer include the non-free firmware for the built-in wifi on Dreamplug.
- Added the "userdir" module for the Apache web server. This allows users in the "admin" group to create a folder called "public_html" under their home folder, and to publicly share files placed in this folder.
- New wiki and manual content licence: *Creative Commons Attribution-ShareAlike 4.0 International* (from June 13rd 2016).
- Switched to using apt-get for module setup in Plinth. This fixes several issues that were seen during package installs.

10.236 Version 0.9 (2016-04-24)

- Fixed Wi-Fi AP setup.
- Prevent lockout of users in 'sudo' group after setup is complete.
- Improved setup mechanism for Plinth modules. Allows users to see what a module is useful for, before doing the setup and package install. Also allows essential modules to be setup by default during FreedomBox install.
- Added HTTPS certificates to Monkeysphere page. Reorganized so that multiple domains can be added to a key.
- Added Radicale, a CalDAV and CardDAV server.
- Added Minetest Server, a multiplayer infinite-world block sandbox.
- Added Tiny Tiny RSS, a news feed reader.

10.237 Version 0.8 (2016-02-20)

- Added Quassel, an IRC client that stays connected to IRC networks and can synchronize multiple frontends.
- Improved first boot user interface.
- Fixed Transmission RPC whitelist issue.
- Added translations for Turkish, Chinese, and Russian. Fixed and updated translations in other languages.
- Added Monkeysphere, which uses PGP web of trust for SSH host key verification.
- Added Let's Encrypt, to obtain certificates for domains, so that browser certificate warnings can be avoided.
- Added repro, a SIP server for audio and video calls.
- Allow users to set their SSH public keys, so they can login over SSH without a password.

10.238 Version 0.7 (2015-12-13)

- Translations! Full translations of the interface in Danish, Dutch, French, German and Norwegian Bokmål, and partial Telugu.
- Support for OLinuXino A20 MICRO and LIME2
- New Plinth applications: OpenVPN, reStore
- Improved first-boot experience
- Many bugfixes and cleanups

10.239 Version 0.6 (2015-10-31)

- New supported hardware target: Raspberry Pi 2
- New modules in Plinth:
  - Shaarli: Web application to manage and share bookmarks
  - Date & Time: Configure time zone and NTP service
  - Service Discovery: Configure Avahi service
- Documentation revamp including new user manual and developer guide
- Improved diagnostic tests, available in Plinth
- Avoid unnecessary changes when installing on existing Debian system
- Network configuration supports PPPoE connections
- Debian packages can be download over Tor

10.240 Version 0.5 (2015-08-07)

- New targets: CubieTruck, i386, amd64
- New apps in Plinth: Transmission, Dynamic DNS, Mumble, ikiwiki, Deluge, Roundcube, Privoxy
- NetworkManager handles network configuration and can be manipulated through Plinth.
- Software Upgrades (unattended-upgrades) module can upgrade the system, and enable automatic upgrades.
- Plinth is now capable of installing ejabberd, jwchat, and privoxy, so they are not included in image but can be installed when needed.
- User authentication through LDAP for SSH, XMPP (ejabberd), and ikiwiki.
- Unit test suite is automatically run on Plinth upstream. This helps us catch at least some code errors before they are discovered by users!
- New, simpler look for Plinth.
- Performance improvements for Plinth.

10.241 Version 0.3 (2015-01-20)

- Tor Bridges: All boxes now act as non-exit Tor bridges, routing traffic for the Tor network.
- Firewall: firewall is on by default and is automatically managed.
- Add BeagleBone support. We now have images for BeagleBone, RaspberryPi, VirtualBox i386/amd64, and DreamPlug.
- Ability to enable and use Tor Hidden Services. Works with Ejabberd/JWChat and ownCloud services.
- Enable Tor obfsproxy with scramblesuit.
- Drop well-known root password (an account with sudo capabilities still exists for now but will be removed soon).
- Switch to unstable as suite of choice for easier development.
- Newer images are built with systemd by default (due to Debian change).
- Install and operate firewall automatically (uses firewalld).
- Major restructuring of Plinth UI using Python3, Django web development framework and Bootstrap3. Code quality is much better and UI is more polished.
- Introduced packaging framework in Plinth UI for on-demand application installation.

