Documentation

Technical information about CrossBox can be found here. Feel free to browse around to learn more about installing and running CrossBox on your server(s).

- CrossBox Server
  - Requirements
  - Quick Installation
  - Options in Details
  - Admin Area
  - Command Line Tools
  - Performance Tuning
  - Uninstall
  - Migration
  - Post machine migration
- Control Panels
  - cPanel
  - Plesk
  - Direct Admin
  - Other Control Panels
  - Docker
- Clusters
  - Creating a CrossBox Cluster
- Third Party Apps
  - Outlook
  - Thunderbird
  - Other Apps
- Building Android and iOS apps
  - Online Apps Builder
  - Android
  - iOS
- App Connector
  - App Connector Documentation

CrossBox Server

CrossBox Server

Requirements

Intro

Even though CrossBox is highly portable, make sure that your server satisfies the listed server requirements.

Supported Operating

Systems

CrossBox Server runs on **any x86-64 (64 bit) Linux** based operating system that has **glibc version 2.17 and up**.

Listed below are minimum required distribution versions. Versions **equal to or greater than** the ones listed below are fully supported:

- RedHat/CentOS/CloudLinux 7
- Ubuntu - 16.04 LTS xenial
- Debian - 9 stretch
- Fedora - 30
- OpenSUSE - 15.1
- Arch Linux
- Slackware - 14.2
- Gentoo
- Mint - 18.3 sylvia

You can check you glibc version by running

> ldd --version

via the command line interface.

Hardware Requirements

CrossBox Server can run on a small virtual machine, a big dedicated server, or anything in between.

CPU and memory requirements vary depending on the version you'll be running, the number of enabled features, and the number of concurrent users.

Standalone Version

The standalone version comes with everything prepackaged and preconfigured. It is intended for installing and running on a fresh, clean server.

Bare-minimum (use only core features):

- **1 CPU**
- **512 MB** of RAM
- **5GB** of Disk Space

Following features require additional memory when enabled:

- Anti-Spam - **250 MB** of RAM
- Anti-Virus - **600 MB** of RAM
- TURN/STUN - **50 MB** of RAM
- Document Converter (CDC) - **100 MB** of RAM
- TrustedLink (email assets SSL proxy) - **50 MB** of RAM

To run a standalone version with all features enabled, we recommend you install CrossBox on a server with at least 1 CPU and 2GB of RAM.

Control Panel Version

It's common for servers that utilize a control panel to have many of the needed services like MTA, mail server, anti-spam service, and anti-virus already running and in place.

Therefore, when installing on a server that's running cPanel, DirectAdmin, or Plesk, we auto-detect and use the following services automatically:

- MySQL
- Exim
- Dovecot
- MailScanner
- ClamAV
- CSF
- Firewalld
- UFW
- Solr

This effectively reduces the total memory needed for CrossBox to run to just 512 MB of RAM, when used alongside the control panel.

Supported Control Panels

cPanel, Plesk, and DirectAdmin currently have deep-integrations available.

However, you can still use CrossBox with any control panel due to its plug and play nature, but unlike with the supported control panels, some functions are not automated and may require manual action.

Hardware Calculation

If you want to calculate your CPU and RAM resources in advance, here are the numbers:

1CPU, 2GB RAM for 100 concurrent users.

For example, if your plan is to host 100 concurrent users, you'll need:

- 1 CPU
- 2GB RAM

If your plan is to host 200 concurrent users:

- 2+ CPU
- 4GB+ RAM

Software Dependencies

You don't need to install any new software dependencies for CrossBox to work. It comes pre-compiled and runs isolated from the rest of the system.

It won't ask you to install anything via your system's package manager other than core packages like locale package etc. We do not interfere with other software/packages/libraries you may already have running on the server.

Firewall

Most of the time firewall will be configured with what's installed on the machine. However, this may not be the case with custom firewall configurations. We only support CSF, UWF, Firewalld. If CrossBox is not accessible, but **crossbox status** says all services are running, the most common reason for limited connectivity is your firewall.

When installing **panel version** of CrossBox, these ports must be free:

100 - TCP IN/OUT CrossBox default port
80 - TCP IN/OUT
443 - TCP IN/OUT
3478 - TCP and UDP IN/OUT TURN
5349 - TCP and UDP IN/OUT TURN Secure
49152:65535 - UDP IN/OUT Port range for TURN

When installing **standalone version** of CrossBox, these ports must be free:

25 - TCP IN/OUT SMTP Local Delivery
80 - TCP IN/OUT CrossBox HTTP
100 - TCP IN/OUT CrossBox default port
110 - TCP IN/OUT POP3
143 - TCP IN/OUT IMAP
443 - TCP IN/OUT CrossBox HTTPS
465 - TCP IN/OUT SMTP Secure
587 - TCP IN/OUT SMTP
993 - TCP IN/OUT IMAP Secure
995 - TCP IN/OUT POP3 Secure
3478 - TCP and UDP IN/OUT TURN
5349 - TCP and UDP IN/OUT TURN Secure
49152:65535 - UDP IN/OUT Port range for TURN

If any of these ports are taken, the installer will likely fail. Make sure you disable/remove all services that bind to these ports.

Or if otherwise configured, as CrossBox allows custom ports in the configuration settings.

Master and Slave servers

Entire cluster must be able to communicate with each other trough 80/443 and default 100 port (unless configured differently). If Master server cannot connect to slave servers, login and other services will not work.

Other Requirements

Hostname

Make sure that before installing CrossBox your server has a resolvable hostname.

PTR (Reverse DNS)

We recommend that the server has a valid PTR (Reverse DNS) record that matches the server's hostname. A valid PTR record is important for good e-mail delivery.

Most VPS/Dedicated server providers allow you to edit a PTR record via the control panel they provide.

Root Access

A root user account and SSH access are required to install a CrossBox Server.

Web Terminal with root user access can also be used (for example WHM Terminal).

Home Directory

Your system needs to have a /home directory. If it doesn't exist, CrossBox Installer will automatically create it.

Let's start

> Are you ready to get started?
> Continue to Quick Installation Guide and have CrossBox up and running in a couple of minutes.

CrossBox Server

Quick Installation

Intro

First, please make sure that you checked out the Requirements before you start with the installation.

To install CrossBox you need to have a valid and active license. If you don't have one, you can purchase it here.

Here we will show you how to install CrossBox on a **single server**. If you plan to install CrossBox on multiple servers, please head out to our CrossBox Cluster documentation to learn how you can unify communication across different servers and configurations.

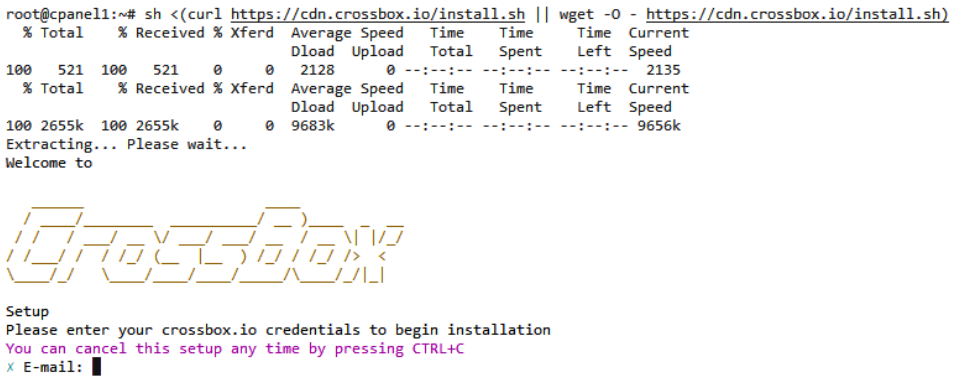
Installation

Run the Command

SSH into your server and run the following command as *root:*

sh <(curl https://crossbox.io/install.sh || wget -O - https://crossbox.io/install.sh)

Then proceed to log in with your CrossBox.io account:



After a successful login, you'll find the URL that you can now use to start the installation via your web browser.

The URL will be similar to the one shown below, but unique to your installation instance:

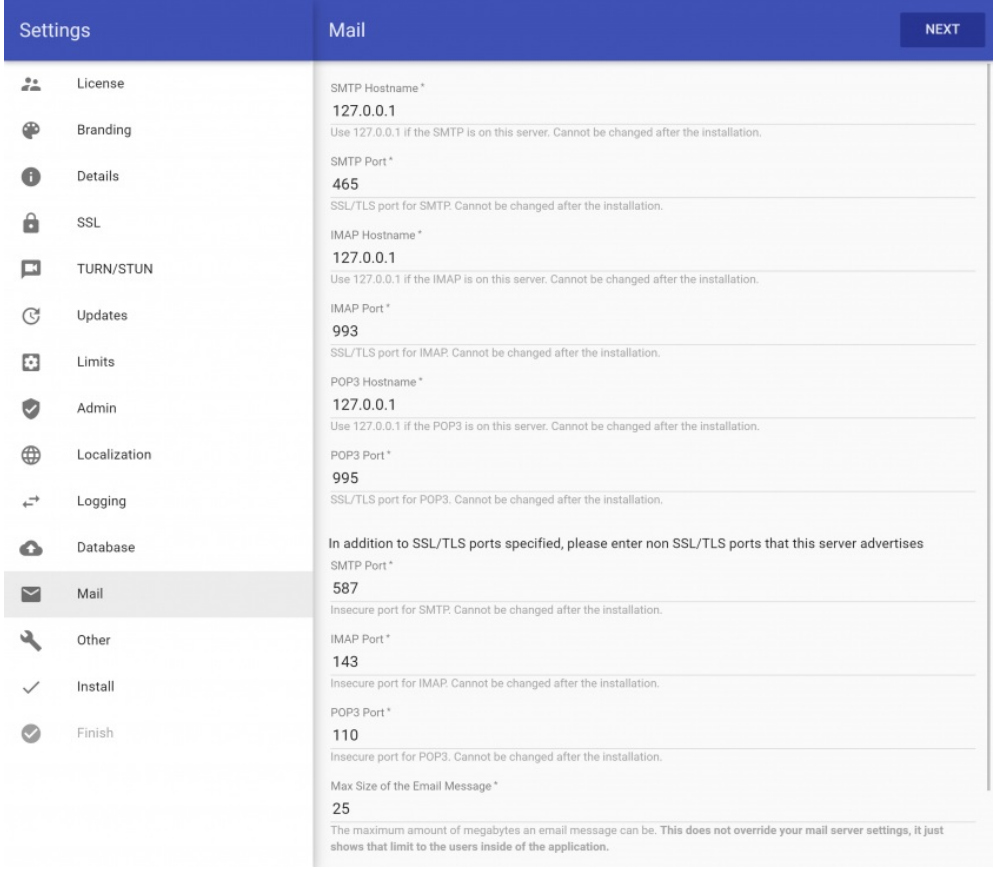


Access the Installation via the Web Browser

Open your favorite web browser and paste the URL that you copied.

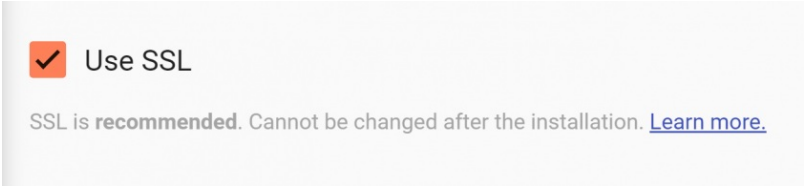
Configure

Almost everything is already pre-populated for you and the entire process takes about ten minutes or often less, depending on your server's performance.



Detailed Help

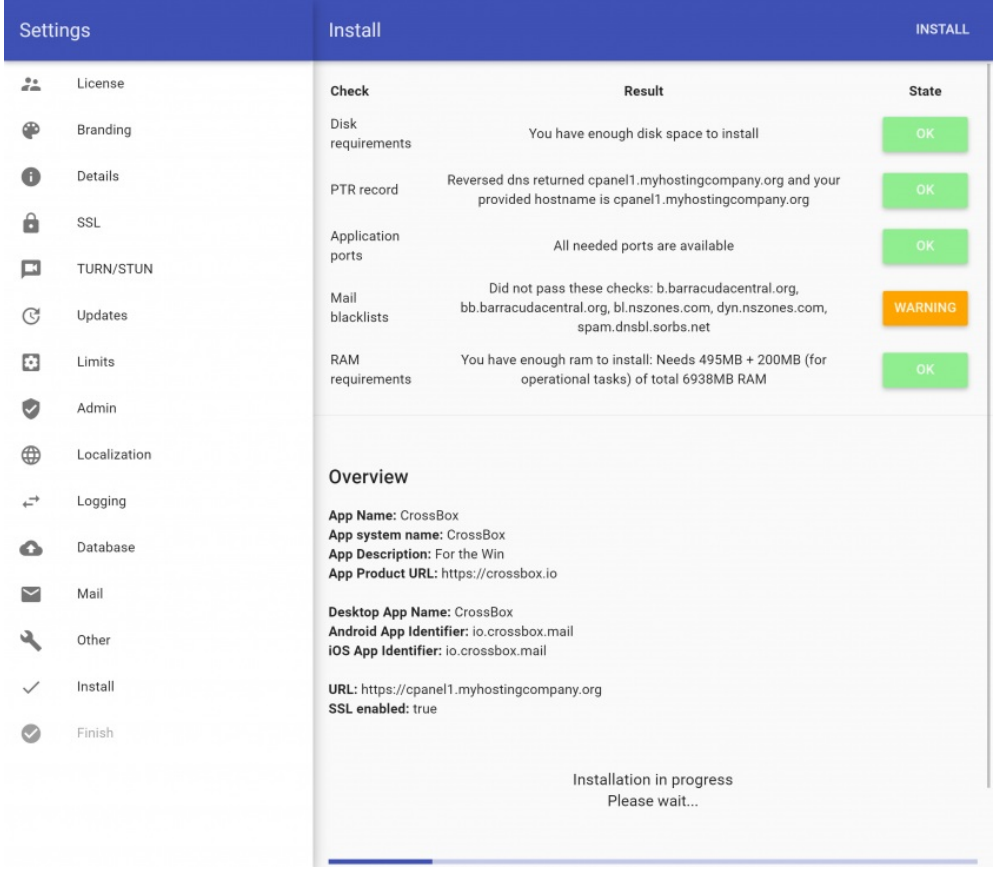
If you're not sure what to enter in a certain field during the installation, just click on the "**Learn more**" link which is displayed beneath it:



By clicking it, you'll be taken directly to that option's more detailed documentation.

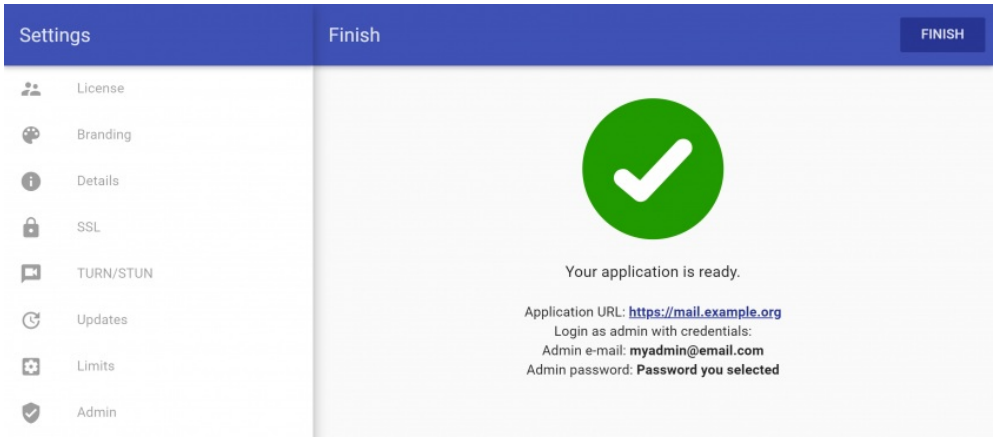
Start the Installation

After you click the "**Install**" button, CrossBox will start setting everything up. You can track the progress in real-time as shown below:



Finish the Installation

You can now navigate to the application URL and log in with your admin email address and password.



CrossBox Server

Options in Details

Server

Tweaking CrossBox Server to fit your needs is as easy as enabling and disabling features with a simple checkbox click. CrossBox will auto-detect if some kind of a control panel is already installed on the server and will auto-adjust itself. Here we have listed all the options and associated help descriptions.

Hostname

CrossBox will try to automatically resolve a hostname of the server it's being installed onto.

However, make sure that before installing CrossBox you have a valid PTR (Reverse DNS) record that matches your hostname and that your hostname can be successfully resolved.

Public IP Address

CrossBox will try to automatically resolve an IP address of the server it's being installed onto.

App Port

Enter a Port number that CrossBox will bind to and listen on.

If you're installing master server this option is disabled and ports 80/443 are enforced. Otherwise, you can specify a custom port.

Please make sure that App port(s) are not used by some other software running on the server.

If you have a firewall in place, please make sure that binding to these ports is allowed.

App Port cannot be changed after the installation.

SSL

We strongly recommend that you enable the SSL feature. Not only it will help protect data being sent over the network from prying eyes, but it is also needed for audio and video calls to work. It is very easy to set up and you have four options to choose from:

- Generate SSL certificates automatically (by using Let's Encrypt Certbot)
- Select from an auto-detected list (will scan the system for existing certificates) - **this is visible if the installer detects SSL certificates**
- Enter .crt/pem .key paths (lets you enter paths to certificate files)
- Upload .crt/pem and .key files (lets you upload certificate files)

TURN/STUN Server

CrossBox uses Coturn as a fast and reliable TURN/STUN server. It is responsible for streaming audio and video data during the CrossBox audio/video conference and for relaying media traffic when persons attending the call can't reach each other directly via the peer-to-peer protocol. This is most often the case when the person is behind a corporate firewall.

Overall, enabling a TURN/STUN server will greatly improve the quality of audio and video calls. The CrossBox Installer will give you the ability to choose if you wish to enable TURN/STUN integration and will also allow you to decide where it should be hosted.

You can choose to install it on the same server as CrossBox or you can enter an external server to offload audio and video streaming to a separate machine in the case you have thousands of concurrent calls being made.

If you plan on having more than 50 people streaming audio and video at the same time, please make sure that you allocate enough CPU/Network resources for your TURN/STUN server. The other approach would be to offload media streaming to a Coturn server hosted on another machine, by choosing "Use external coturn server" and filling in *Host* and *SSL Host* fields.

Coturn Non-SSL Port

Enter a non-SSL port that Coturn will use. The default Coturn non-SSL port is 3478.

Coturn UDP Port Range

You may leave the default value. If not, please enter a port range in format 12345:13456. The default value is 49152:65535.

Updates

CrossBox has a built-in update system that keeps your installation up to date and automatically updates CrossBox when new updates are available.

At what hour to perform an update

Enter an hour when you wish to check for updates. Uses a 24h format and accepts a number between 0 and 23. 0 means 00:00.

Application Limits

Max Number of Workers

Max number of CPU cores that the application will utilize. Depending on the application traffic/usage you can specify a maximum number of workers based on the number of cores your server has.

Each worker takes about 70-100 MB of RAM.

Max Request Size

In megabytes, enter maximum request body size that the end user can send. For example, users won't be able to upload a file larger than what's defined here. The default value is 1024MB.

Chat Channel Max Participant Count

The maximum number of participants allowed in a chat channel. Increase this if you plan on having more than 50 persons attending. This is a per one channel (room) limitation.

CrossBox Admin

Admin E-mail Address

Admin e-mail address. You may use any email address you want.

Admin Password

This password should be very complex.

Localization

Default Language

Choose the default language for the App. If you want to add your own language, you can do so by translating a language file and sending it back to us. We will then ship it with the next update.

Logging

Error Reporting

In case of an error, the server will securely send us the traceback and the message - so our team can detect and fix the issue ASAP, with an aim of minimizing the bug's impact on larger audiences.

MySQL

Use CrossBox MySQL database

Enable if you don't have a MySQL server already running on the server.

If you enable this feature, CrossBox will use its own MariaDB (MySQL) server for storing users' data. Otherwise, it will try to connect to a MySQL server already running on the system and use that server for storing database data.

You can also uncheck this option to enter MySQL server details manually. This is useful if you're connecting CrossBox to remote MySQL server or a MySQL Cluster.

> ⚠ **Root MySQL user is required.** The root privileges are required because CrossBox manages the creation of users' databases.

Mail

Enable mail server integration

CrossBox has two ways of integrating with a mail server:

1. Use the mail server already running on the machine (for example when installing on a cPanel/Plesk/Direct Admin server)
2. Automatically install and set up a CrossBox mail server (when installing a bare metal/standalone version, or a mail server is not found on the server)

In the first case, the installer will try to find locally available IMAP/SMTP servers and use those for everything related to email.

In the second case, you'll be able to enable or disable following mail server features:

Enable Anti-Virus Service

Enable this if you want an anti-virus to scan emails for malicious content. Currently, only ClamAV is supported. If you have ClamAV already running on the server, CrossBox will use that one automatically.

> ⚠ ClamAV requires 600 MB of RAM.

Enable Anti-Spam Service

Enable this if you want an anti-spam service to check whether the e-mail being sent/received is spam or not. Automatic spam/ham learning is also enabled with this service.

⚠ Enabling this service requires an additional 250 MB of RAM.

E-mail Message Max Size

Enter the maximum allowed email message size in megabytes. Generally, it should not exceed 50MB.

Automatically create postmaster@ and admin@

This can save you some time by creating often needed email addresses automatically for you.

Other

Google Analytics

To track App usage in real time you need to enter a valid Google Analytics Tracking ID in UA-XXXXX-Y format.

CrossBox Document Converter (CDC)

Service that allows users to view (.docx, .doc, .xls, .ppt + many others) without having to download the files on their device. Users can also download PDF versions of various formats that are supported. This service requires at least 100MB of RAM

Branding

Branding is a great way to white label CrossBox as your own product or service. You can give it a custom name, upload your own logo and change the color scheme to best match your brand. Also, branding addon allows you to have a fully branded CrossBox Client Apps (Web, Desktop, Android & iOS).

✓ Branding feature is most useful for service and hosting providers.

App Name

Enter a custom name for the application.

App Description

Enter a custom Application slogan or a motto. This is usually shown beneath the App Name.

App Product URL

URL of the web page advertising the application. Users are taken to this URL when they want to find out more about the App.