10.242 Version 0.2 (2014-03-16)

- Support for Raspberry Pi and VirtualBox (x86) in addition to the DreamPlug.
- New Services:
  - Configuration Management UI.
  - Instant Messaging.
  - OwnCloud.
  - dnsmasq.
  - Low-Level Configuration Management.
  - Service Announcement.
  - LDAP Server.
  - LXC Support.
  - Source Packages.
- The privoxy setup is now the default from Debian.

10.243 Version 0.1 (2013-02-26)

- First FreedomBox software release (0.1 image, developer release).
- Full hardware support in Debian
- Support for DreamPlug.
- Basic software tools selected as common working environment:
  - User interface system "plinth"
  - Cryptography tools: gpg or "monkeysphere"
  - Box-to-box communication design: Freedom-buddy (uses TOR network)
  - Web cleaning: "privoxy-freedombox".

11 Customizing

Though FreedomBox's philosophy is to have the user make as few decisions as possible about the FreedomBox itself, a few options for customization have been provided to facilitate some advanced use cases.

11.1 Change Default App

Available since version: 0.36.0

Skill level: Basic

Use Case: A FreedomBox that primarily runs only one public-facing application whose web application is set as the landing page when someone visits the domain name of the FreedomBox over the internet.

e.g. A university using MediaWiki running on FreedomBox as a course wiki wants its students typing in the domain name into their browser to directly go to the wiki bypassing the FreedomBox home page.

Configuration: Change the Default App in the configure page to whichever app you want to be served as default.

11.2 Custom Shortcuts

Available since version: 0.40.0

Skill level: Advanced

Use Case: The administrator of a community deployment of FreedomBox manually installs a few additional unsupported applications on the FreedomBox and wants users to be able to transparently access them through the web and mobile applications of FreedomBox.

Note: This feature is meant to be used with applications that are end-user facing, i.e have a web or mobile client.

Configuration:

FreedomBox reads custom shortcuts you have created from various files paths on the server. A shortcuts file can be any file with `.json` extension in the following directories: `/etc/freedombox/custom-shortcuts.json.d/`, `/var/lib/freedombox` and `/usr/share/freedombox/custom-shortcuts.json.d/`. Or they can be a file named `custom-shortcuts.json` in these directories: `/etc/freedombox/`, `/etc/plinth/` (only for compatibility), `/var/lib/freedombox/` and `/usr/share/freedombox/`. The file should follow the same JSON schema as the FreedomBox API. You can refer to the JSON schema by visiting `https://<my-freedombox-url>/plinth/api/1/shortcuts`.

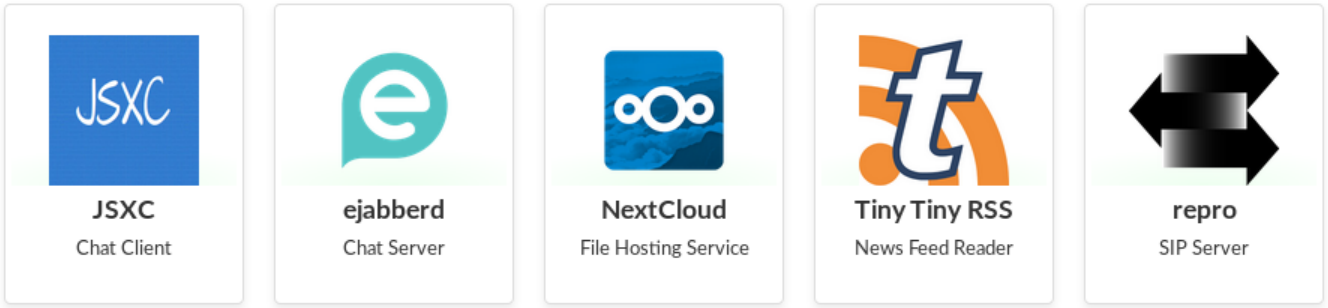
As an example, to add one additional shortcut for NextCloud app, create a file called `/etc/freedombox/custom-shortcuts.js` (create directories as needed).

```
{
    "shortcuts": [{
        "name": "NextCloud",
        "description": ["Nextcloud is a suite of client-server software for creating and ↩
            using file hosting services."],
        "icon_url": "/plinth/custom/static/icons/nextcloud.png",
        "clients": [{
            "name": "nextcloud",
            "platforms": [{
                "type": "web",
                "url": "/nextcloud"
            }]
        }],
        "tags" : ["Groupware", "File sync"]
    }]
}
```

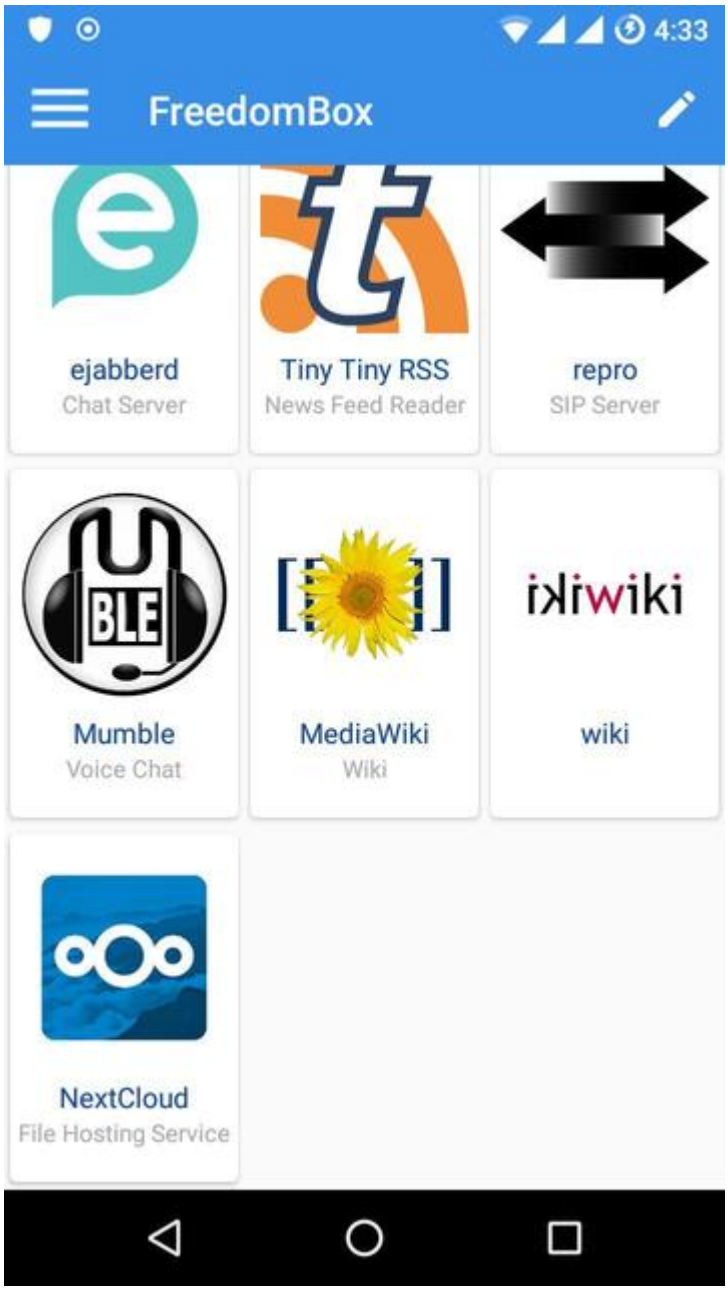
The corresponding icons for the shortcuts listed in the above file should be placed in the directory `/var/www/plinth/custom/stat`
The file names of the icons should match with those provided in `/etc/freedombox/custom-shortcuts.json.d/nextclou`

After adding an entry for NextCloud in `nextcloud.json` and an icon, restart Plinth by executing the command `systemctl restart plinth` on the FreedomBox. You can also restart the FreedomBox from the web interface.

After restart the Plinth home page will display an additional shortcut for NextCloud as shown below:



The same shortcut will also be displayed in any Android apps connected to the FreedomBox.