Theme

Lets you brand the App with your own color scheme. Apart from that, you can also upload your own App logo. Image for the logo should be in a PNG format with 1024x1024 px dimensions. CrossBox will then use that image to create logo versions of different dimensions used throughout the application.

iOS and Android App Settings

iOS and Android sections let you define App manifests that Apple and Google stores require when publishing the Apps. You will need to fill in a couple of details:

- **App Identifier** (This is the Application ID. For example, com.example.appname)
- **App Author** (The name of the Author)
- **App Author E-mail** (E-mail address of the author)
- **App Support URL** (URL where you provide customer support)
- **App Store URL** (URL for downloading iOS app)
- **Play Store URL** (URL for downloading Android app)

Push Notifications

This enables push notifications for your branded CrossBox Android and iOS Apps. They keep your end users up to date with new emails, chats, calls, files, and reminders.

> GCM is not recommended anymore. Please use FCM - Firebase Cloud Messaging. **We strongly recommend that you use FCM for both Android and iOS versions.**

> When using FCM, you need to obtain **FCM Sender ID** and **FCM Key**. Please follow this blog post to find out how to obtain these.

If, however, you must use GCM, then you'll need to fill in GCM Sender ID and GCM Key. To obtain these, please start here.

We do not recommend using APNs, use FCM instead. If you insist on using APNs, then you'll need to obtain APNs certificate as explained in "Generate an APNs client TLS certificate" and then upload it by using "Upload APNs Certificate" upload box during the installation.

CrossBox Server

Admin Area

Intro

CrossBox features a simple to use administration interface which is used for managing a CrossBox server.

Accessing Admin Area

To access the administration, visit your CrossBox URL and log in by using your admin credentials.

Restoring access to Admin Area

If you lost or forgot your admin credentials, these are the CLI commands that you can run via SSH (as root) to restore them:

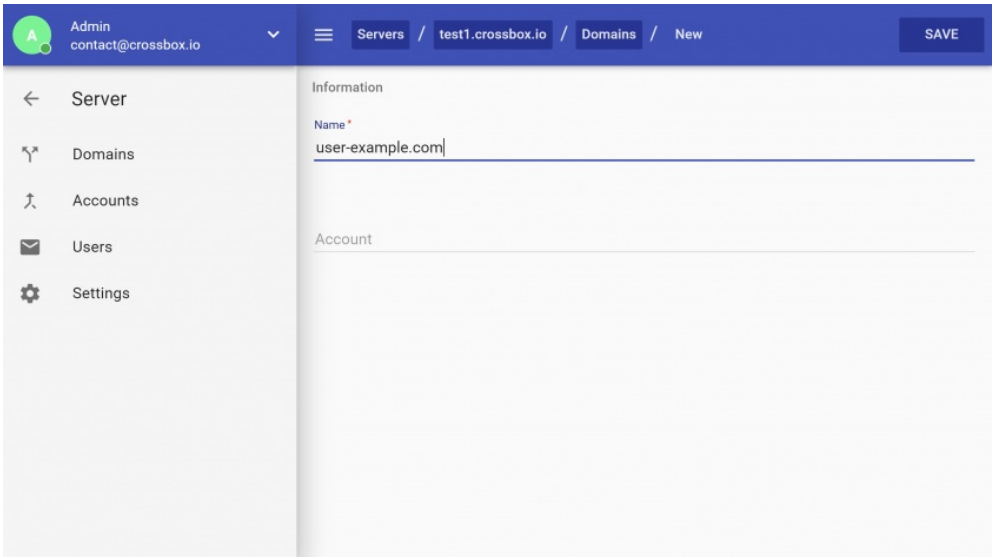
`$: crossbox info` - get general information about the CrossBox installation

`$: crossbox admin password` - change admin password

`$: crossbox admin email` - change admin email address

Adding a Domain (Standalone only)

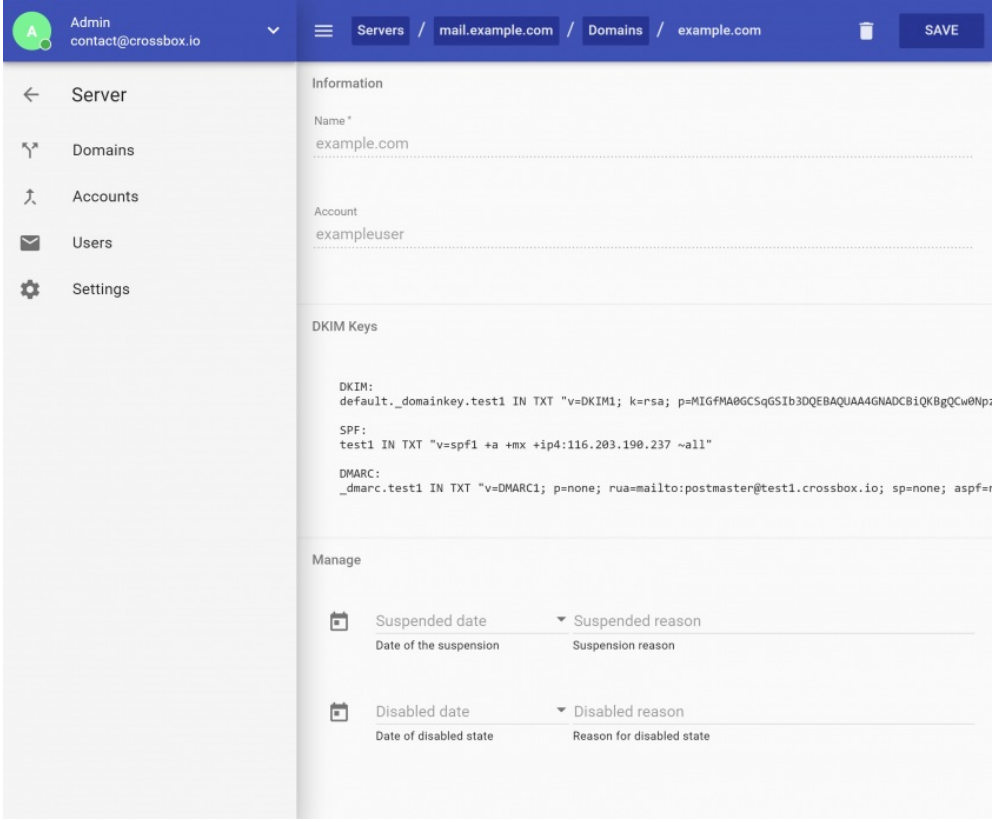
- Go to Admin -> Servers
- Pick a server where you want the domain to reside
- Go to Domains -> Click on the FAB button in the bottom-right corner
- Fill in the form:



- A **name** is the domain's name.
- An **account** is a Linux user that you wish to associate with the domain. If you don't have one, please add a Linux user first by going to the "Accounts" in the side nav.
- Make sure that the domain's DNS zone has an **MX record pointing to** the CrossBox server.

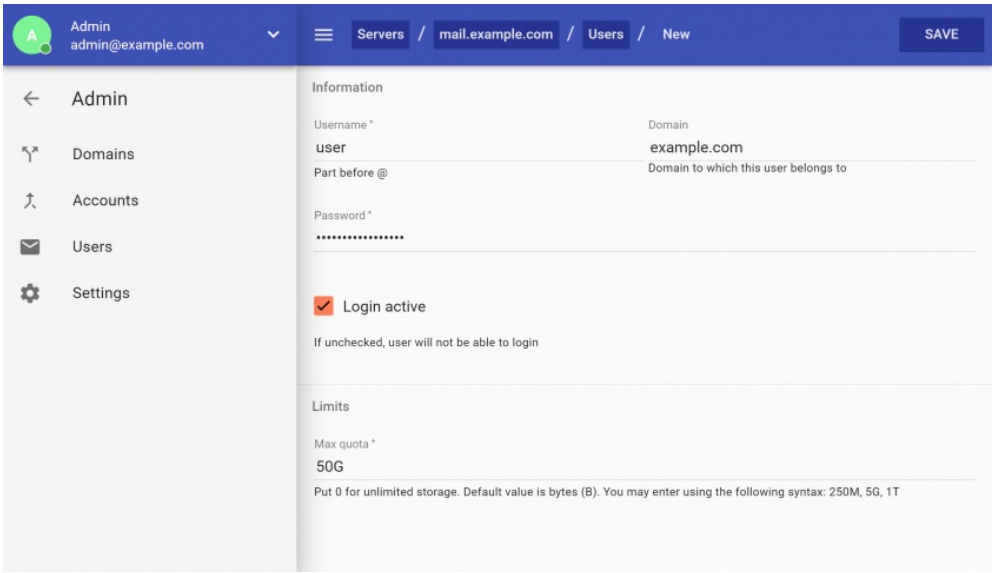
After you click Save, the domain will be created and you can proceed to add e-mail addresses by creating your first CrossBox user.

When submitted, the form will also output DNS records for DKIM, SPF, and DMARC. **Make sure you add these to your domain's DNS zone so the outgoing emails can be correctly signed, which maximizes the deliverability:**



Adding a CrossBox Account and Email Address (Standalone only)

- Go to Admin -> Servers
- Pick a server where the domain resides
- Go to Users -> Click on the FAB button in the bottom-right corner
- Fill in the form:

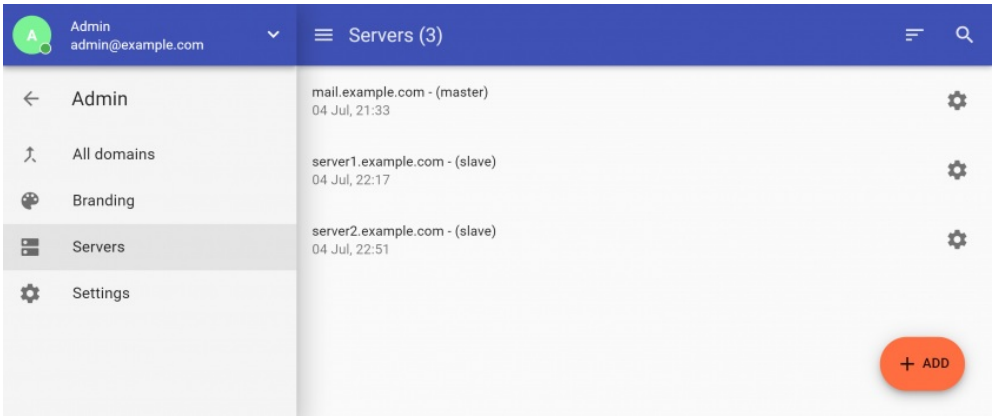


- **Username** is the first part of the e-mail address. For example **joe.doe**
- A **domain** is the second part of the e-mail address. For example **domain.com**
- Enter the **password** user will use to access CrossBox and IMAP/POP3/SMTP protocols
- **Login active** checkbox defines if a user is allowed to log in
- **Max quota** is a quota limit that CrossBox will not allow the user to exceed

> After you click Save, the CrossBox user you specified will be created along with a fully functional e-mail address.

Managing a Server

To manage a CrossBox Server login to your Admin Area, and click on a server you wish to manage.

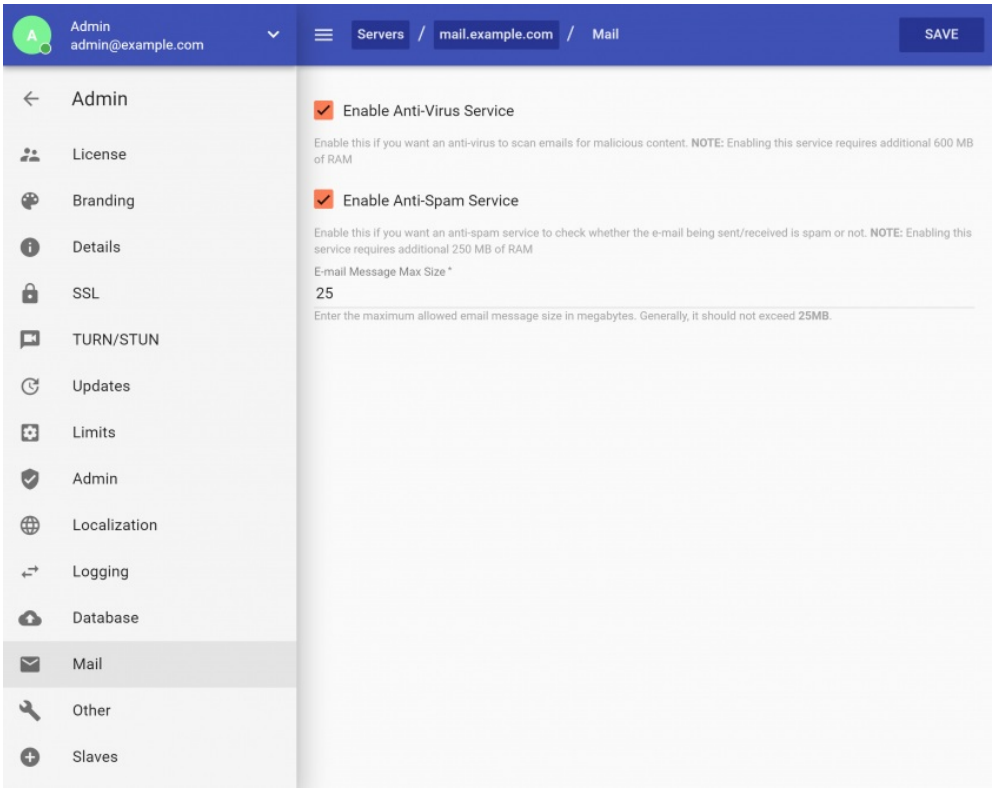


Easy to use Admin Interface lets you stay in full control over:

- Server status (stop/start/restart)
- Domains
- Users
- System Accounts

Changing Server's Settings

To change the [CrossBox Server Options], login to your Admin Area, pick a server you wish to modify and click "*Settings*".



CrossBox Server

Command Line Tools

The "crossbox" CLI Command

This is the CrossBox command line utility. Use it to view status, restart services, view logs, trigger updates, and a lot more.

> ⚠ A **root** user is required to run the commands.

Monitoring

> crossbox stop|start|restart|status|enable|disable

Easily stop, start, restart, enable, disable crossbox services. You can also view status of all services, for exmaple: `crossbox status`

Updating

> crossbox update

Perform a manual update check and update the application if possible.

Logs

> crossbox log [--lines]

Displays all logs in real time. If it's too verbose, try using crossbox log <service>

--lines
defines a number of "last lines" to display for each log file

Service Specific

> crossbox stop|start|restart|disable|enable <service>

You can also specify several <services> by separating multiple services by space, so for example: `crossbox restart redis exim`

Medic

> crossbox medic

CrossBox Medic will collect all the relevant CrossBox logs from your server, send them to us and give you a unique key which you can reference when requesting support.

Utility

> crossbox info

Will tell you general information about your CrossBox installation.

> crossbox admin password

Allows you to reset the admin's password.

crossbox admin email

Allows you to reset admin's e-mail address.

crossbox changelog

Shows latest update messages.

To view full list of crossbox commands, run `crossbox -h`

Debugging

You can enable the debug mode if you are faced with a bug and you want to find out more about what the application is doing.

crossbox debug disable|normal|unsafe

Use unsafe mode only when you don't see anything indicating a problem. When using the unsafe debug mode the application may leak information, so use it with care.

NOTE: Changing the debug mode will restart all CrossBox services.

CrossBox Server

Performance Tuning

CrossBox Server

Max Number of Workers

Increase the number of workers by setting Max Number of Workers to your total CPU count. This will allow CrossBox to utilize a multi-core processing. For example, if you have a server with 8 CPUs, set this to 8.

Remember that each worker also allocates 70-100 MB of RAM. Multiply this by a number of workers and you'll get the total amount of additional RAM which the change of this setting will require.

Linux Server

For a high-performance system trying to serve thousands of concurrent network clients, default Linux kernel parameters are often too low. Consider making following changes

- Increase max open files to 100,000 from the default (typically 1024). In Linux, every open network socket requires a file descriptor. Increasing this limit will ensure that lingering `TIME_WAIT` sockets and other consumers of file descriptors don’t impact our ability to handle lots of concurrent requests.
- Decrease the time that sockets stay in the `TIME_WAIT` state by lowering `tcp_fin_timeout` from its default of 60 seconds to 10. You can lower this even further, but too low, and you can run into socket close errors in networks with lots of jitter. We will also set `tcp_tw_reuse` to tell the kernel it can reuse sockets in the `TIME_WAIT` state.
- Increase the port range for ephemeral (outgoing) ports, by lowering the

minimum port to 10000 (normally 32768), and raising the maximum port to 65000 (normally 61000). *Important:* This means you can't have server software that attempts to bind to a port above 9999! If you need to bind to a higher port, say 10075, just modify this port range appropriately.
- Increase the read/write TCP buffers (`tcp_rmem` and `tcp_wmem`) to allow for larger window sizes. This enables more data to be transferred without ACKs, increasing throughput. We won't tune the total TCP memory (`tcp_mem`), since this is automatically tuned based on available memory by Linux.
- Decrease the VM `swappiness` parameter, which discourages the kernel from swapping memory to disk. By default, Linux attempts to swap out idle processes fairly aggressively, which is counterproductive for long-running server processes that desire low latency.
- Increase the TCP congestion window, and disable reverting to TCP slow start after the connection is idle. By default, TCP starts with a single small segment, gradually increasing it by one each time. This results in unnecessary slowness that impacts the start of every request – which is especially bad for HTTP.

Kernel Parameters

To start, edit `/etc/sysctl.conf` and add these lines:

```
# /etc/sysctl.conf
# Increase system file descriptor limit
fs.file-max = 100000

# Discourage Linux from swapping idle processes to disk (default = 60)
vm.swappiness = 10

# Increase ephermeral IP ports
net.ipv4.ip_local_port_range = 10000 65000

# Increase Linux autotuning TCP buffer limits
# Set max to 16MB for 1GE and 32M (33554432) or 54M (56623104) for 10GE
# Don't set tcp_mem itself! Let the kernel scale it based on RAM.
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.core.rmem_default = 16777216
net.core.wmem_default = 16777216
net.core.optmem_max = 40960
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216

# Make room for more TIME_WAIT sockets due to more clients,
# and allow them to be reused if we run out of sockets
# Also increase the max packet backlog
net.core.netdev_max_backlog = 50000
net.ipv4.tcp_max_syn_backlog = 30000
net.ipv4.tcp_max_tw_buckets = 2000000
net.ipv4.tcp_tw_reuse = 1
net.ipv4.tcp_fin_timeout = 10

# Disable TCP slow start on idle connections
net.ipv4.tcp_slow_start_after_idle = 0

# If your servers talk UDP, also up these limits
net.ipv4.udp_rmem_min = 8192
net.ipv4.udp_wmem_min = 8192

# Disable source routing and redirects
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.accept_source_route = 0

# Log packets with impossible addresses for security
net.ipv4.conf.all.log_martians = 1
```

Since some of these settings can be cached by networking services, it's best to reboot to apply them properly (`sysctl -p` does not work reliably).

Open File Descriptors

In addition to the Linux `fs.file-max` kernel setting above, we need to edit a few more files to increase the file descriptor limits. The reason is the above just sets an absolute max, but we still need to tell the shell what our per-user session limits are.

So, first edit `/etc/security/limits.conf` to increase our session limits:

```
# /etc/security/limits.conf
# allow all users to open 100000 files
# alternatively, replace * with an explicit username
* soft nofile 100000
* hard nofile 100000
```

Next, `/etc/ssh/sshd_config` needs to make sure to use PAM:

```
# /etc/ssh/sshd_config
# ensure we consult pam
UsePAM yes
```

And finally, `/etc/pam.d/sshd` needs to load the modified `limits.conf`:

```
# /etc/pam.d/<g class="gr_ gr_808 gr-alert gr_spell gr_inline_cards gr_run_anim
ContextualSpelling ins-del multiReplace" id="808" data-gr-id="808">sshd</g>
# ensure pam includes our limits
session required pam_limits.so
```

You can confirm these settings have taken effect by opening a new ssh connection to the box and checking `ulimit`:

```
ulimit -n
100000
```

TCP Congestion Window

Finally, let's increase the TCP congestion window from 1 to 10 segments. This is done on the interface, which makes it a more manual process that our `sysctl`settings. First, use `ip route` to find the default route, shown in bold below:

```
route
default via 10.248.77.193 dev eth0 proto kernel
10.248.77.192/26 dev eth0  proto kernel  scope link  src 10.248.77.212
```

Copy that line, and paste it back to the `ip route change` command, adding `initcwnd 10` to the end to increase the congestion window:

```
route change default via 10.248.77.193 dev eth0 proto kernel initcwnd 10
```

To make this persistent across reboots, you'll need to add a few lines of bash like the following to a startup script somewhere. Often the easiest candidate is just pasting these lines into `/etc/rc.local`:

```
defrt=`ip route | grep "^default" | head -1`
```

```
ip route change $defrt initcwnd 10
```

Hardware

- More CPUs means more concurrency
- More than 4GB of RAM is often not required, even with high concurrency
- Faster Disk means faster IO, therefore we always recommend having an SSD

CrossBox Server

Uninstall

> Run the command **crossbox uninstall**

This command will completely erase CrossBox from the machine, including user data generated by CrossBox.

After 10 seconds the uninstallation will begin - once this step starts you should not interrupt the script as it might not completely delete the software. Depending on the amount of work and performance of the machine the uninstallation might take from a few seconds to several minutes

CrossBox Server

Migration

Server migration

This is a guide on how to migrate the CrossBox instance to a new server

Preparation

Make sure that you do the following steps before the migration starts:

- If you are just going to move the entire disk to another machine, you should read the following guide: https://crossbox.io/documentation/page/post-machine-migration
- The new server should have rsync, ssh installed and a default public key generated
- The old server ssh public key must be added to the new server
- The new server should have the same hostname and PTR (rDNS) as the old one
- Change your A/CNAME record to the new server ip address
- On your old server, ping the new server hostname, it should resolve to the new ip address
- **If you are moving CrossBox from a panel**, you should **first** migrate the user data and the panel itself. **CrossBox should be migrated last.**
- Plan a complete outage of the server during the migration period for at least an hour. Speed of the process is determined by the data transfer between the old and the new server.

Steps

Run the following command on the old server:

```
crossbox move newhostname.com
```

The command will inform you what's going to happen with each step.

When migrating panel integrated installations, only CrossBox core data will be copied to the new server. The end user data will not be copied, because panel migration tools will do that for you.

Standalone, or master server / node CrossBox installations are migrated completely (excluding the situations where you have special configurations, like custom MySQL hosting, file hosting etc)

After the command completes, you will have to wait at least 10 minutes until the CrossBox recognizes the changed machine and readjusts it's internal configurations. Make sure that all needed incoming and outgoing ports are allowed.

Check the CrossBox status command to verify all services are up:

```
crossbox status
```

The CrossBox instance should be accessible via relevant hostname and port combinations based on your installation configuration.

CrossBox Server

Post machine migration

If you just migrated your disk to another machine with a different ip, you should do the following set of commands:

```
crossbox config set -k public_ip_address -s "new_ip_address"
crossbox sync all
```

After that you should wait at least 10 minutes until CrossBox recognizes the new environment.

Then you should verify if all services are running:

```
crossbox status
```

Make sure all firewall rules are working correctly and check if the instance is accessible once again.

Control Panels

Control Panels

cPanel

Intro

CrossBox for cPanel is one of our most popular control panel integrations. It's Plug-and-Play in its core, covering 95% of all cPanel setups out of the box. Our deep integration with cPanel hooks and APIs enables an automated and fine-tuned experience for both the end-user and service provider.

All data and computing are 100% hosted/done on the cPanel server.

Benefits

End-users:

- Easy access either via cPanel's webmail system or direct URL
- No complicated IMAP/SMTP/POP3 configuration, only email address, and password are required
- Instant access to
  - Email
  - Chat
  - Audio/video calls
  - Conferences
  - Files
  - Contacts
  - And many other features, directly integrated with a cPanel hosting account
- Real-time push notifications
- Change password within CrossBox
- Recover access to an email account

Service Providers:

- Provide users with a modern communication suite that is deeply integrated with and running on a cPanel server
- Easier customer onboarding
- Better brand exposure
- No changes to the cPanel server, services, or configuration files needed
- Everything is 100% hosted on the cPanel server itself
- cPanel account quota support
- WHM Feature List support

Automatic cPanel Integration

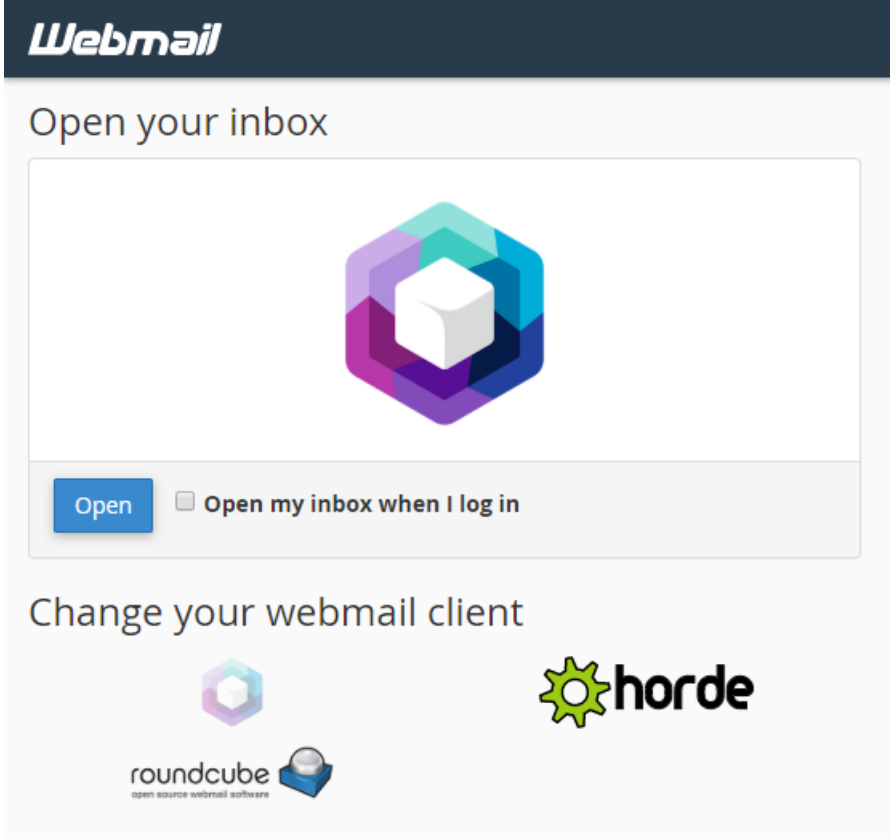
MTA/Mail Server

CrossBox supports Exim and Dovecot out of the box, and won't make any changes to your existing MTA/mail server configuration.

From standard Exim configurations to smart external relays, CrossBox supports any configuration.

Webmail

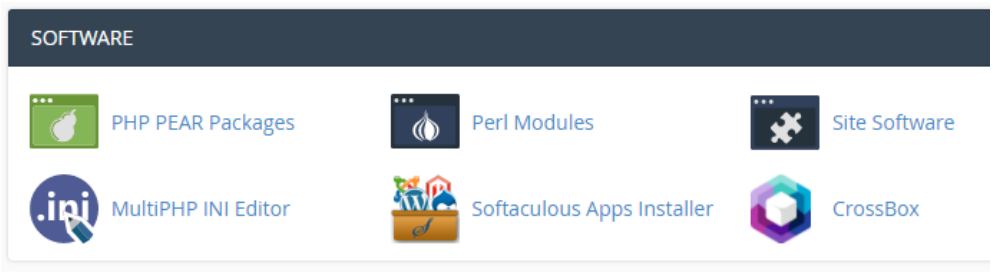
cPanel users can choose CrossBox as their Webmail Application of choice when accessing cPanel's Webmail.



Note that if you have a Branding Addon purchased, the cPanel Webmail interface will show your custom App Logo.

cPanel UI

Shortcut for CrossBox is also placed in the *Software* section of the user's cPanel interface.



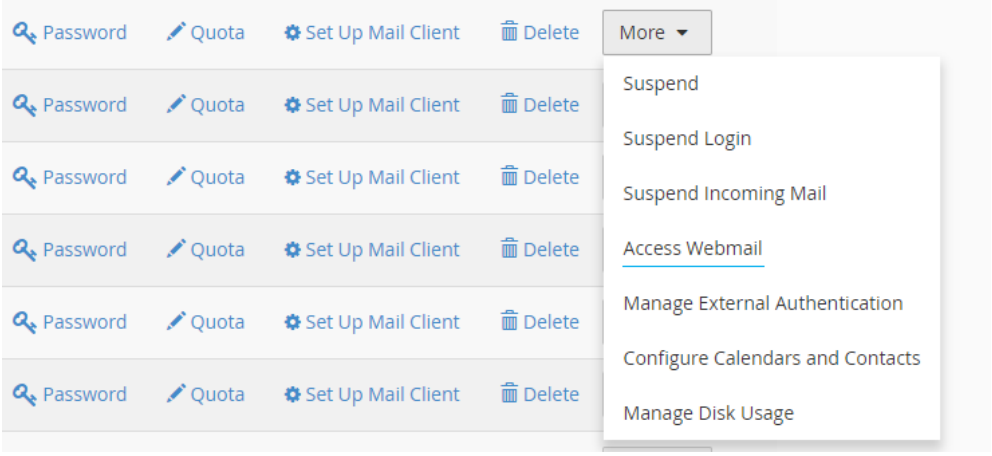
Note that if you have a Branding Addon purchased, the cPanel's software section will show App Logo and App Name you defined during the installation.

Mailbox Format Support

Both *Maildir* and *MDBOX* are fully supported when using CrossBox with cPanel.

E-mail Auto-Login

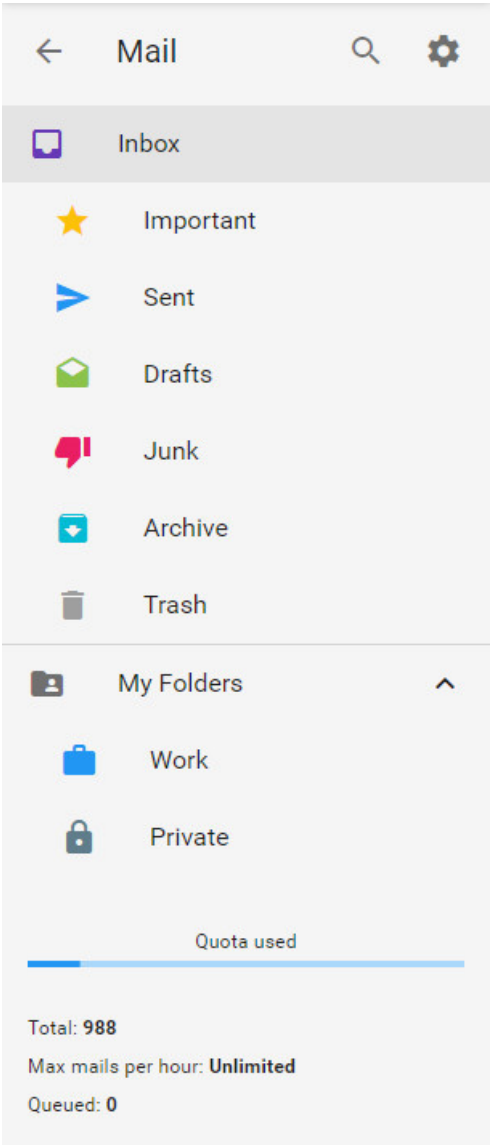
cPanel's *Access Webmail* option that features passwordless auto-login is fully supported.



E-mail Quota

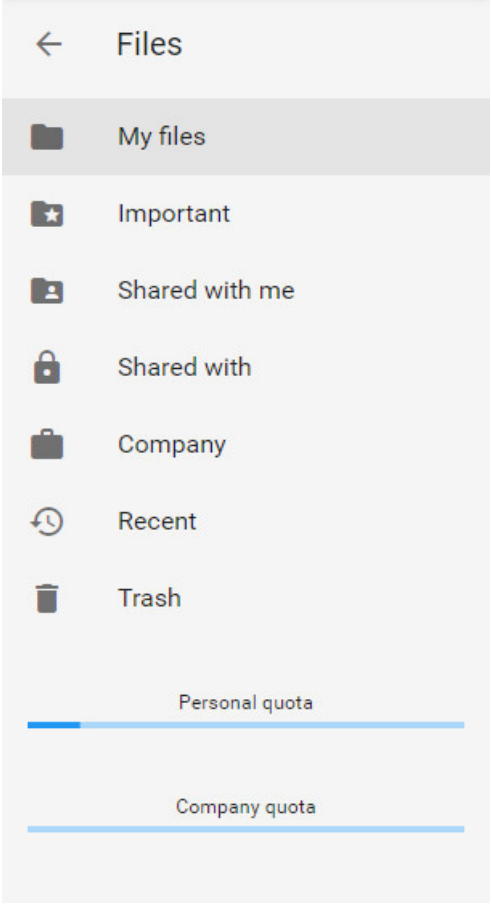
CrossBox supports cPanel's e-mail disk quota and sending limitations, as seen in the image below.

Also, it can display the number of queued emails and whether the outgoing limitations apply.



Disk Quota

All the data CrossBox users create and files they store are counted against their hosting account's disk quotas.



MySQL Quota

CrossBox respects the hosting account's MySQL quota and each hosting account that has CrossBox enabled has its own MySQL database.

The database is created under that hosting account's ownership after the user successfully logs into CrossBox for the first time.

Password Change Syncing

When a user changes its e-mail password via the cPanel interface, the user's CrossBox password is also changed, and vice versa.

Hosting Account Suspension

When a hosting account is suspended, CrossBox users of that hosting account are also suspended and all users automatically logged out of the application.

Hosting Account Termination

When a hosting account gets terminated, CrossBox for that account is also deleted, including users' CrossBox data.

Backup

CrossBox users' data is stored in their hosting account's directory and each hosting account has its own CrossBox database.

This means that backing up CrossBox can be done automatically by the cPanel's native *Backup System*.

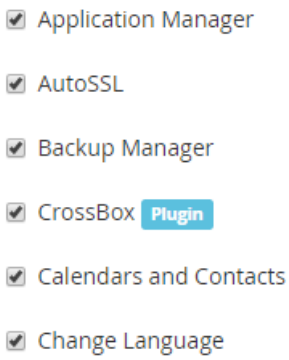
You can also use any other backup solution for cPanel servers like JetBackup and it will work out of the box.

WHM Feature Manager

In WHM, you can use the Feature Manager to enable or disable different features for different hosting packages.

By enabling or disabling the CrossBox Plugin in the feature list, you can control who has access to CrossBox. This is useful if you want to allow access to CrossBox only to users on your more expensive hosting packages.

By default, the CrossBox plugin is enabled for all users.



Note that if you have a Branding Addon purchased, the WHM Feature Manager will show your custom App Name.

Compatibility With Other Software

MailScanner Support

Fully supported.

SpamExperts Support

Fully supported.

MailChannels Support

Fully supported.

SpamAssassin Support

Fully supported.

Apache Solr Support

Fully supported.

Firewall Support

CrossBox integrates automatically with *CSF*, one of the most popular firewalls for cPanel servers. CrossBox will also automatically detect and use *Firewalld*, the native firewall for RedHat/CentOS servers.

ClamAV Support

If you already have *ClamAV* running on your cPanel server, CrossBox will auto-detect it and then use it to scan users' CrossBox data in real-time, thus preventing any harmful files from being uploaded to the server.

CloudLinux Support

CrossBox is fully compatible with CloudLinux and CageFS.

Plesk

Intro

CrossBox fully supports Plesk. It integrates with Plesk automatically and installing CrossBox on a Plesk server is as easy as clicking your way through the Web Installation Wizard.

Installation

Installing CrossBox on a Plesk server does not in any way differ from installing CrossBox on any other server, but there are a couple of things to keep in mind.

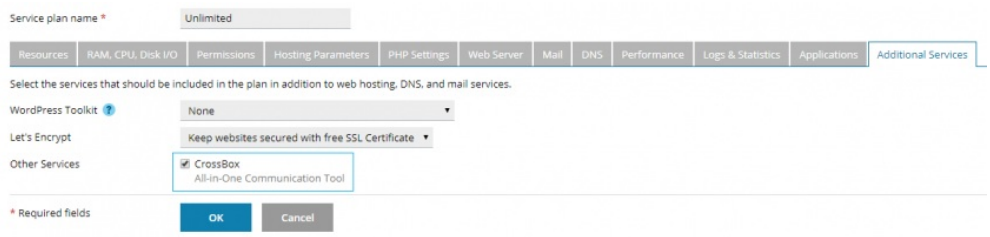
Hosting Providers

If you are a hosting provider and you have multiple Plesk servers, then you should think about Creating a CrossBox Cluster. By doing so, you'll be able to advertise a single CrossBox link for all users (my-app.your-hosting.com), no matter what specific Plesk server they are hosted on. Combine that with our Branding Addon which lets you white-label CrossBox as your own and you'll be able to offer your Plesk based hosting customers a powerful set of communication tools with all, or just specific hosting plans, while keeping everything hosted on your customer's Plesk hosting account, and in full compliance with hosting package limits.

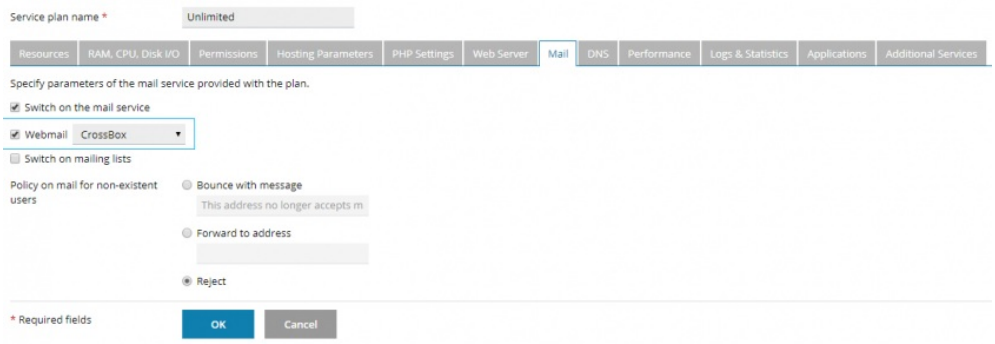
Single Plesk Server

To install CrossBox on a single Plesk server, please follow the Quick Installation Guide. After you've finished installing CrossBox, you'll need to enable it for some or for all hosting accounts. You can do so by enabling the CrossBox Plugin using a Plesk's "*Additional Services*" feature found under "*Service Plans > Hosting* Plans".

> You need to do these steps for every Hosting Plan that you wish to have CrossBox enabled.



> Also, make sure that CrossBox is enabled as a Webmail option under the "*Mail*" tab.



> Note that if you have a Branding Addon purchased, the App Name you chose during the installation will be shown instead of CrossBox.

Plesk Integration

Mail Server

Plesk already ships with Postfix/Qmail and Dovecot/Courier, but currently, only Postfix, Qmail, and Dovecot are supported. When using CrossBox with Plesk, CrossBox will not use its own mail server setup, which is common for a Standalone/Bare Metal version. Instead, it will scan for a local mail server and use that one by default. By doing so, CrossBox just **plugs and plays** seamlessly with your existing mail server setup, with no additional changes needed.

Webmail

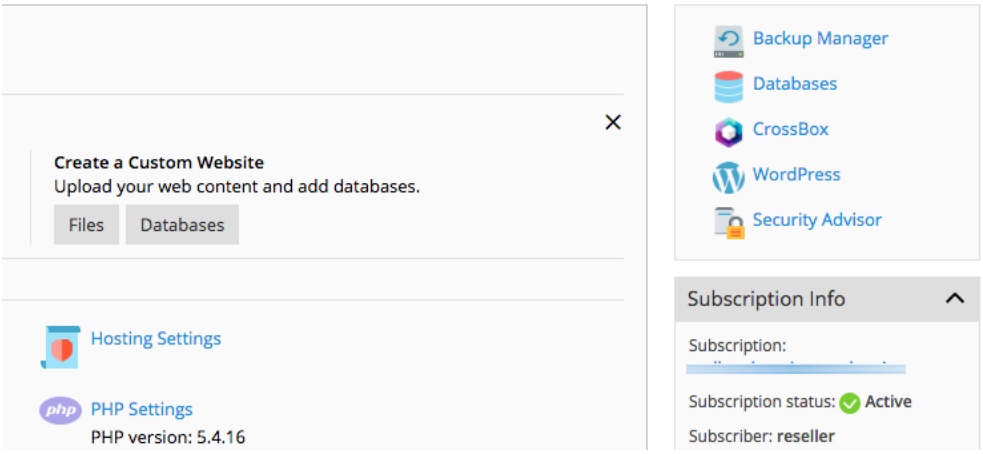
> Plesk hosting accounts that have CrossBox enabled can choose CrossBox as their Webmail Application of choice when accessing Plesk's Webmail.

Mail Service	Webmail	Mail sent to non-existent users
On	CrossBox	Reject
On	CrossBox	Reject
On	CrossBox	Reject
On	CrossBox	Reject

> Note that if you have a Branding Addon purchased, the Plesk Webmail interface will show App Logo and App Name you chose during the installation.

Plesk UI

> Shortcut for CrossBox is also placed in the *Additional Services* section of user's Plesk interface.



Note that if you have a Branding Addon purchased, the Plesk's Additional Services section will show App Logo and App Name you defined during the installation.

E-mail Accounts are CrossBox Users

Every e-mail address on a hosting account that has CrossBox enabled is treated as a CrossBox user. By creating an e-mail address via Plesk's "E-mail Accounts" feature, the user is also automatically creating a CrossBox account. The e-mail address being created is treated as the main user identifier.

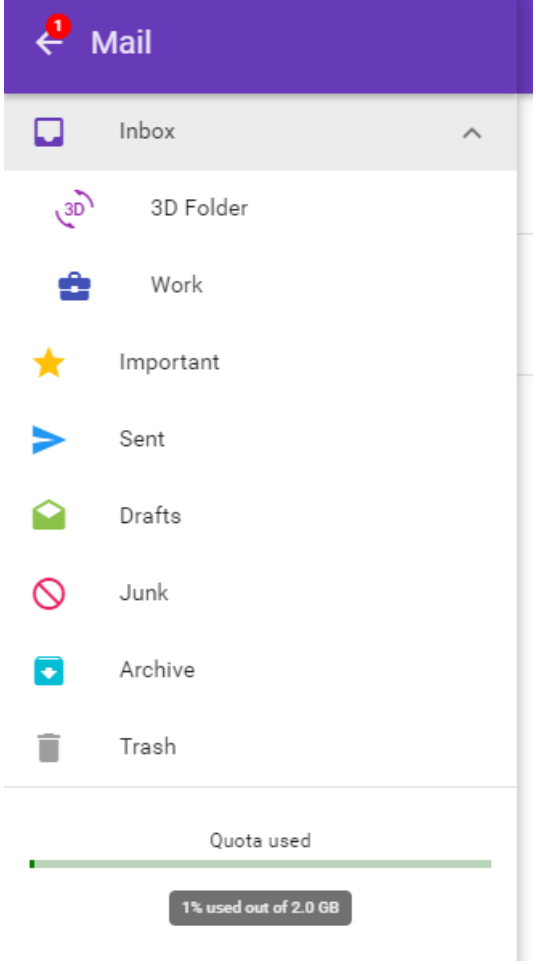
When a user changes its e-mail password, user's CrossBox password is also changed.

Mailbox Format Support

Maildir is fully supported when using CrossBox with Plesk.

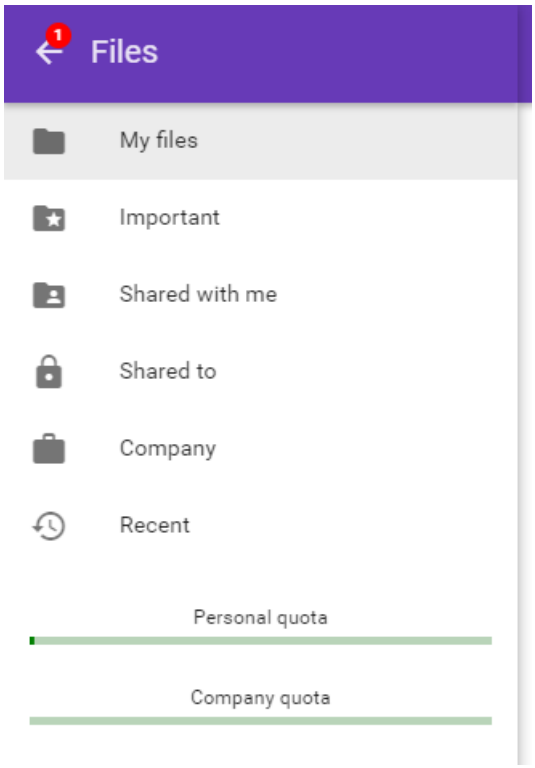
E-mail Quota

CrossBox fully supports Plesk's e-mail quota limitations, as seen in the image below.