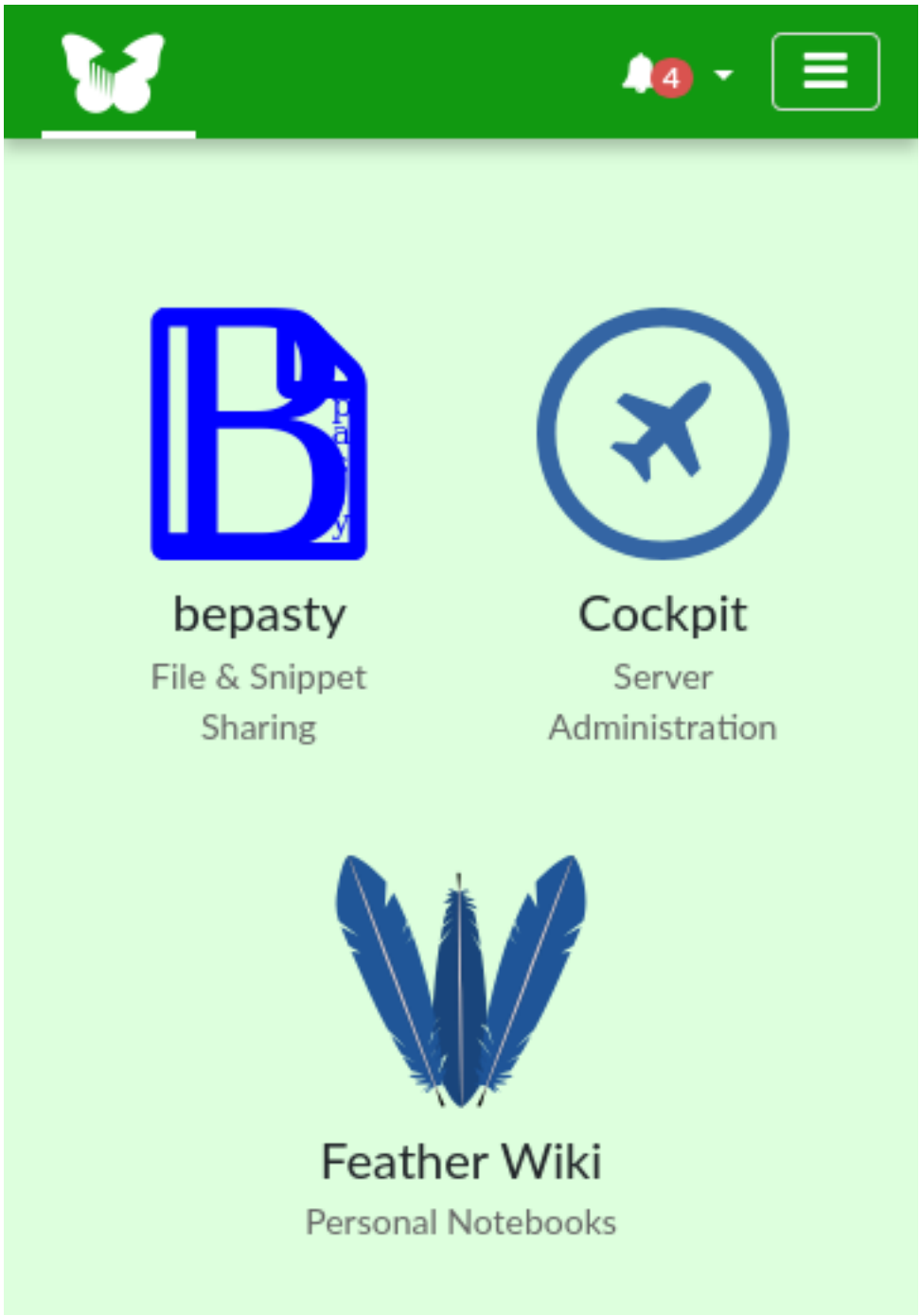
11.3 Custom Styling

Available since version: 24.25

Skill level: Advanced

Use Case: The administrator of a community or home deployment of FreedomBox wants to customize the web page styling of FreedomBox.

Configuration: Create a file in the path `/var/www/plinth/custom/static/css/user.css` and write CSS styling rules. This file has the highest priority as per the cascading rules. Use the web browser's developer console to understand which rules to override and how much specificity is needed.



12 Contributing

12.1 Get Involved

From code, design and translation to spreading the word and donation, here are a number of ways to contribute to FreedomBox.

12.1.1 Quick Links

- Our vision explains what we want to achieve.