Disk Quota

All the data CrossBox users create and files they store are counted against their hosting account's disk quotas.



Bandwidth Quota

The bandwidth CrossBox users make **is not** counted against their hosting account's bandwidth quota. The reason behind this is that Plesk currently

doesn't expose a mechanism to do this.

MySQL Quota

CrossBox respects hosting account's MySQL quota and each hosting account that has CrossBox enabled has its own MySQL database, which is created under that hosting account's ownership after the user successfully logs into CrossBox for the first time.

Hosting Account Suspension

When a hosting account is suspended, CrossBox users for that hosting account are also suspended.

Hosting Account Termination

When a hosting account gets terminated, CrossBox for that account is also deleted, including users' CrossBox data.

Backup

Taking into consideration that CrossBox users' data is stored in the hosting account's directory and that every hosting account has its own CrossBox database, backing up CrossBox can be done automatically by Plesk's native *Backup System*. You can also use any other backup solution for Plesk and it should work out of the box.

Compatibility With Other Software

Firewall Support

CrossBox integrates automatically with *CSF*, one of the most popular firewalls for Plesk servers.

CrossBox integrates automatically with *Firewalld*, the native firewall for RedHat/CentOS servers.

CrossBox integrates automatically with *UFW*, the native firewall for Debian/Ubuntu servers.

CageFS Support

CrossBox is fully compatible with CloudLinux *CageFS* virtualized file system.

SpamAssassin Support

CrossBox fully supports *SpamAssassin,* which is a popular spam filter for Plesk servers.

SpamExperts Support

If you are using *SpamExperts* on your Plesk server, you can keep on using it without any problems with CrossBox.

ClamAV Support

If you already have *ClamAV* running on your Plesk server, CrossBox will auto detect it and use it to scan users' CrossBox data.

Control Panels

Direct Admin

Intro

As of v0.3.9, CrossBox fully supports Direct Admin. It integrates with Direct Admin automatically and installing CrossBox on a Direct Admin server is as easy as clicking your way through the Web Installation Wizard.

Installation

Installing CrossBox on a Direct Admin server does not in any way differ from installing CrossBox on any other server, but there are a couple of things to keep in mind.

Hosting Providers

If you are a hosting provider and you have multiple Direct Admin servers, then you should think about Creating a CrossBox Cluster. By doing so, you'll be able to advertise a single CrossBox and IMAP/SMTP/POP3 URLs for all users (my-app.your-hosting.com), no matter what specific Direct Admin server they are hosted on. Combine that with our Branding Addon which lets you white-label CrossBox as your own and you'll be able to offer your Direct Admin based hosting customers a powerful set of communication tools with all, or just specific hosting plans, while keeping everything hosted on your customer's Direct Admin hosting account, and in full compliance with hosting package limits.

Single Direct Admin Server

To install CrossBox on a single Direct Admin server, please follow the Quick Installation Guide.

Automatic Direct Admin Integration

Mail Server

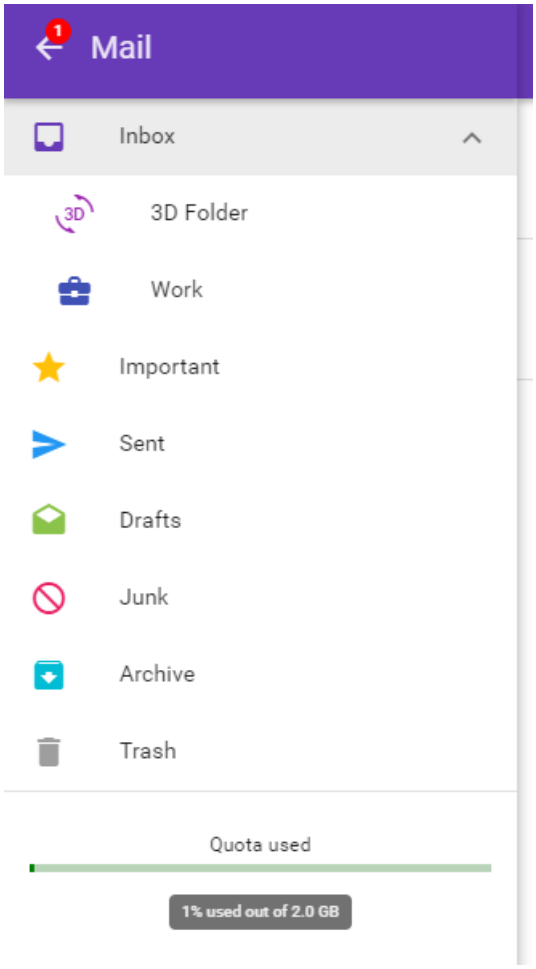
When using CrossBox with Direct Admin, we will scan for a local mail server and use that one by default. By doing so, CrossBox just **plugs and plays** seamlessly with your current mail server setup, without making any changes to your existing infrastructure.

Mailbox Format Support

Both *Maildir* and *MDBOX* are fully supported when using CrossBox with Direct Admin.

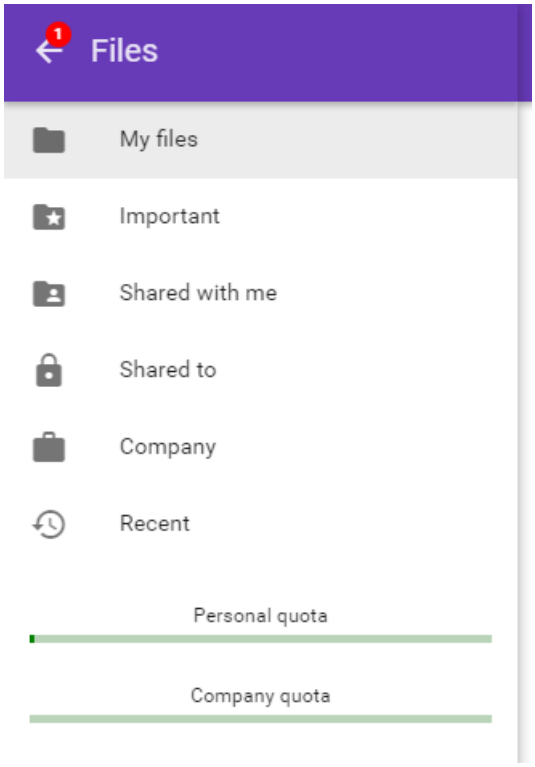
E-mail Quota

CrossBox fully supports Direct Admin's e-mail quota limitations, as seen in the image below.



Disk Quota

All the data CrossBox users create and files they store are counted against their hosting account's disk quotas.



MySQL Quota

CrossBox respects hosting account's MySQL quota and each hosting account that has CrossBox enabled has its own MySQL database, which is created under that hosting account's ownership after the user successfully logs into CrossBox for the first time.

Hosting Account Suspension

When a hosting account is suspended, CrossBox users of that hosting account are also suspended and automatically logged out of the application.

Hosting Account Termination

When a hosting account gets terminated, CrossBox for that account is also deleted, including users' CrossBox data.

Backup

Taking into consideration that CrossBox users' data is stored in their hosting account's directory and that each hosting account has its own CrossBox database, backing up CrossBox is done automatically by Direct Admin's native *Backup System*. You can also use any other backup solution for Direct Admin and it will work out of the box.

Compatibility With Other Software

MailScanner Support

If you are using *MailScanner* on your Direct Admin server, you can keep on using it without any problems with CrossBox.

SpamExperts Support

If you are using *SpamExperts* on your Direct Admin server, you can keep on using it without any problems with CrossBox.

MailChannels Support

If you are using *MailChannels* with your Direct Admin server, you can keep on using it without any problems with CrossBox.

SpamAssassin Support

CrossBox fully supports *SpamAssassin,* which is a popular spam filter for Direct Admin servers.

Apache Solr Support

CrossBox utilizes a fast Full-Text Search Indexing if Apache Solr is enabled on a Direct Admin server.

Firewall Support

CrossBox integrates automatically with *CSF*, one of the most popular firewalls for Direct Admin servers. It can also detect and use *Firewalld*, the native firewall for RedHat/CentOS servers.

ClamAV Support

If you already have *ClamAV* running on your Direct Admin server, CrossBox will auto-detect it and then use it to scan users' CrossBox data in real-time, thus preventing any harmful files being stored on the server.

CageFS Support

CrossBox is fully compatible with CloudLinux *CageFS* virtualized file system.

Control Panels

Other Control Panels

Intro

Due to CrossBox having a plug and play design by its nature, it can be installed on a server running any control panel, even if the official implementation (so-called deep-integration) for that panel doesn't exist.

CrossBox will auto-detect software which is already running on the server and adjust its operations accordingly.
For example, if the server has an active mail server already installed, CrossBox will not install its own and will use that one instead.

Installation

Install CrossBox as you would on any other server, by following our Quick Installation Guide.

CrossBox will not interfere with any software you already have installed or running on the server as it uses its own packages and libraries that are prebuilt and shipped with CrossBox. This means it will not install any software via your system's package manager or touch your system's dependencies in any way.

Post Installation

After having successfully completed the installation, use the Admin area to enable CrossBox for domains hosted on the server.

Control Panels

Docker

You can easily Dockerize CrossBox instead of installing it on a VPS or a Dedicated server.

```
mkdir -p /var/crossbox
cd /var/crossbox
```

Create a Dockerfile

```
nano Dockerfile
```

Copy and paste:

```
FROM ubuntu:latest
MAINTAINER Support "support@crossbox.io"

RUN apt-get clean && apt-get update && apt-get install -y \
    locales \
    language-pack-fi  \
    language-pack-en && \
    export LANGUAGE=en_US.UTF-8 && \
    export LANG=en_US.UTF-8 && \
    export LC_ALL=en_US.UTF-8 && \
    locale-gen en_US.UTF-8 && \
    dpkg-reconfigure locales
ENV LANG en_US.UTF-8
ENV LANGUAGE en_US:en
ENV LC_ALL en_US.UTF-8
RUN apt-get install -y git wget net-tools
```

Run the command

```
docker build -t crossbox .
```

Start the container

```
docker run -t -d -p 25:25 -p 80:80 -p 110:110 -p 143:143 -p 443:443 -p 465:465 -p
587:587 -p 993:993 -p 995:995 -p 3478:3478 -p 5349:5349 -p 49152:65535 -p
49152:65535/udp -p 5000:5000 crossbox
```

Port mappings above are needed to forward the traffic from your host machine to the Docker container:

- 25 - SMTP Local Delivery
- 80 - CrossBox HTTP
- 110 - POP3
- 143 - IMAP
- 443 - CrossBox HTTPS
- 465 - SMTP Secure
- 587 - SMTP
- 993 - IMAP Secure
- 995 - POP3 Secure
- 3478 - TURN TCP and UDP
- 5349 - TURN Secure TCP and UDP
- 49152:65535 - Port range for TURN UDP RANGE
- 5000 - CrossBox Installer

> Note:
> If you plan to use non-default ports, you'll need to make sure that the ones

you enter in the command above match with the ones you choose during the CrossBox Installation. Also, these ports must not be already used at your host machine.

443:443 means proxy from port 443 on a host machine to a Docker container where CrossBox is also running on 443.

5000:5000 is needed for the installer and should be removed after you're done with the installation.

After you run the command, you'll get the long ID, something like:

```
88982f9169b45259d04eb5df211fff4c3dd57c8a1ea6b60d9020bf7b3bdab105
```

Use that ID in the command below to enter into the Docker container's command line

```
docker exec -it 88982f9169b45259d04eb5df211fff4c3dd57c8a1ea6b60d9020bf7b3bdab105
bash
```

Now when you've entered the shell, run the following commands:

```
cd /root
sh <(curl https://cdn.crossbox.io/install.sh || wget -O -
https://cdn.crossbox.io/install.sh)
```

The CrossBox Installer will now start, and you can open the given link in the browser to begin the installation.

Follow the instructions on how to set up CrossBox here: https://crossbox.io/documentation/page/quick-installation

After the installation is done type `exit` to exit the docker container.

Your CrossBox now runs in a container and you can start using it by visiting the URL you've configured during the installation.

To manage the CrossBox Docker container instance you need to enter the command line and you'll be able to run the crossbox command:

https://crossbox.io/documentation/page/command-line-tools

Clusters

Clusters

Creating a CrossBox Cluster

Intro

If you are a service provider, or if you ever plan to have more than one CrossBox server, then we advise you to think about creating a CrossBox Cluster. Here we

will explain how you should plan your deployment and what are the benefits of building a CrossBox Cluster.

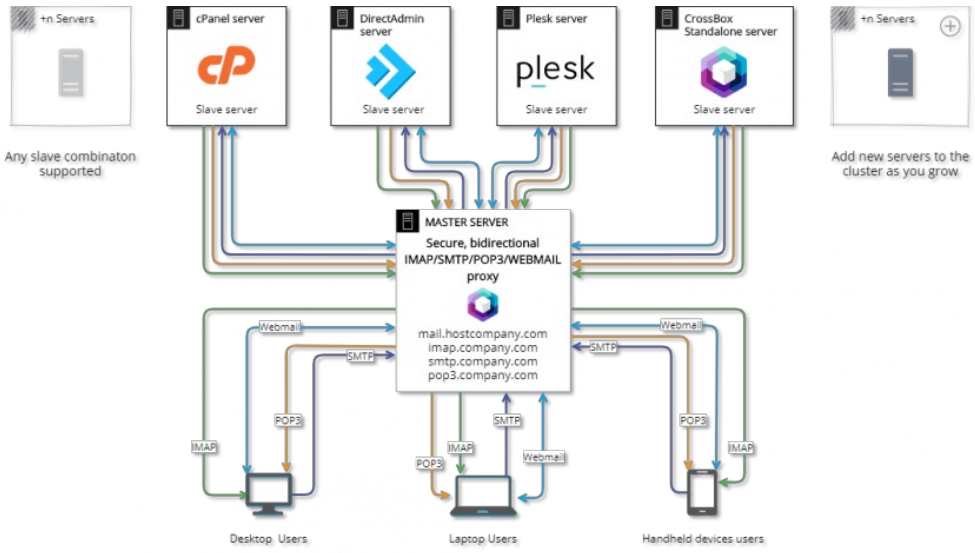
Benefits

- A single IMAP/SMTP/POP3 hostname for all users, no matter what specific slave server they are hosted on
- A single Webmail URL (CrossBox) for all users, no matter what specific slave server they are hosted on
- Unify a communication experience across different cPanel, Plesk, Direct Admin and Standalone servers within the cluster
- Scale indefinitely by simply adding more Slave and Master servers to the cluster
- Manage all servers from the Admin area

Master(s)-Slave(s) Topology

As you can see in the diagram below, the **Master** server creates an abstract around multiple **Slave** servers attached to it.

- You can have a single master server or multiple master servers to achieve load-balancing and high availability of the master server
- You can have a single or multiple slave servers
- You can attach any combination of DirectAdmin/cPanel/Plesk/Standalone servers to the cluster



Master Server

This server is used to resolve domains and email addresses to your slave servers (where they are actually hosted) and it acts as a single entry point for all your customers and users. For example, this is where **mail.your-hosting.com** would reside.

- Master server hosts the application's static files (html, css, js)
- Master servers runs MailProxify service that proxies IMAP/SMTP/POP3 traffic to the resolved slave server
- Master server can also run a HTTP proxy that proxies app's HTTP traffic to the slave servers (useful for users that are unable to access non-standard ports behind a corporate firewall)
- If the user's domain is using the master server in the MX record and the slave server is unreachable (offline), the master server will accept and queue any emails and deliver them once the slave server is reachable again

- The master server delegates all of the storage and computing to the slave server which is resolved for the email account
- Because there is no heavy computing happening on the master server itself, you can install it on a small dedicated server or a VPS/VM (Virtual Machine)
- In larger setups, the master server can be load-balanced and made fully redundant by creating a multi-master setup

Slave Server(s)

These servers hold all your customer's CrossBox data (emails, files, databases, etc) and do all the computing related to the activity of CrossBox accounts that are hosted there.

You can install a slave server on

- DirectAdmin
- cPanel
- Plesk
- or a Standalone server

Building a CrossBox Cluster

First, install a Master server, log in to the Admin area and then add Slave server(s).

Installing a Master server

Pick a small dedicated server or a VPS and install CrossBox as explained in a Quick Start Guide, making sure that you **choose "Master" as a server type** during the installation.

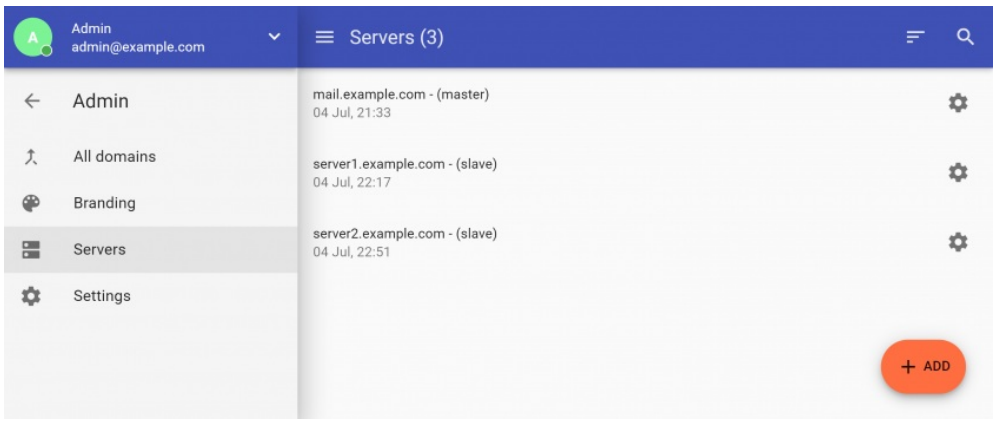
Adding a server to the cluster

You can attach two types of servers to your cluster:

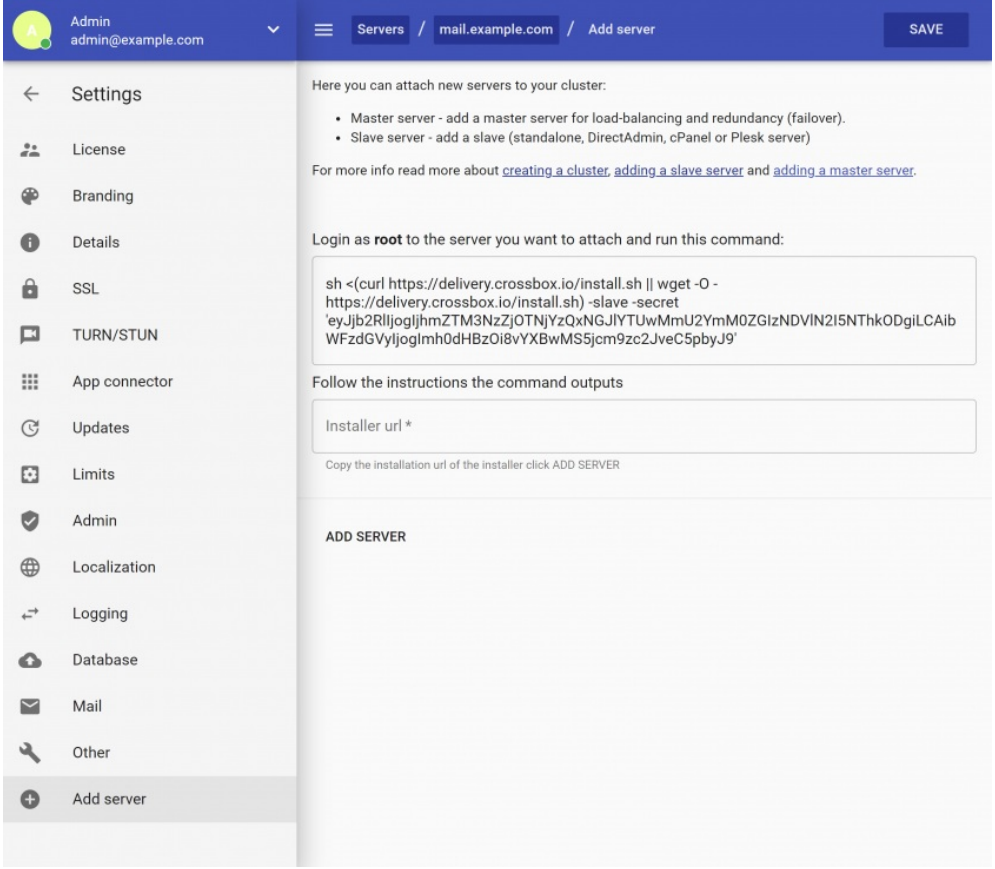
- Slave server (DirectAdmin, cPanel, Plesk, Standalone)
- Additional master server(s) for load-blancing and redundancy

To connect a new server to your cluster, log in as admin user and go to the "*Servers*" area.

The list of all your CrossBox servers will appear. In the bottom right corner click on the **+ ADD** fab button.



Here you will be instructed to log in as root to the new server's SSH so you can issue the installation command:



Now just copy the URL that the SSH command outputs, paste it back to the "*Installer URL*" field and click the "*Add Server*" button.

This will start the installation of CrossBox on the targeted server and you'll be able to complete the installation from the Admin area.

> ⚠ - If you're adding **a slave server**, make sure that you choose **"Server" license type** during the installation
> - If you're adding **a master server for load-balancing and reduncancy (failover)**, make sure that you choose **"Master Server" license type** during the installation

You can tweak all the CrossBox Options during the installation to customize the settings for the new server.
After the installation is completed, the added server will be automatically connected and made part of your CrossBox Cluster.

Creating a Multi-master Setup

Benefits:

- Make your master server fully redundant
- Round robin master server's traffic across multiple servers

The process of creating a multi-master setup is the same as with adding a new server to the cluster - just make sure that you choose "Master Server" license type during the installation.

After the installation is completed, add a new IP to your DNS. For example, if your master server hostname is mail.your-hosting.com, this is how your A records should look like:

mail.your-hosting.com IN A 1.1.1.1 (where 1.1.1.1 is the IP address of your first

master server)
mail.your-hosting.com IN A 1.1.1.2 (where 1.1.1.2 is the IP address of your second master server)
mail.your-hosting.com IN A 1.1.1.3 (where 1.1.1.3 is the IP address of your third master server)

There is no limit on the number of master servers you can have within a cluster.

Third Party Apps

Outlook

Autodiscover

Outlook can auto-detect IMAP/SMTP/POP3 settings by looking at the domain's "autodiscover" DNS record.

SRV record

A typical SRV record looks like this:

- Service: _autodiscover
- Protocol: ._tcp
- Port Number: 443
- Host: host.name.com
- Priority: 0
- Weight: 0

BIND example:

Domain: _autodiscover._tcp
TTL: 14440
Record Type: SRV
Record: 0 0 443 host.name.com.

CNAME record

A typical CNAME record looks like this:

- Name: autodiscover
- TTL: 3600
- RR Type: CNAME
- Target: host.name.com

BIND example:

Domain: autodiscover
TTL: 14440
Record Type: CNAME
Record: host.name.com.

Please make sure to add either CNAME or SRV record in your domain's DNS

zone to have the autodiscover feature in Outlook working.

CrossBox Standalone version exposes the autodiscover config for Outlook clients at:

https://host.name.com/**autodiscover/autodiscover.xml**

host.name.com is the hostname of the server where CrossBox is running.

Manual Configuration

If you don't want to use the autodiscover feature, you can still configure IMAP/SMTP/POP3 settings in Outlook manually.

Server Information

Account Type: IMAP
Incoming Mail Server: host.name.com
Outgoing Mail Server (SMTP): host.name.com

Logon Information

Username: email@example.com

Advanced Settings

Incoming

IMAP

Server Port Numbers

Recommended

Incoming Server (IMAP): 993
Type of connection: SSL

Optional

Incoming Server (IMAP): 143
Type of connection: TLS or None

POP3

Server Port Numbers

Recommended

Incoming Server (POP3): 995
Type of connection: SSL

Optional

Incoming Server (POP3): 110

Type of connection: TLS or None

Outgoing

SMTP

Server Port Numbers

Recommended

Outgoing Server (SMTP): 465
Type of connection: SSL

Optional

Outgoing Server (SMTP): 587
Type of connection: TLS or None (TLS for standalone and None for Cluster)

Third Party Apps

Thunderbird

Autoconfig

Thunderbird can auto-detect IMAP/SMTP/POP3 settings by looking at the domain's "autoconfig" DNS record.

CNAME record

A typical CNAME record looks like this:

- Name: autoconfig
- TTL: 3600
- RR Type: CNAME
- Target: host.name.com

BIND example:

Domain: autoconfig
TTL: 14440
Record Type: CNAME
Record: host.name.com.

> Please make sure to add CNAME record in your domain's DNS zone to have the autoconfig feature in Thunderbird working.

CrossBox Standalone version exposes the autoconfig for Thunderbird clients at:

https://host.name.com/**mail/config-v1.1.xml**

> host.name.com is the hostname of the server where CrossBox is running.

Manual Configuration

If you don't want to use the autoconfig feature, you can still configure IMAP/SMTP/POP3 settings in Thunderbird manually.

Server Information

Incoming Mail Server: host.name.com
Outgoing Mail Server (SMTP): host.name.com

Logon Information

Username: email@example.com

Advanced Settings

Incoming

IMAP

Recommended

Port: 993
SSL: SSL/TLS
Authentication: Normal password

Optional

Port: 143
SSL: STARTTLS or NONE
Authentication: Normal password

POP3

Recommended

Port: 995
SSL: SSL/TLS
Authentication: Normal password

Optional

Port: 110
SSL: STARTTLS or NONE
Authentication: Normal password

Outgoing

SMTP

Recommended

Port: 465
SSL: SSL/TLS
Authentication: Normal password

Optional

Port: 587
SSL: STARTTLS or NONE (STARTTLS for standalone and None for Cluster)
Authentication: Normal password

Third Party Apps

Other Apps

Building Android and iOS apps

Building Android and iOS apps

Online Apps Builder

Intro

Online Apps Builder is an online tool that you can use to generate and download your own white-labeled Android and iOS CrossBox apps.

The process is entirely automated and requires no technical knowledge.

The system will generate Android and iOS apps based on the branding profile that is active on your server. This includes the application name, logo (icon), theme, loading animation, the default language, etc.

At the end of the process, you'll have both apps ready for publishing under your name to Google Play and Apple App Store.

Here are some examples of app builts with the Online Apps Builder:

Android:

- https://play.google.com/store/apps/details?id=com.azymcloud.cloudmail
- https://play.google.com/store/apps/details?id=com.mxroute.crossbox
- https://play.google.com/store/apps/details?id=com.bulutmail.suite

iOS:

- https://apps.apple.com/app/id1522254361
- https://apps.apple.com/app/id1490617582
- https://apps.apple.com/app/id1494638876

Benefits

End-users:

- Easy to get started on any Android or iOS powered device with one-click app installation (via Google Play and Apple App Store)
- No complicated IMAP/SMTP/POP3 configuration, only email address, and password are required
- Instant access to
  - Email
  - Chat
  - Audio/video calls
  - Conferences
  - Files
  - Contacts
  - And many other features
- Real-time push notifications
- Updates automatically

Service Providers:

- Provide Android/iOS users with seamless access to email, chat, calls and conferences, files, and contacts
- Easier customer onboarding
- Better brand exposure with custom Android/iOS apps
- Apps work for all nodes/slaves in the cluster

Requirements

To build smartphone apps you need to have:

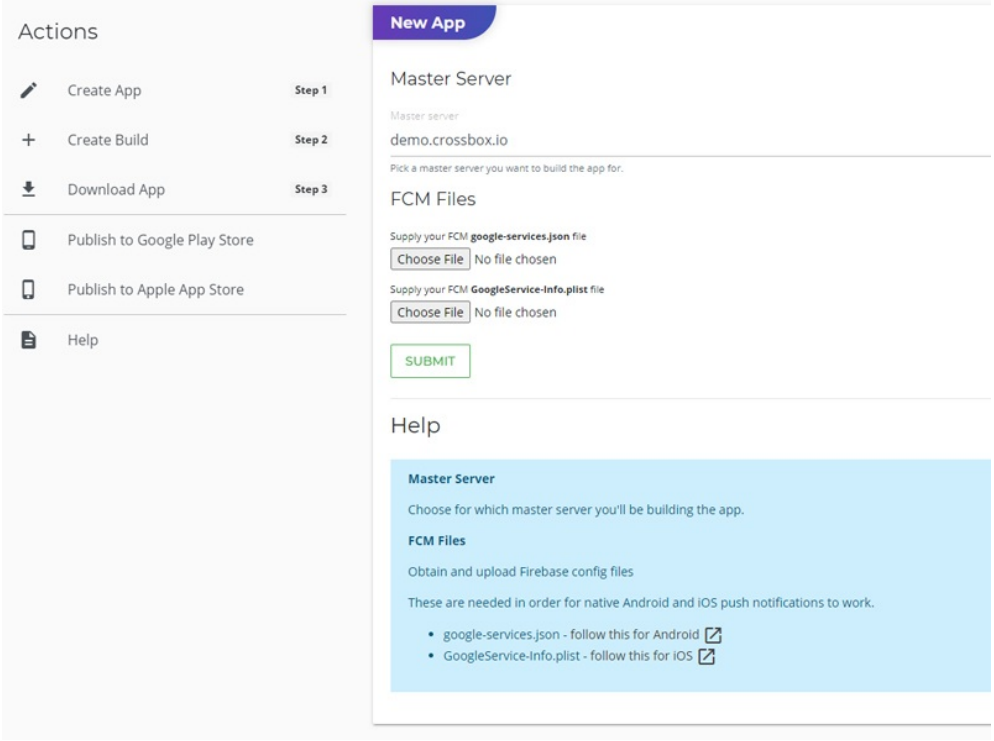
- An active CrossBox Cluster
- An active Branding (White-label) Addon

Building

First, log into your crossbox.io account and navigate to Online App Builder.

Create an App

Click on the Create App menu item. You'll see the form as shown in the screenshot:



Here, you'll need to fill in:

Master Server

Choose for which master server you'll be building the app.

The app builder will pull the branding profile (app name, logo, theme, etc) from this server to use for generating the apps.

Obtain and upload Firebase config files

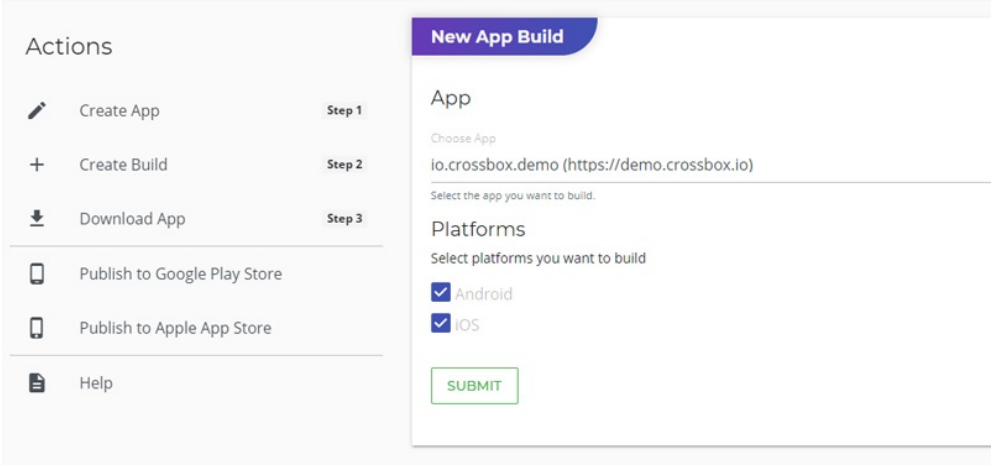
These are needed in order for native Android and iOS push notifications to work.

- google-services.json - follow this for Android
- GoogleService-Info.plist - follow this for iOS

Now that both files are on your computer, attach them to the form and click the Submit button.

Submit Build Request

After you've finished creating the app, you can proceed to issue a new build request by clicking the Create Build button:



In the form shown above, choose the app you created in the previous step and the platforms that you want to build.

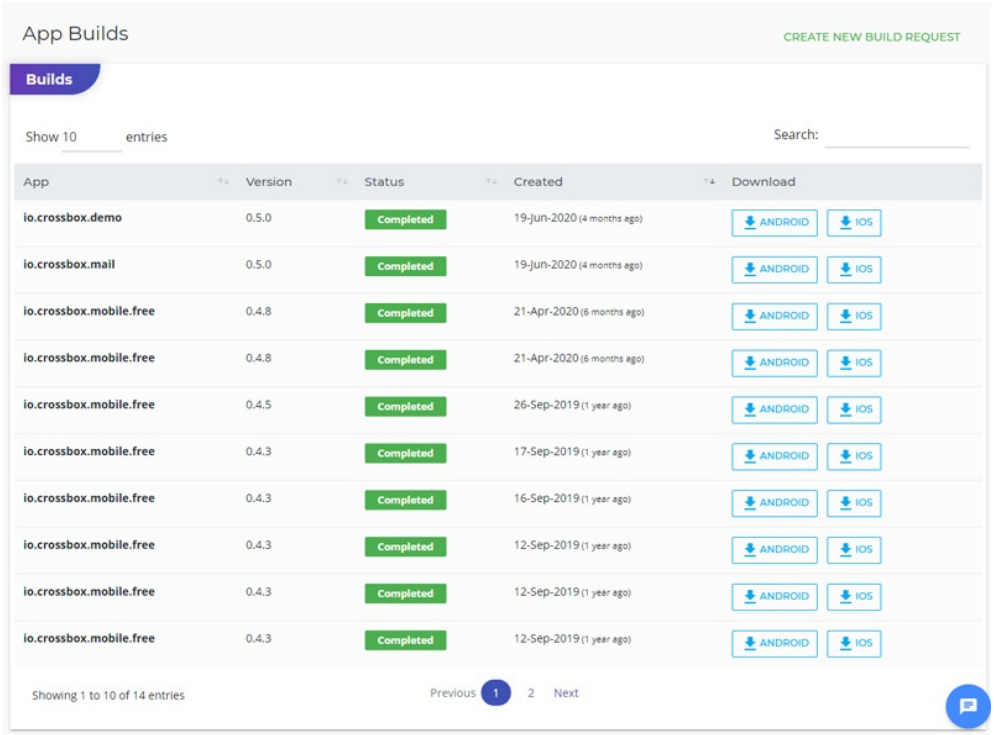
Click the submit button.

You'll get an email notification after the build process is completed.

Download the Apps

Now that the build process completed successfully, you can log in to your crossbox.io account and navigate to the Download App page.

Here you can download **android.zip** and **ios.zip** archives to your computer:



These archives contain both debug and release versions:

- The debug version can be installed onto the devices without signing and is used for testing the app on a device prior to publishing the app to the app stores.
- The release version is a production-ready version which you need to sign and then publish to the respective app stores.

Publish to Stores

- Sign and publish an Android app
- Sign and publish an iOS app

Building Android and iOS apps

Android

Building Requirements

To build an Android app:

- Windows/Linux/macOS
- Google Play Developer Account
- Keytool
- Jarsigner

Download and Install JDK

If you don't already have JDK installed, download and install from the official JDK downloads page.

Generate Keystore

Generate a Keystore file (if you already don't have one).

Use the `keytool` provided by the JDK:

```
keytool -genkey -v -keystore my.keystore -keyalg RSA -keysize 2048 -validity 10000
-alias app
```

> Make sure to save the generated Keystore file and its password somewhere safe. You'll need them again if you ever decide to republish the app.

Sign

The last step is to sign the app with the previously generated Keystore. This is done using the `jarsigner` tool provided by JDK.

The Jarsigner tool can be found in:

- Linux and MacOS: /usr/bin/jarsigner
- Windows: jdk/bin/jarsigner.exe

Use it like this:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore my.keystore
app.aab app
```

The signed ABB can be verified with:

```
jarsigner -verify app.abb
```

Publish App to the Google Play Store

Submit the Android app via the Play Store.

Building Android and iOS apps

iOS

Building Requirements

To build an iOS app:

- macOS latest version
- Xcode latest version
- Apple Developer Account
- Cocoapods

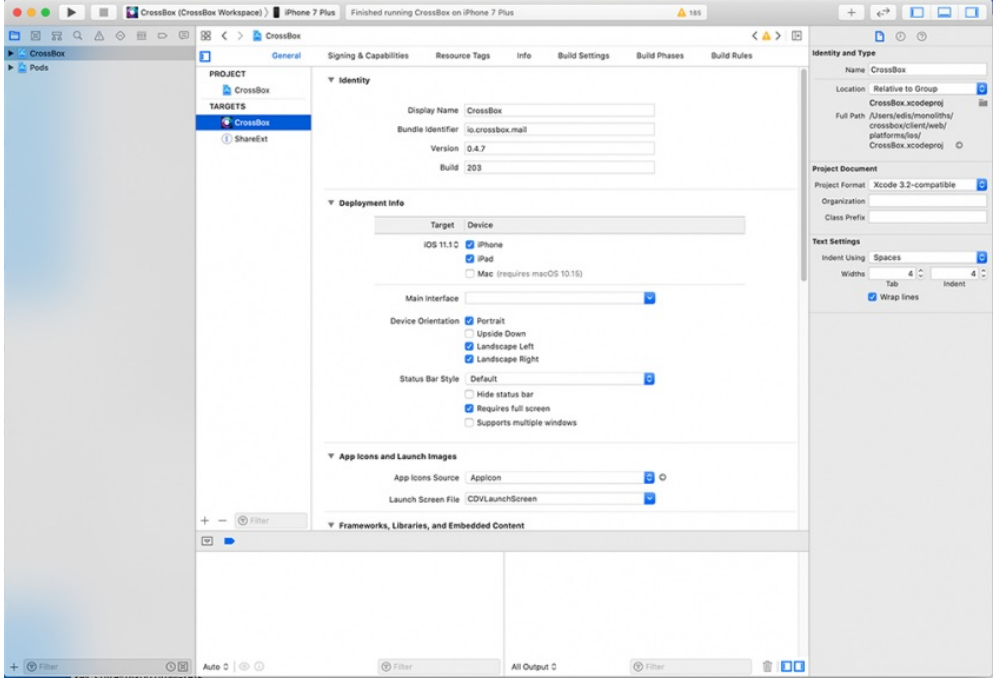
Open the project

Make sure you open the **.xcworkspace** and **not** .xcodeproj

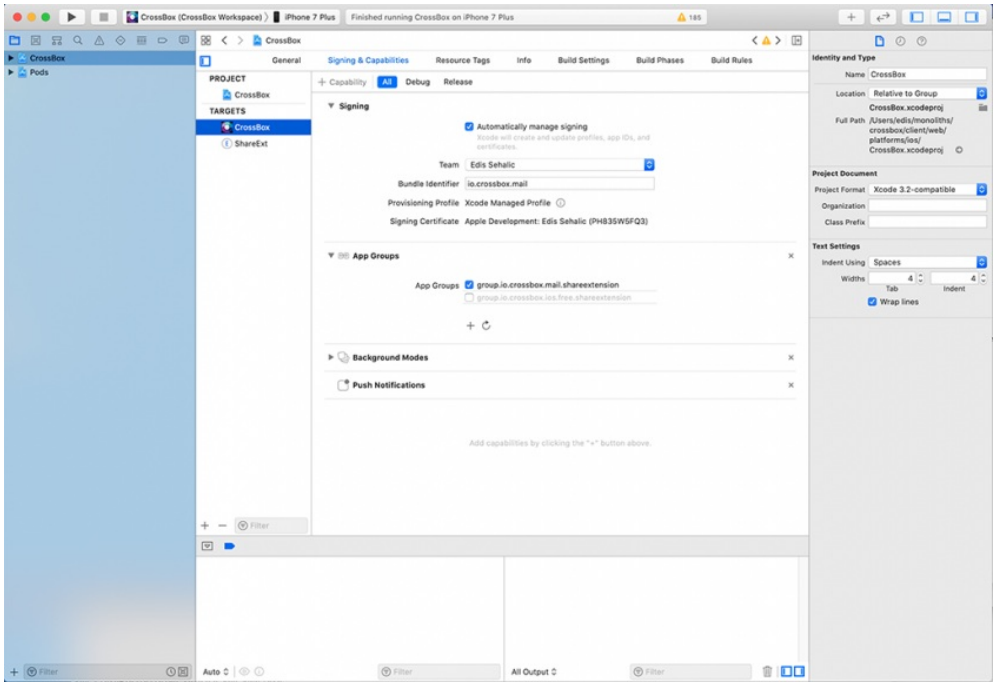


Certificates and capabilities

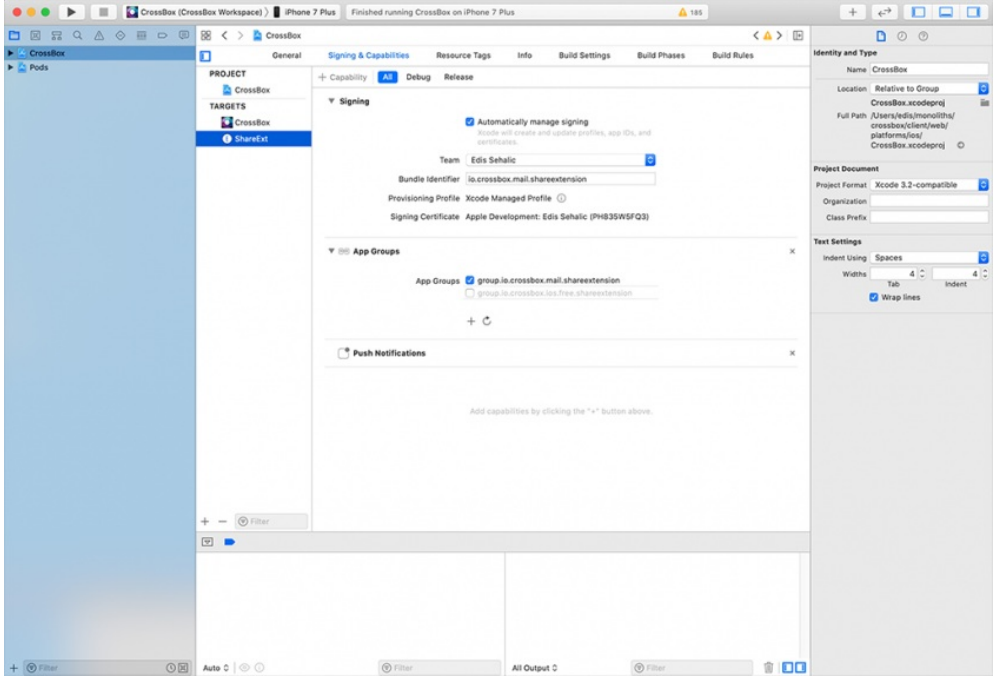
- Click on the First Project in the left side nav
- Go to "**Signing and Capabilities**"
- Add Capability "**Push notifications**"
- Add Capability "**App Groups**"
- **Turn off** automatic signing for both Debug and Release
- Then **turn on** automatic signing for both Debug and Release and select your developer account
- Add a group with the name **group.my.identifier.shareextension** (replace my.identifier with com.yourdomain.app):



- Click on the First Project in the left side nav
- Go to "**Signing and Capabilities**":

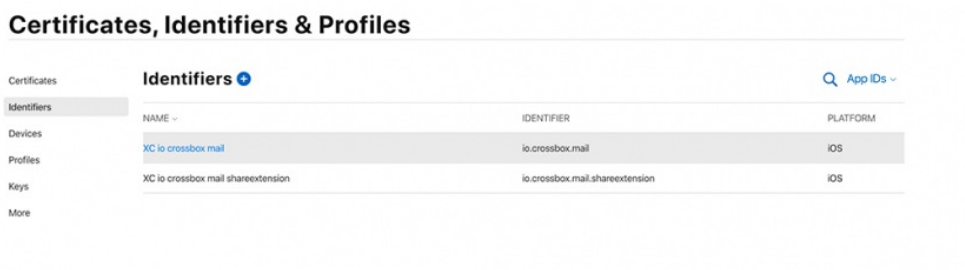


- Add Capability "**App Groups**"
- **Turn off** automatic signing for both Debug and Release
- Then **turn on** automatic signing for both Debug and Release and select your developer account
- Add a group with the name **group.my.identifier.shareextension** (replace my.identifier with com.yourdomain.app):

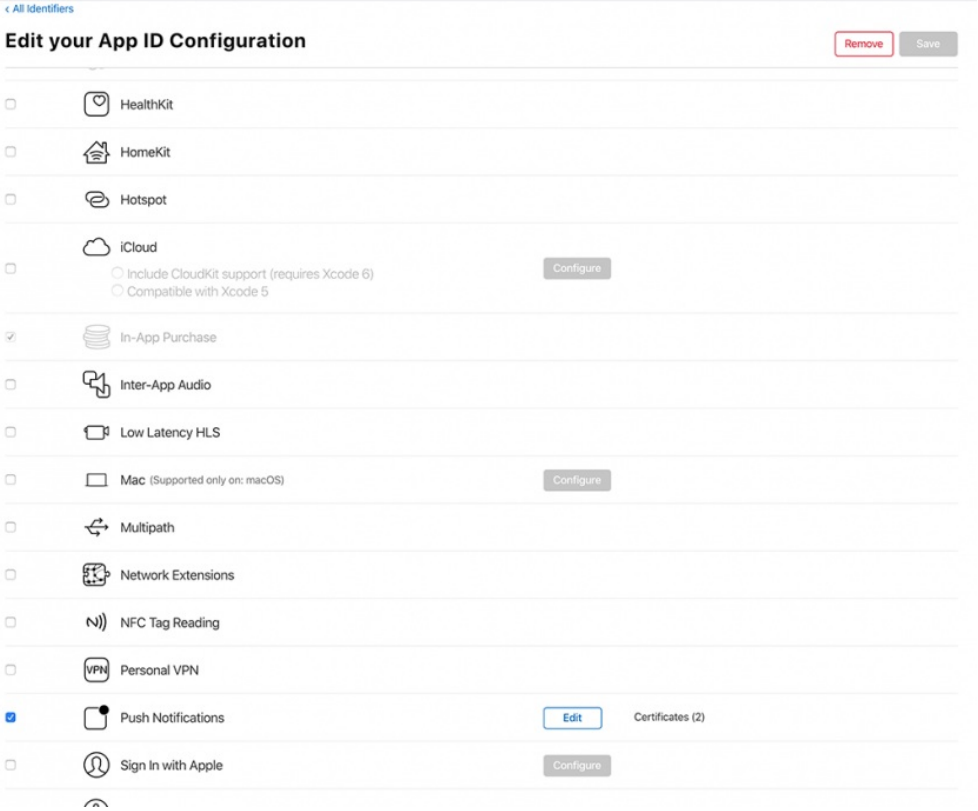


Push Notifications Certificates

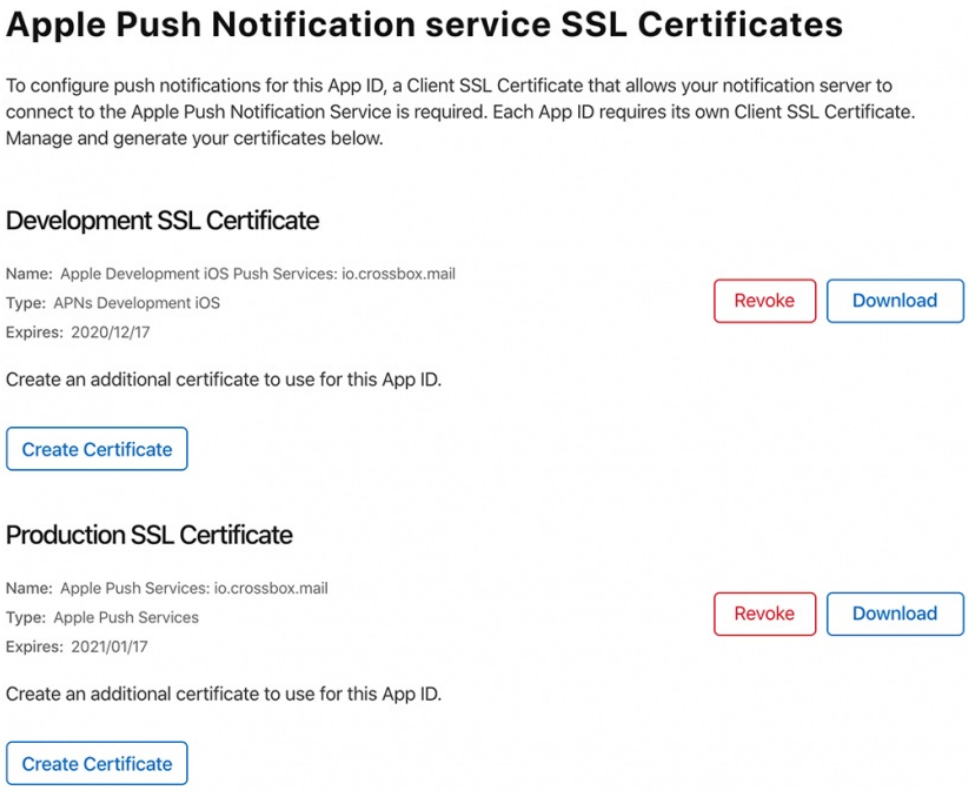
- Go to https://developer.apple.com/ and go to **Identifiers**



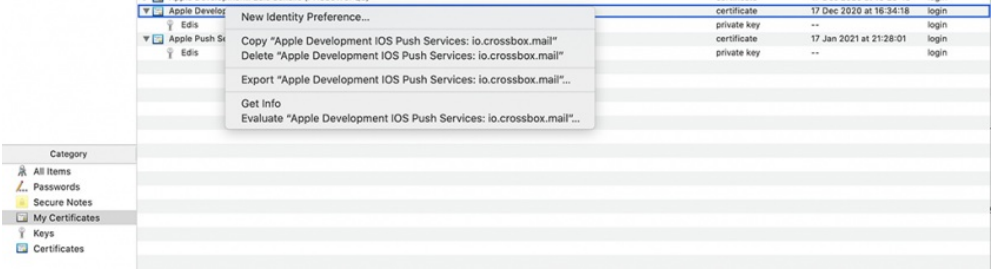
- Select your app identifier and find **Push notifications** on the list:



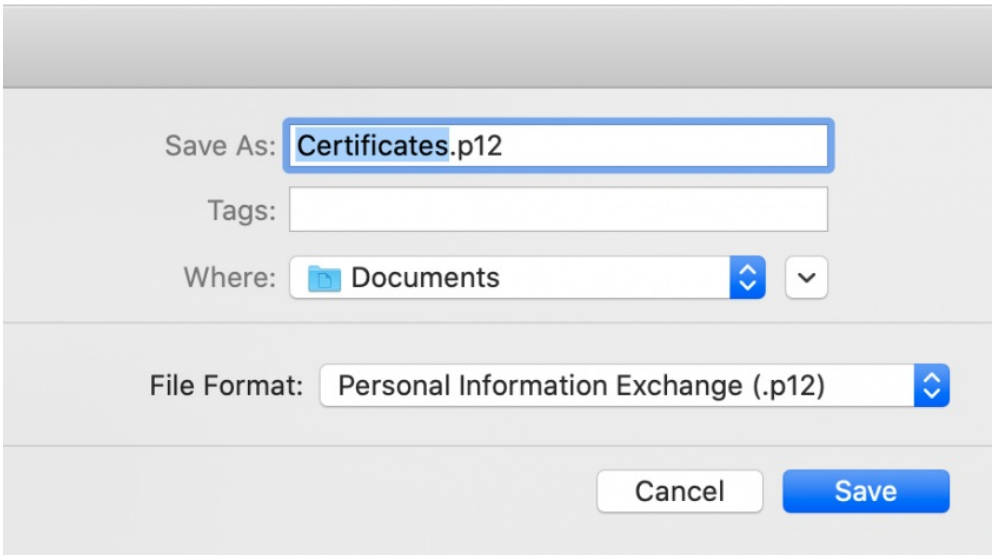
- Create both certificates and **download them**:



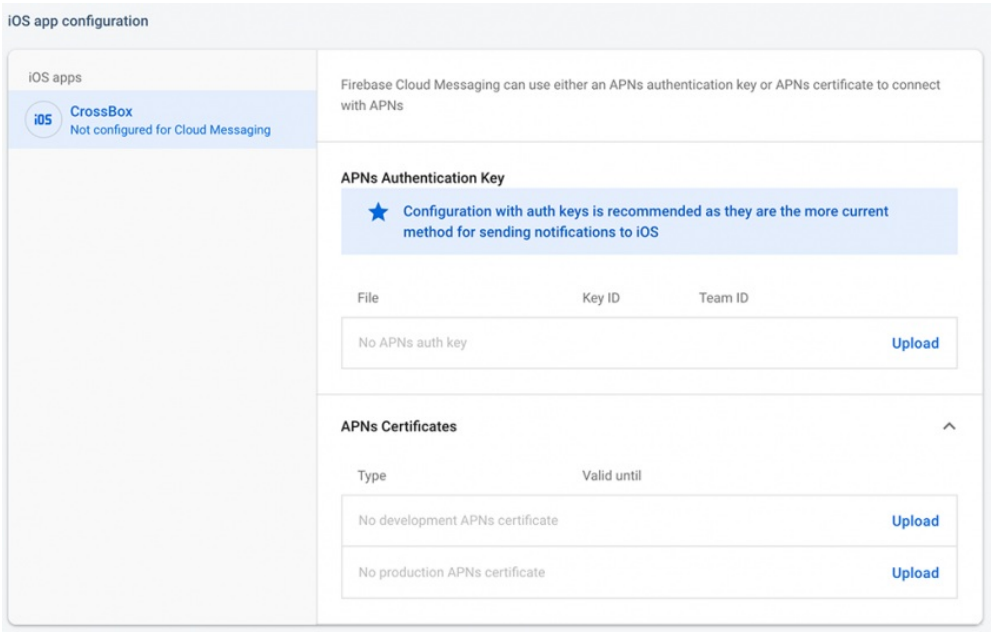
- Double click on the downloaded certificates so they are loaded in the Keychain Access
- Open **Keychain Access**
- Go to **My Certificates**
- Find both certificates:



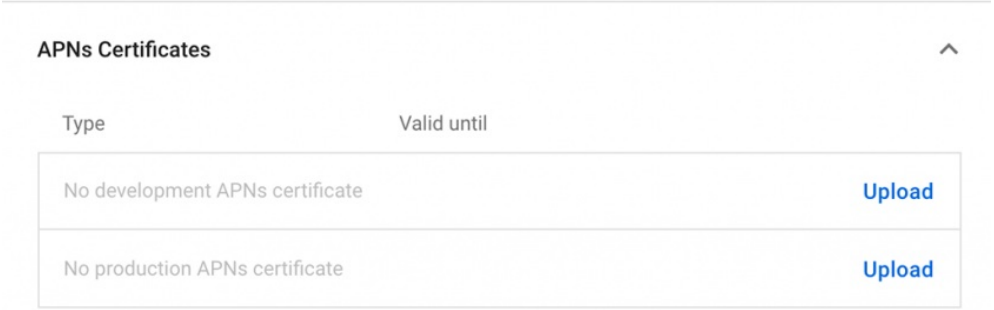
- Click **Export** and choose **.p12** as file format:

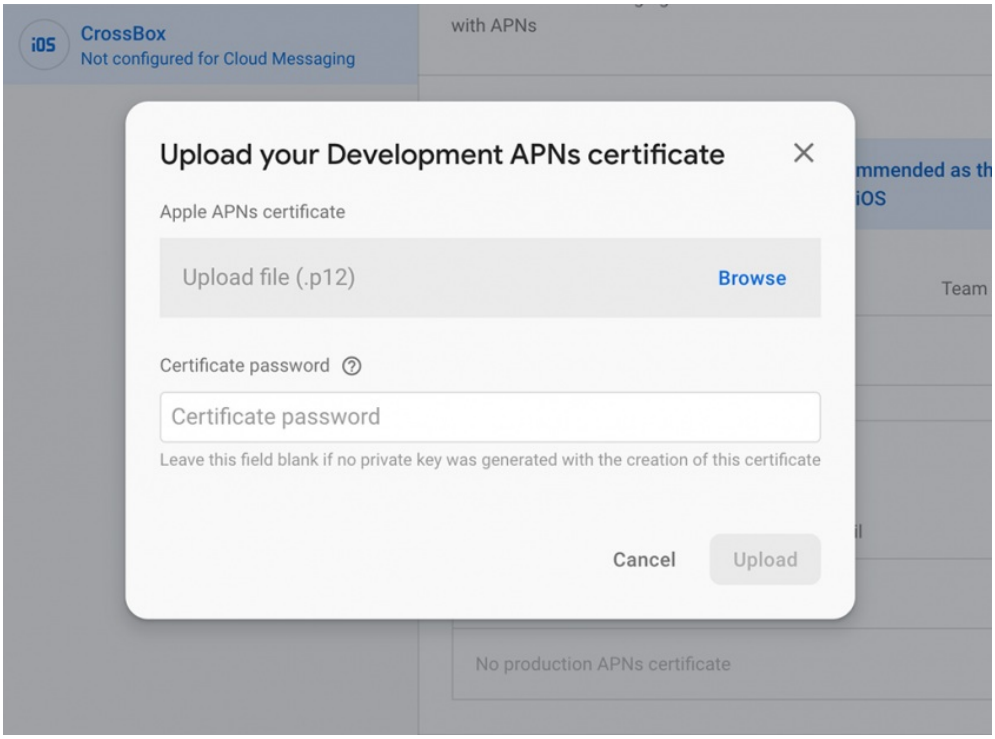


- Make sure you know which certificate is for Development and Production
- Go to Firebase console and go to the project you previously created.
- Open **Cloud Settings**:



- **Upload Production and Development certificates** accordingly:





Submission and Testing

- Run the project on the simulator and try logging in
- Use the app and test out if the project has been properly configured
- Open Terminal.app and navigate to the project folder with "cd" command
- Run the command:

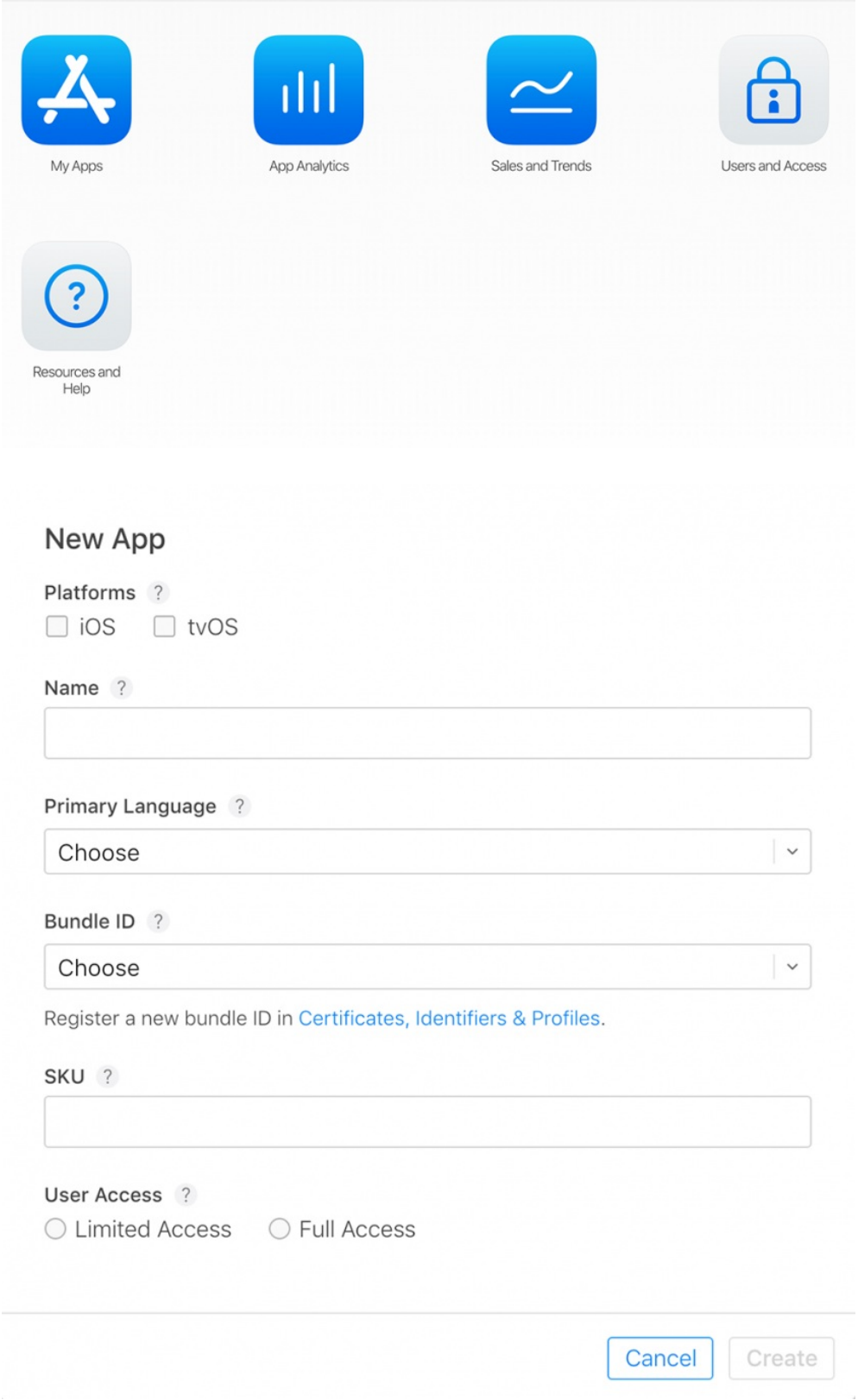
```
sh prepare-production.sh
```

This script prepares the needed libraries for production. You will no longer be able to run the app in the simulator/test device unless you run:

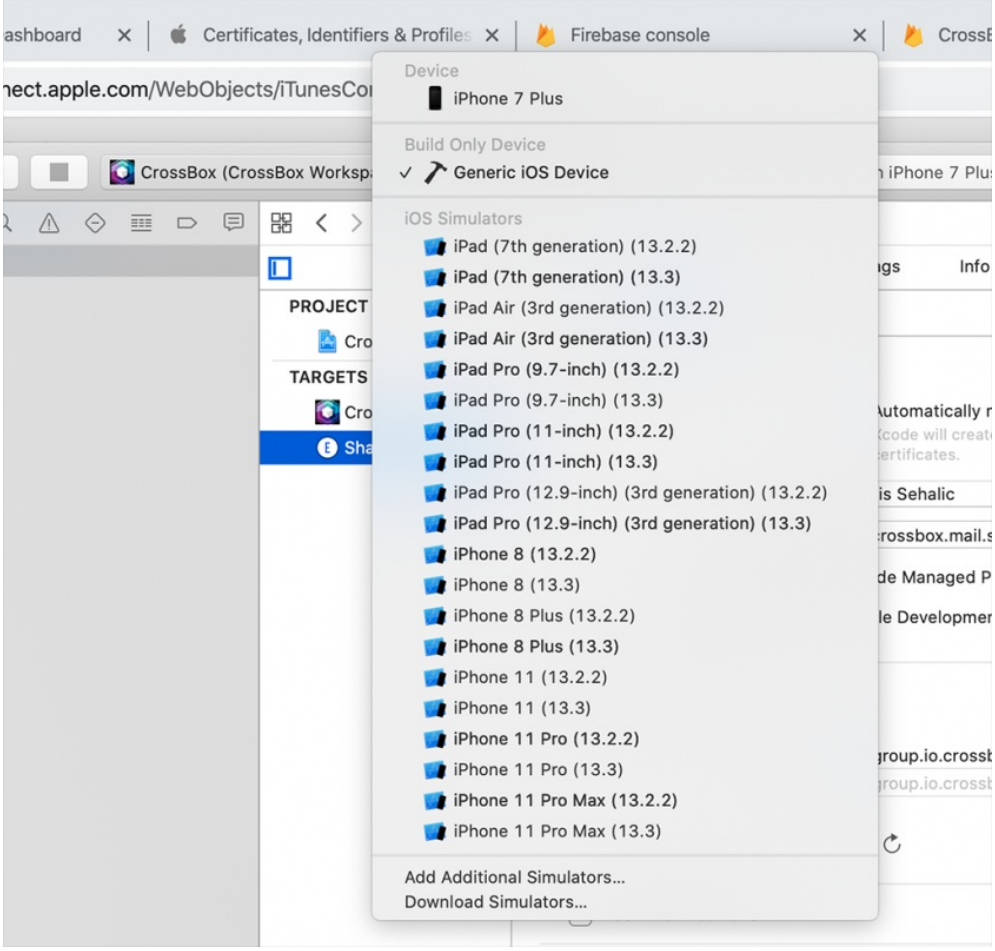
```
sh prepare-test.sh
```

to revert back libraries.

- Go to https://appstoreconnect.apple.com/ and create a new app:



- Go to XCode and select target build "**Generic iOS Device**"



Distribution

- Make sure you have **distribution** certificates for your account (you can check if you have them here: https://developer.apple.com/account/resources/certificates/list)

- If the list is empty, create one:

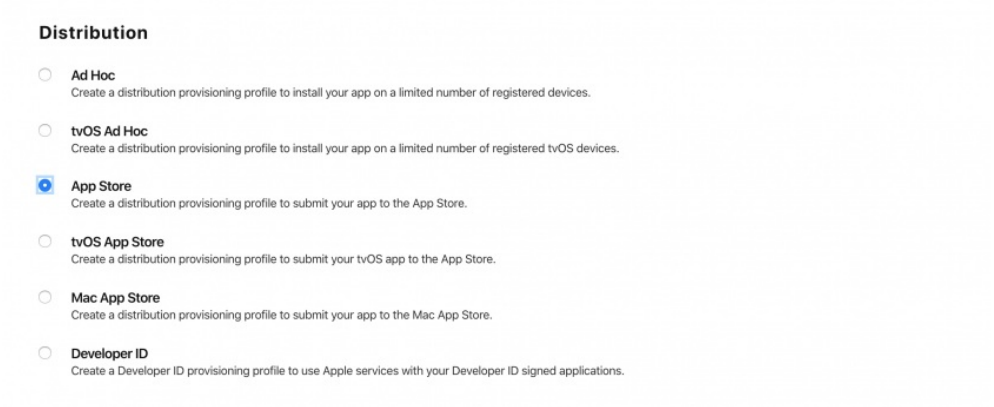