- FreedomBox Developer Manual
- Progress calls
- TODO page
- Donation page

12.1.2 Welcome to newcomers

As a new contributor, you are more than welcome to introduce yourself to others on the FreedomBox discussion forum, mailing list or on the #freedombox IRC channel. In addition to make useful contacts, you can start reporting bugs and translate (see below) the wiki website and the FreedomBox web interface.

12.2 Contributions needed

Sorted by ascending difficulty

12.2.1 Donate

Obviously, donating is the easiest way to contribute.

You can help the project financially by donating via PayPal, Bitcoin or by mailing a check. Please see the donation page for details on how to donate.

FreedomBox project is run by volunteers. The FreedomBox Foundation is a 501(c)(3) federal nonprofit corporation with recognition from the IRS.

12.2.2 Spread the Word

Spreading the word about FreedomBox only requires to know about the basic goals of the project and an overall description of the product.

Speak to your family, friends, local community or at global conferences about the importance of FreedomBox. To be a successful project we need much more awareness and many more participants, be it users or contributors. Write about your efforts on the wiki.

If you are into marketing, you can get in touch with the marketing team through the marketing section of our forum.

12.2.3 Feed Us Back (Comment)

After some time, we lose the fresh view of newcomers. Just providing your feedback helps us a lot.

Browse our websites and documentation and play with FreedomBox and comment your impressions on the feedback section of our forum.

12.2.4 Request applications

Check our applications wishlist and help us extend it by searching the web for other interesting free software to include in FreedomBox.

12.2.5 Translate

All text visible to users of FreedomBox needs to be localized to various languages. If you know english and speak another language you can contribute translating. Translating is a nice way to get familiar with the project while contributing.

This translation work includes:

- Web Interface of FreedomBox
- FreedomBox documentation, including the FreedomBox wiki
- FreedomBox website and foundation website.
- Django web framework that FreedomBox uses.
- Individual applications that FreedomBox exposes to users.

The localization of FreedomBox web interface happens mainly on the web-based tool at Weblate.

If you wish to see FreedomBox available for one of your languages, please start a discussion on the FreedomBox discussion forum's development category to work with others translating for that language.

For more information, please visit the FreedomBox translators page.

12.2.6 Document: User Manual, Website and Wiki, HowTo/demo videos

FreedomBox needs better documentation for users and contributors. Sometimes, just rewording or presenting the information another way, already helps. Others, a certain knowledge is needed.

FreedomBox manual is prepared by aggregating various pages on the wiki and exporting to various formats. The manual is then used in FreedomBox Service and elsewhere.

If you wish to contribute to the FreedomBox wiki (and consequently the FreedomBox manual), you can create a wiki account and start editing.

For contributing to the website please start a discussion on the FreedomBox discussion forum's development category.

Another way of documenting FreedomBox is to record and publish screencasts showing how to use it, or videos on why to use it. Like these or these.

12.2.7 Assure Quality (Test and Check)

- FreedomBox already runs on many platforms and it is not possible for developers to test all possible platforms. If you have one of the supported hardware you can help with testing FreedomBox on the platform.
- When an application is made available on FreedomBox, not all of its functionality is tested in the real world by developer doing the work. Deploying the application and testing it will help ensure high quality applications in FreedomBox.

See the quality assurance page for a basic list of test cases to check for and information on reporting bugs.

12.2.8 Code

If you are a developer, you can contribute code to one of the sub-projects of FreedomBox. Step-by-step process of contributing code to FreedomBox is available.

- FreedomBox Service: a web interface to administer the functions of FreedomBox.
- Freedom Maker: a script to build FreedomBox disk images for use on various hardware devices or virtual machines.

You can pick up a task from one of the TODO lists. The individual page project pages contain information availabily of the code, how to build and TODO lists.

12.2.8.1 Fix Bugs

List of bugs, feature requests and improvements are tracked on the FreedomBox issue tracker. In addition to that, see list of bugs to help out the Debian package we depend on. Also see the FreedomBox packaging team's dashboard for status of various packages that we use.

12.2.8.2 Add an Application

If you are a developer and wish to see an application available in FreedomBox, you can contribute by adding the application to FreedomBox. See the FreedomBox Developer Manual. Many applications that can be added to FreedomBox have been identified on the leaving the cloud page.