- Pick "**Apple Distribution**"



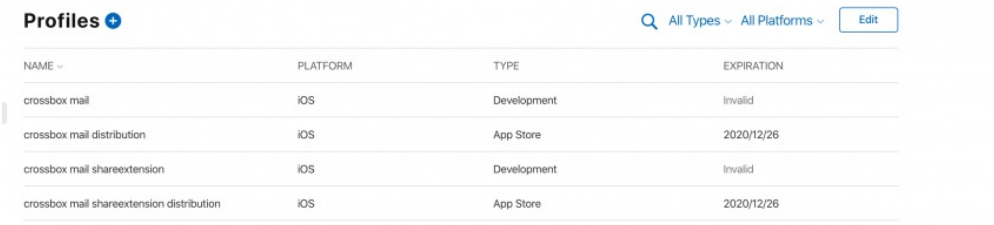
- Then, go to https://developer.apple.com/account/resources/profiles/list

Click + and add the profile for your identifiers your_app_id.shareextension and your_app_id for the app store:

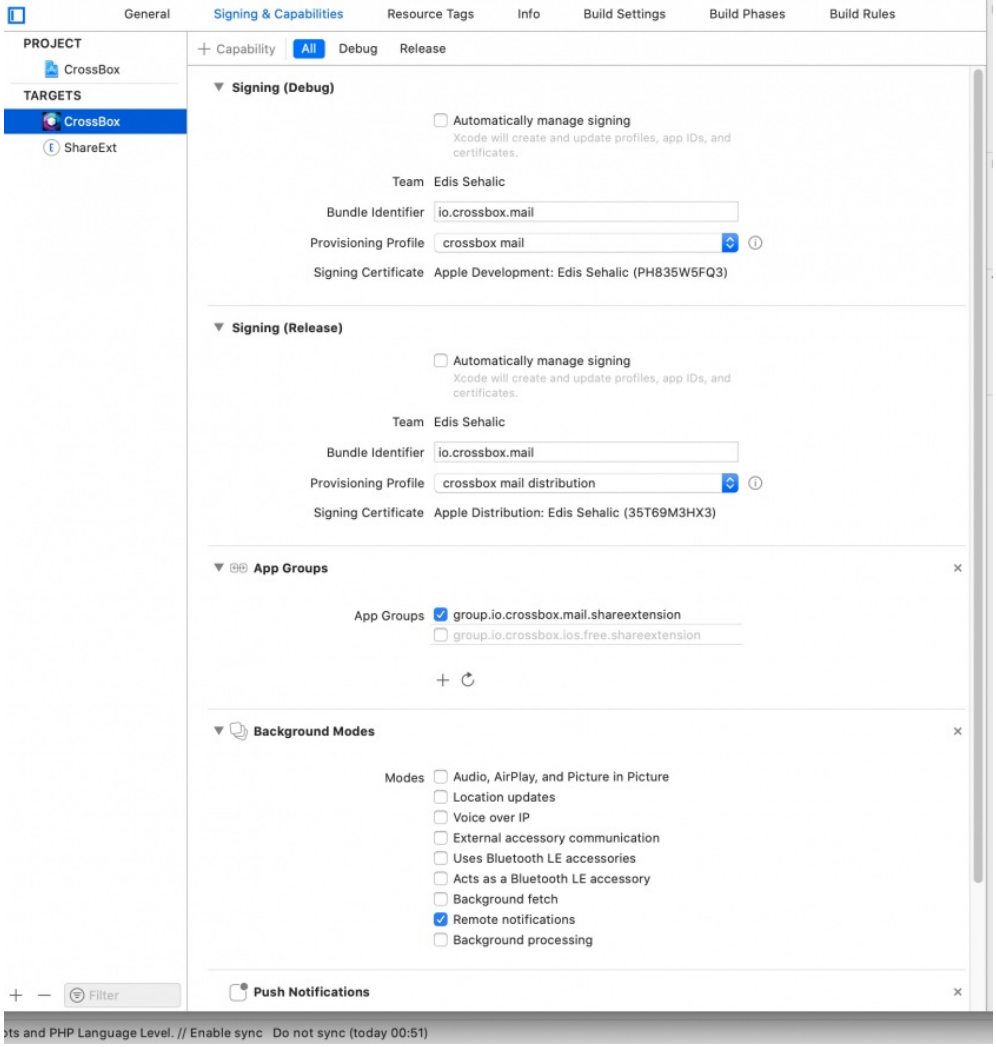


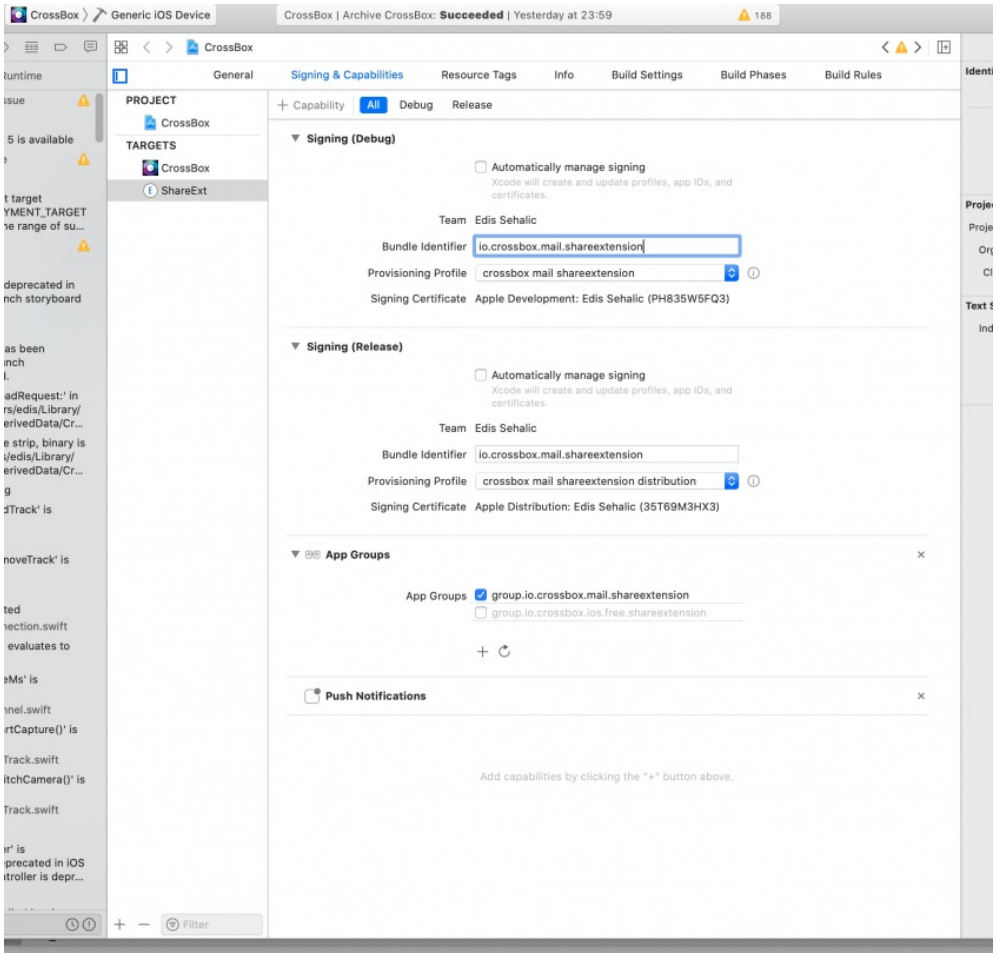
- Name your profile so you differentiate it easily by adding a suffix "distribution"
- Click continue to finish the wizard
- Make sure you have profiles for both identifiers

Once you have this list of profiles you can proceed with setting up the signing settings



You will have to configure your **Debug** and **Release** profiles to match. They should look like this:



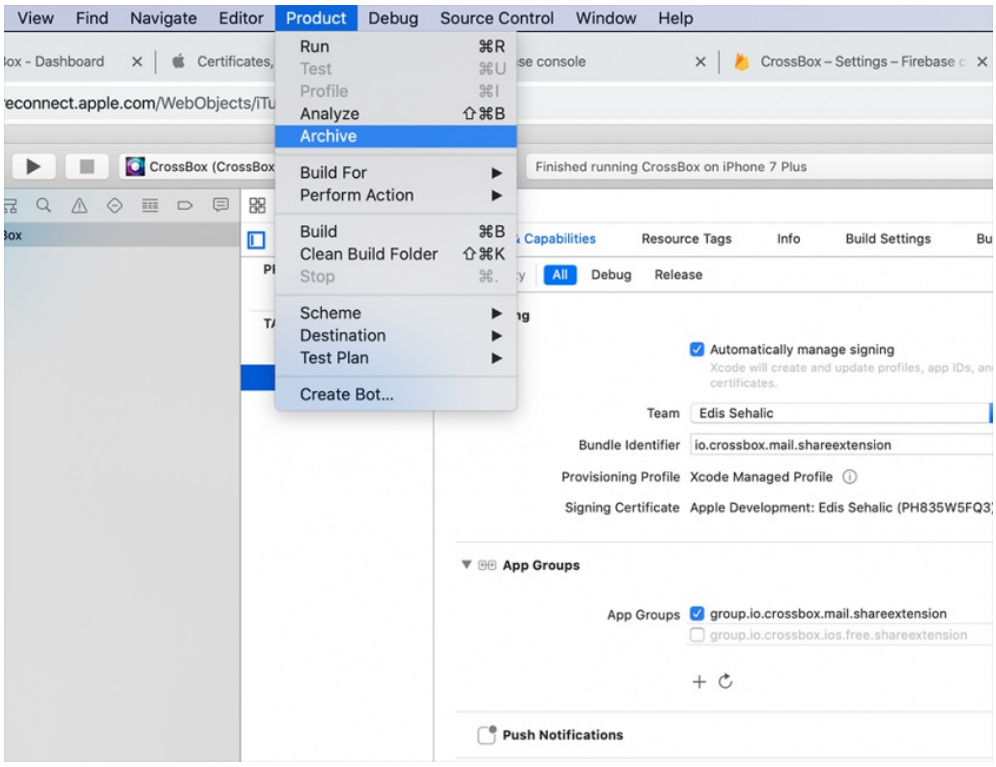


Notice the Signing certificate for Debug and Release: Debug(Apple **Development**) Release(Apple **Distribution**)

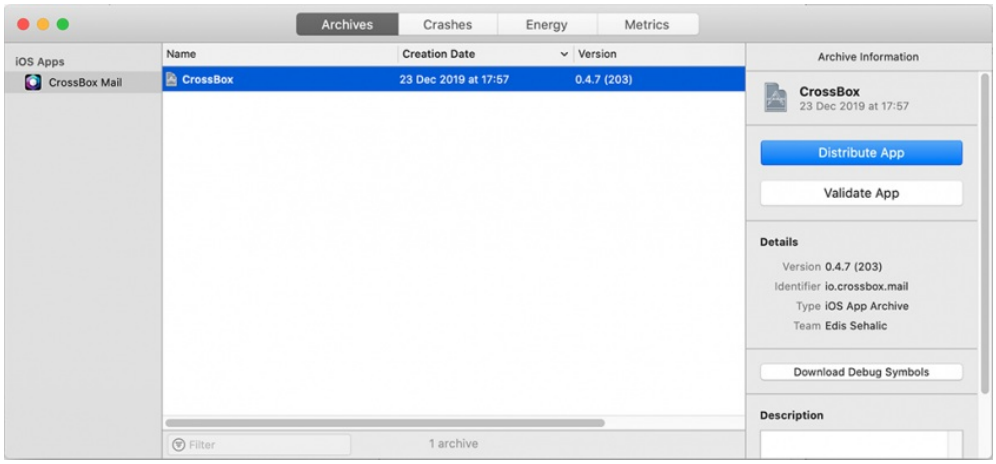
Both targets must have proper certificates configured

XCode will complain if the certificates are not properly configured. If you need help with that, you can contact us via Tickets or live Chat.

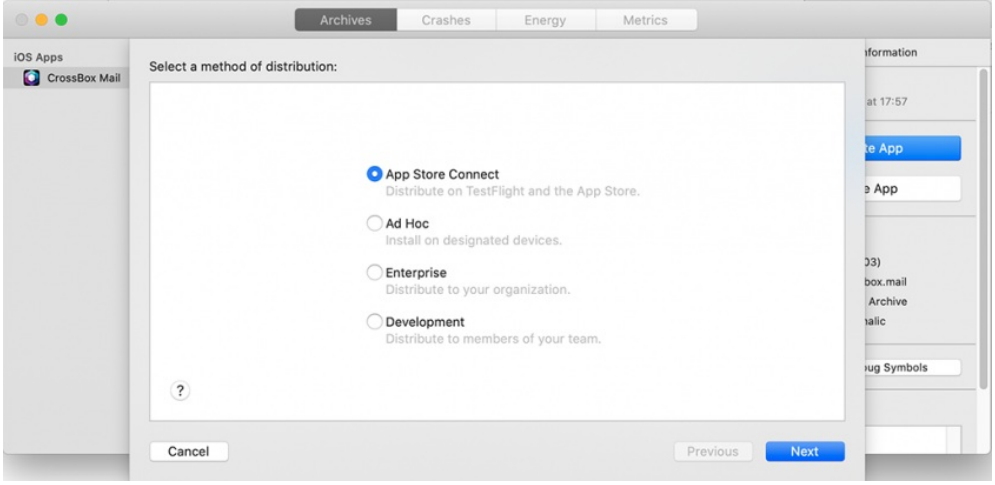
Then go to **Product > Archive**

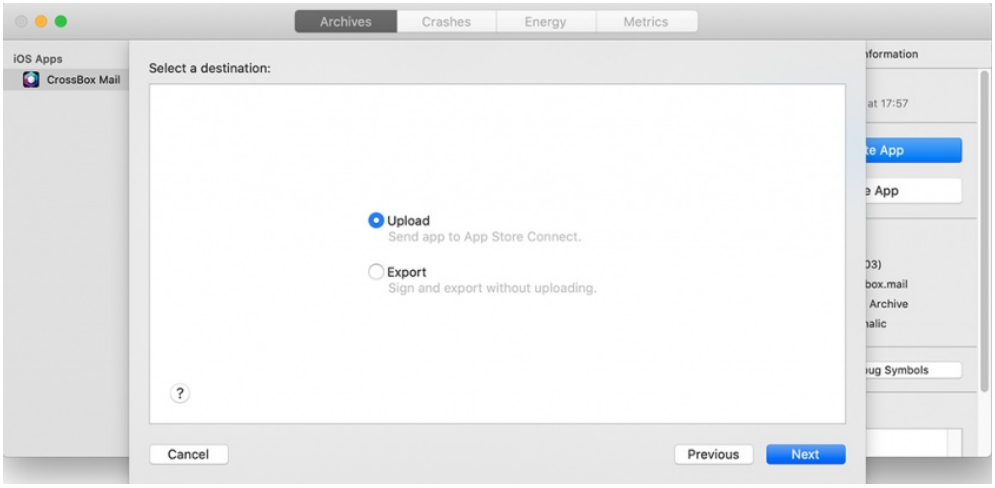


- Click **Distribute App**

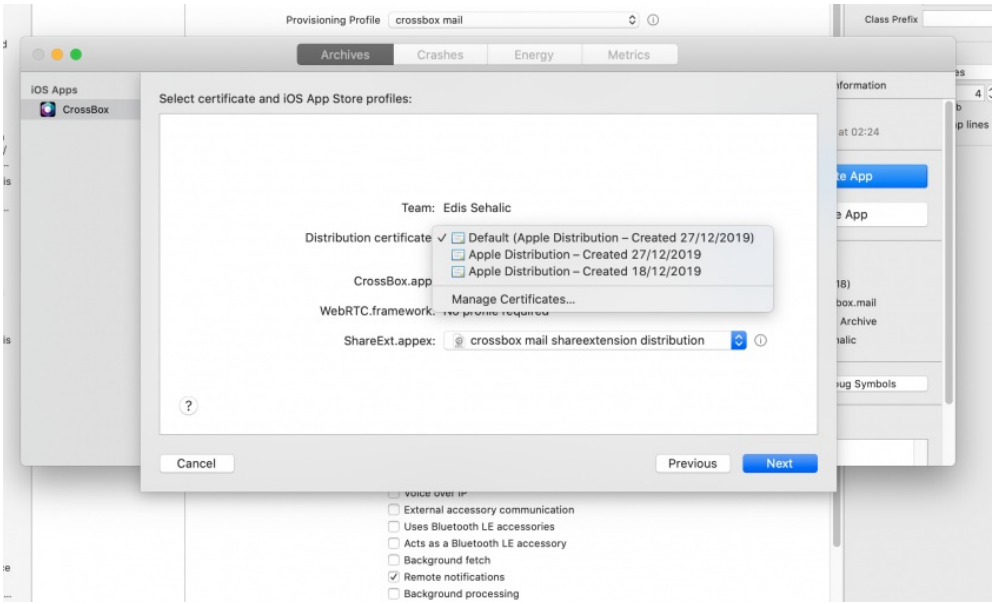


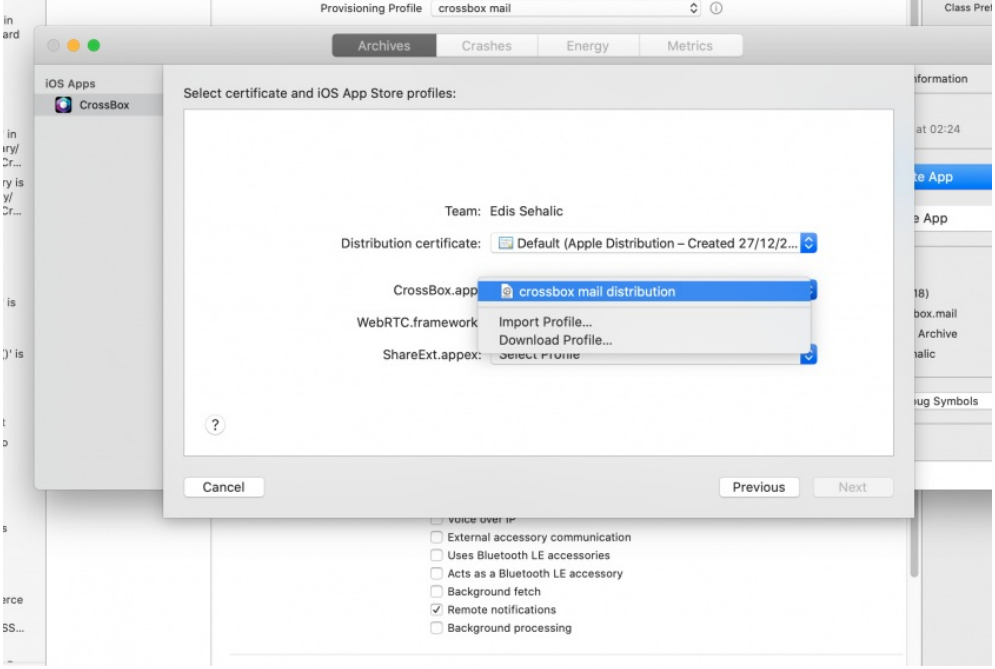
- The app will be validated and uploaded to https://appstoreconnect.apple.com/

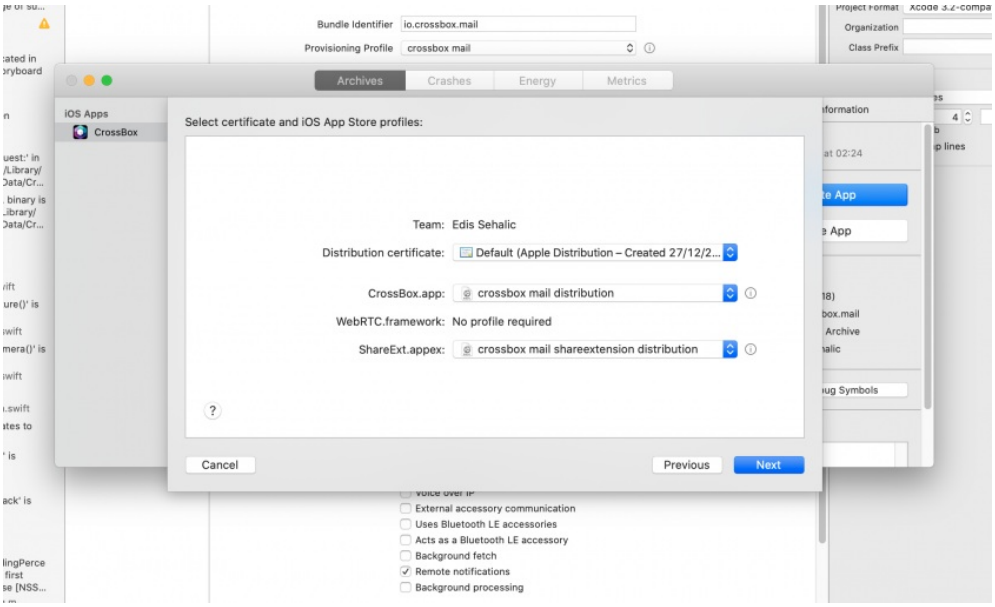




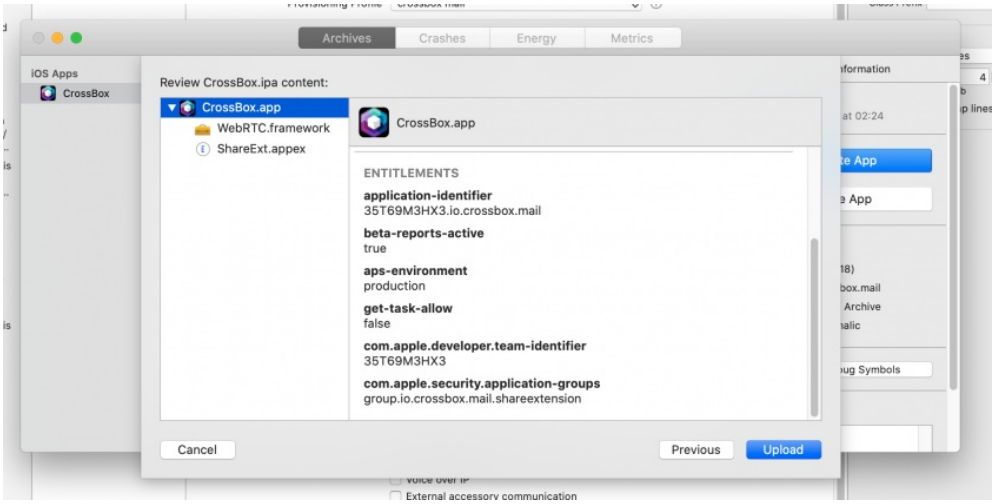
After this step, you will be asked to confirm your certificates again, please select the ones you set up initially:

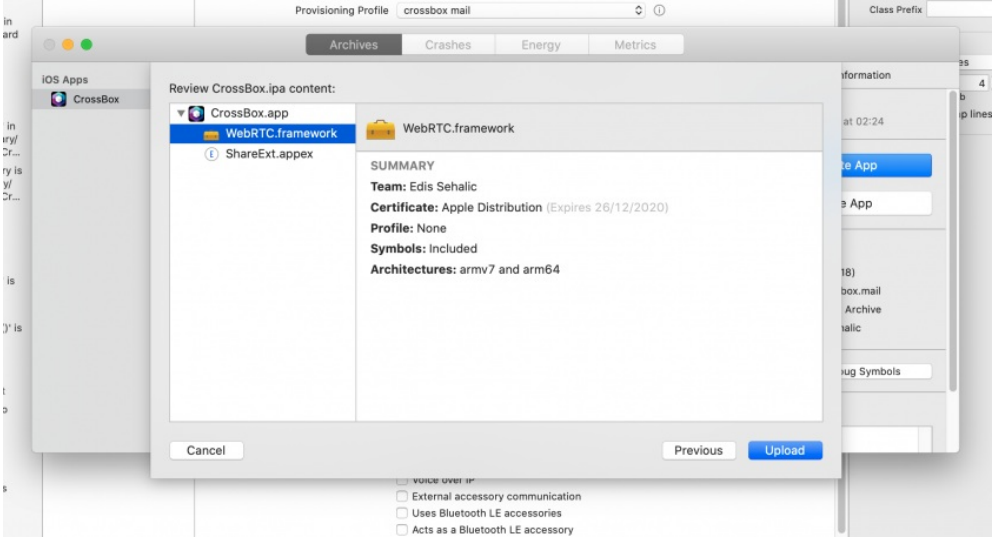


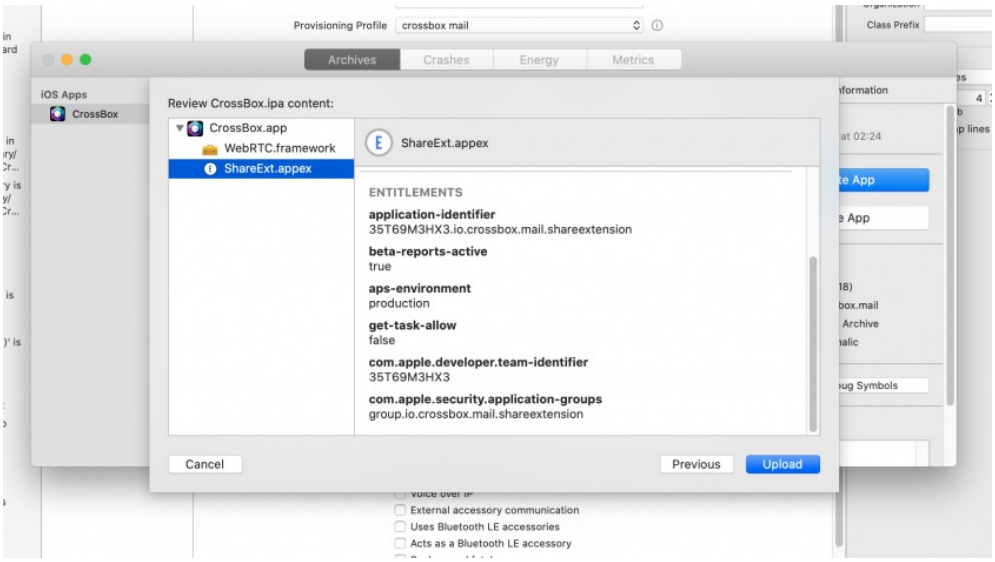




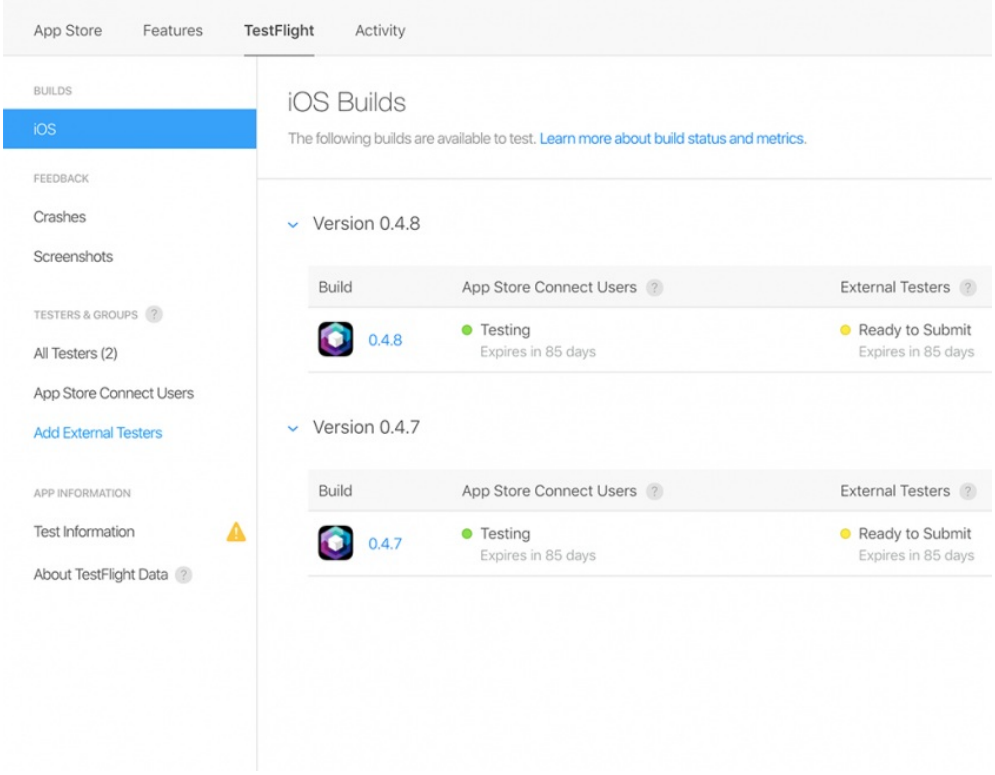
After this step, you will see the final step







- After the upload, the processing may take up to an hour.
- You will get an e-mail after the build has been processed by Apple.
- After the submitted app has been processed you can start filling up the information and prepare for submission. Apple will require you to provide a test account which they'll use to review the app



> NOTE: There's a known issue with TestFlight that it might say that processing is not complete, in order to fix the problem logout and login again. Sometimes it may take between 30 minutes to a few hours to complete.

- Apple may ask you to answer some questions about the App.

We prepared the answers for you. Replace placeholders
APP_NAME, COUNTRY_REGION, SERVICE_PROVIDER_SIGNUP_LINK,
and SERVICE_PROVIDER_WEBSITE with your own.

1. What is the purpose of the app?
APP_NAME is the All-in-one Communication Suite designed for Small and Medium Enterprises in COUNTRY_REGION(optional) for their modern-day communication needs.

2. Who is your target audience?
Small and Medium Enterprises and individuals

3. Is this app built for a specific company and its internal use? If yes, which company?
No

4. Or is the app built for internal use for your company?
No

5. How do users obtain an account?
By signing up to SERVICE_PROVIDER_SIGNUP_LINK and purchasing the hosting service for their domain. After signup, they can create an account via admin interface.

6. If users create an account to use your app, are there fees involved?
User pays for the service monthly or yearly via the SERVICE_PROVIDER_WEBSITE