12.2.8.3 Development priorities

Upcoming priorities are discussed on an regular basis. You find the progress of the FreedomBox Service with its priorities here: issues board and milestones.

Please check next progress calls to keep yourself on track and meet members of the release team. A TODO page aggregates the complete list of the items to work on for FreedomBox.

12.2.9 Design

12.2.9.1 User Experience Design

If you are a user experience designer, you can help FreedomBox with the following items:

- UI experience for the FreedomBox Service web interface
- Web design for freedombox.org, freedomboxfoundation.org and the wiki pages
- Logo and branding (we currently have an identity manual and logos)
- Possible designs for custom FreedomBox cases on single board computers
- User experience design

12.2.9.2 Technical Design

FreedomBox needs your technical expertise to devise implementation plans for upcoming features. You can contribute to the discussion on various technical design and implementation aspects of FreedomBox. See FreedomBox discussion forum's development category.

12.2.10 Package Applications

FreedomBox is a Debian Pure Blend. In order to add applications to FreedomBox we need applications first to be Debian-packaged. Check our applications wishlist.

13 Developer Guide

The FreedomBox Developer Manual provides a step by step tutorial for writing apps for FreedomBox and an API reference. It is available from docs.freedombox.org.

14 Hacking

FreedomBox consists of two main projects:

- FreedomBox Service (Plinth), the web interface
- Freedom Maker, a script to build disk images for various hardware

14.1 FreedomBox Service (Plinth)

FreedomBox Service (Plinth) is a web interface to administer the functions of the FreedomBox.

FreedomBox Service is Free Software under GNU Affero General Public License version 3 or (at your option) a later version.

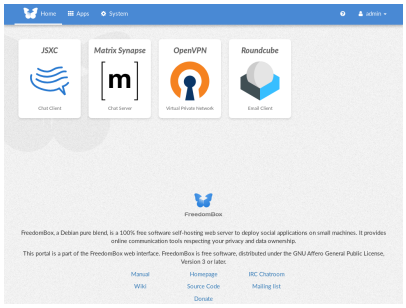
14.1.1 Using

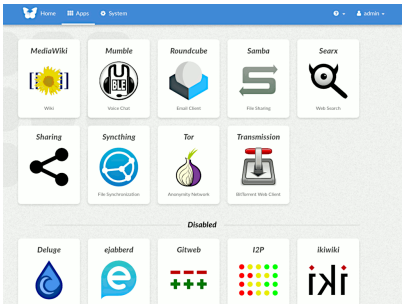
- FreedomBox Service comes installed with all FreedomBox images. You can download FreedomBox images and run on any of the supported hardware. Then, to access FreedomBox interface see quick start instructions.
- If you are on a Debian box, you may install FreedomBox Service from Debian package archive. Currently, only bullseye (stable), bookworm (testing), and sid (unstable) are supported. To install FreedomBox Service run:

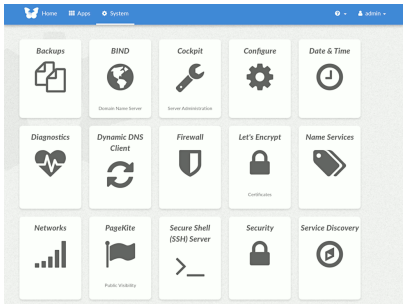
```
$ sudo apt-get install freedombox
```

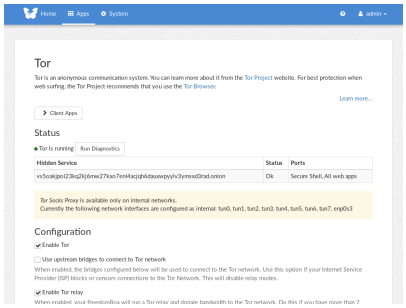
- You can also get FreedomBox Service from its Git repository and install from source.

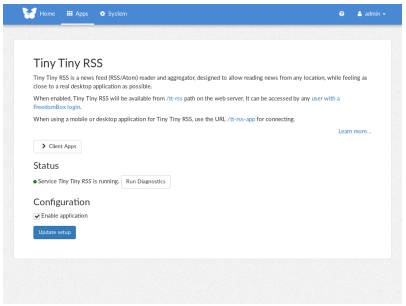
14.1.2 Screenshots

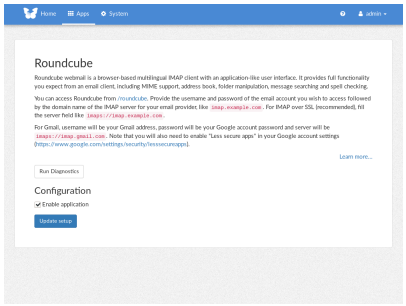


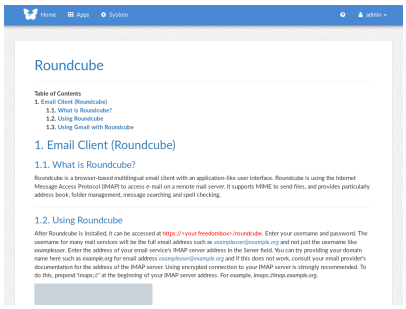


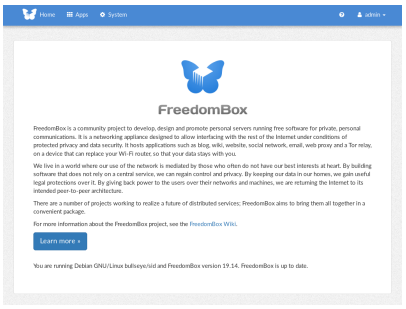












14.1.3 Support

You may ask for support on

- The discussion forum
- The mailing list
- #freedombox IRC channel
- FreedomBox Matrix channel

14.1.4 Contributing

We are looking for help to improve FreedomBox Service. You can contribute to FreedomBox Service by not just by coding but also by translating, documenting, designing, packaging and providing support.

- Instructions on how to contribute code are available.
- The primary Git repository is hosted at FreedomBox Salsa Page.
- Instructions for installing from source and hacking the source are available.
- List of bugs, TODO items and feature requests are available on the issue tracker.
- Before contributing to FreedomBox Service code, you need understand Python and Django on top which it is built.
- You can request for development assistance on the discussion forum, the mailing list or the #freedombox IRC channel.

14.1.4.1 Debian Package

- FreedomBox Service is packaged for Debian. FreedomBox Service is a native package and packaging source code is part of the main package source code.
- Issues related to packaging are listed on Debian BTS.

14.2 Freedom Maker

Freedom Maker is a script to build FreedomBox disk images for use on various hardware devices or virtual machines.

Freedom Maker can currently build FreedomBox disk images for the following:

- A20-OlinuXino-LIME
- A20-OlinuXino-LIME2
- A20-OLinuXino-MICRO
- Banana Pro
- BeagleBone
- Cubieboard2
- Cubietruck
- pcDuino3
- Raspberry Pi 2
- Raspberry Pi 3 Model B

- Raspberry Pi 3 Model B+
- VirtualBox
- QEMU
- AMD64 (x86-64) Machines, X86 Machines and other virtual machines (using raw disk images)

If a hardware platform is capable of running Debian, it should not be too much effort adopt Freedom Maker to create FreedomBox images for the platform.

Freedom Maker is Free Software licensed under GNU General Public License version 3 or (at your option) a later version.

14.2.1 Building FreedomBox Images

- You can get Freedom Maker from its Git repository and follow the instructions in the README to build a FreedomBox image.

14.2.2 Support

You may ask for support on

- The discussion forum
- The mailing list
- #freedombox IRC channel
- FreedomBox Matrix channel

14.2.3 Contributing

We are looking for help to improve Freedom Maker.

- Instructions on how to contribute code are available.
- Freedom Maker is hosted at FreedomBox Salsa Project. The primary Git repository is hosted there.
- You can contribute to FreedomBox by adding support for more hardware platforms. Freedom Maker can be easily adopted to newer platforms if they already support running Debian.
- You can create and test images with Freedom Maker regularly to test for new features and check for regressions.
- List of bugs, TODO items and feature requests are available on the issue tracker.
- You can request for development assistance on the discussion forum, the mailing list or the #freedombox IRC channel.

15 Tell people around you

- FreedomBox
- FreedomBox in the Press
- Conferences
- Talks and presentations
  - Available Material Slides and other raw material
- Facebook
- Twitter
- Mastodon
- Debconf11 Videos