7. Does your app access any paid content or services?

The user only pays for the service monthly or yearly via the SERVICE_PROVIDER_WEBSITE

8. What are the paid content or services, and what are the costs?
Costs are N to N $ as listed on the website SERVICE_PROVIDER_WEBSITE

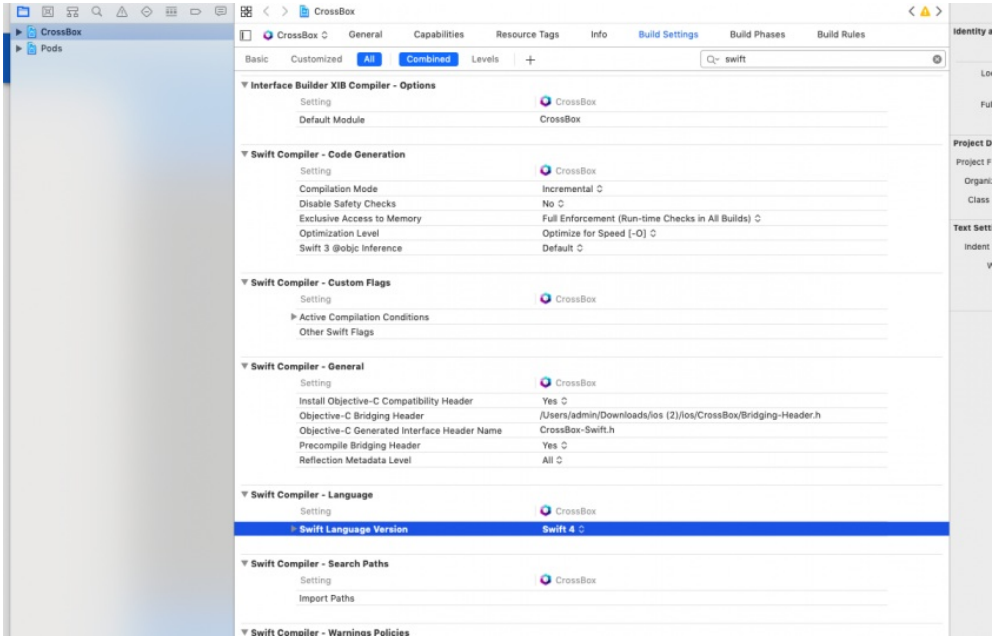
9. Do individual customers pay for the content or services?
Yes, they can purchase the service via SERVICE_PROVIDER_WEBSITE as there is no requirement to be a business.

10. If no, does a company or organization pay for the content or services?
N/A

Known issues on IOS

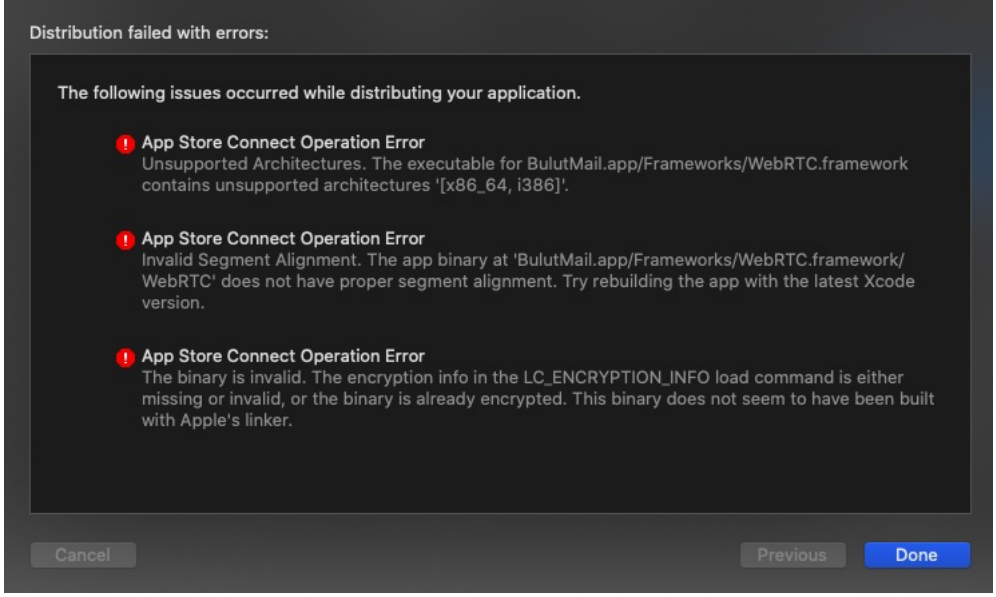
If the build process in Xcode fails, you can try to fix it by:

1. Installing cocoa pods on your machine - https://cocoapods.org/
2. Correcting the build settings for the Swift version - make sure you select Swift Version 4



If you get any of the following errors:

- Unsupported Architectures. The executable contains unsupported architectures '[x86_64, i386]
- Invalid Segment Alignment. The app binary does not have proper segment alignment
- The binary is invalid. The encryption info in the LC_ENCRYPTION_INFO load command is either missing or invalid, or the binary is already encrypted



These errors happen if you forgot to run sh prepare-production.sh before starting the submission to the Apple Store as explained here.

App Connector

App Connector

App Connector Documentation

Intro

App Connector is a CrossBox module that integrates multiple webmail and communication apps into a single interface. Users can switch between different apps and email accounts with ease, while the app management interface allows domain administrators full control over the apps running under their hosting account.

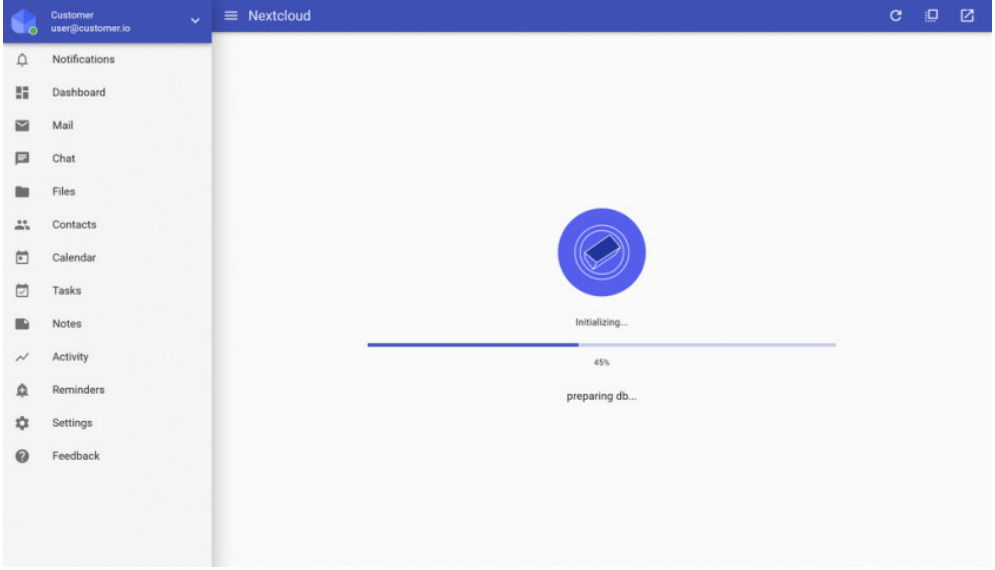
Apps

Currently, App Connector provides integration of the following applications:

- Nextcloud
- Mattermost
- Roundcube
- Rainloop
- Horde
- SquirrelMail

Installation

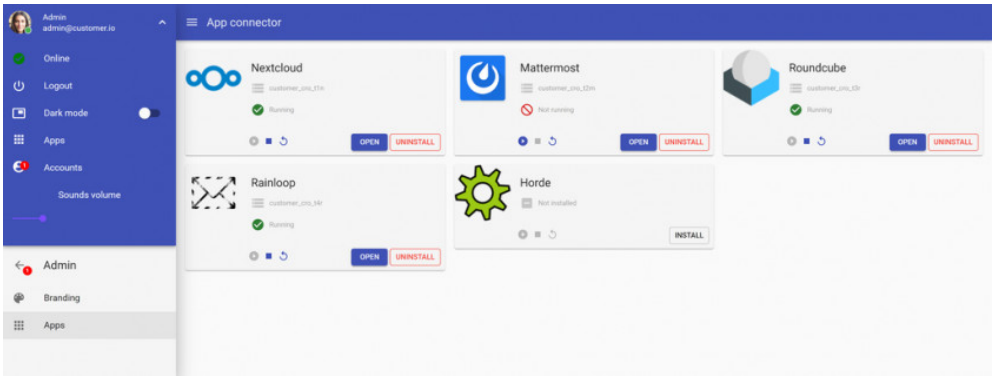
To make apps accessible to the user as fast as possible, CrossBox takes care of the entire setup process. The first time a user tries to access the given app, they are presented with a progress indicator, as shown in the image:



The process of setting up the app takes just a few seconds and completes with the given app loading in the interface. At this point, the app is ready for use, with no additional actions needed from the user.

Start/Stop/Restart/Uninstall

Domain admin (hosting account owner) can use the "Apps" interface to start, stop, restart, or uninstall apps with a simple click.



Quotas

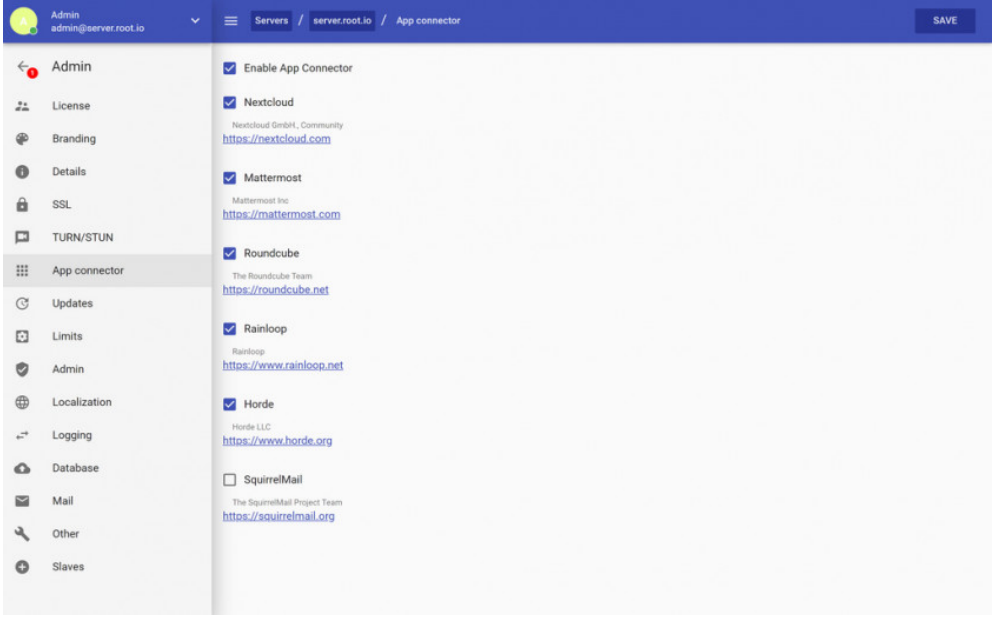
All third-party apps installed by the App Connector adhere to the email and hosting account's disk and database quotas.

Enabling and disabling apps

Enabling and disabling apps can be done on multiple levels:

- server admin (the top level) can decide what apps are available for use generally on the server
- reseller admin (the mid-level) can decide what apps are available for use to the accounts that are under its ownership
- domain admin (the bottom level) can decide what apps are available for the domains they own

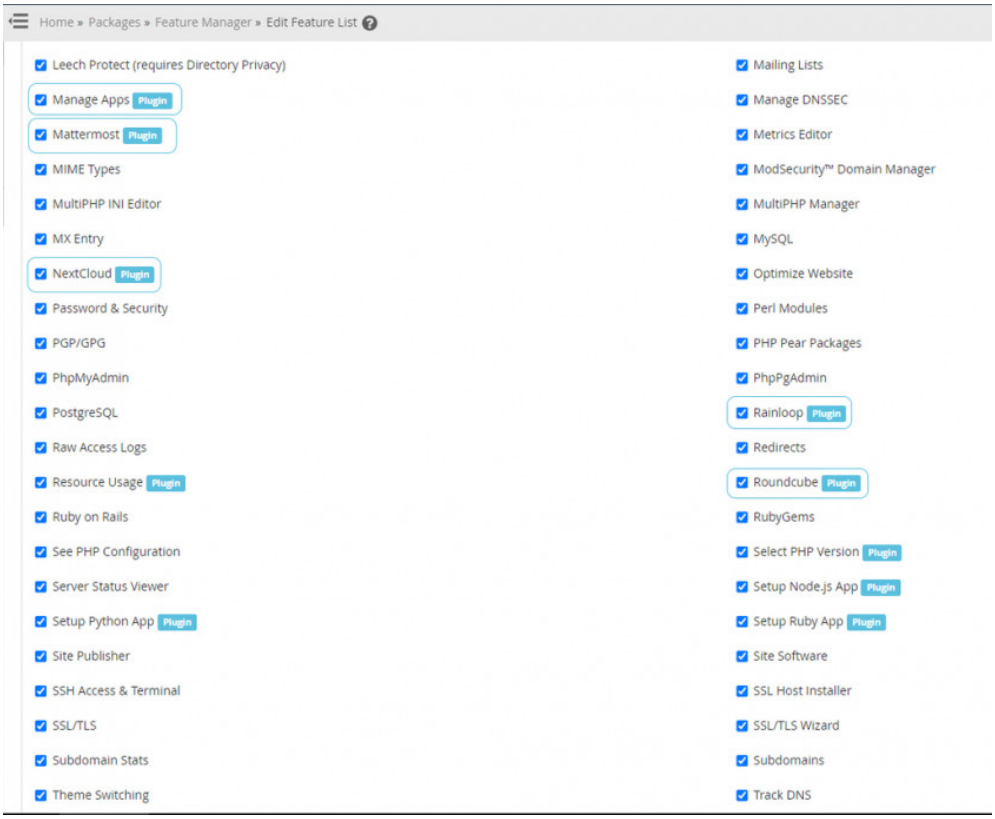
Apps availability is inherited from top to bottom, meaning that if the server admin disables, for example, Mattermost, the levels below won't be able to install it.



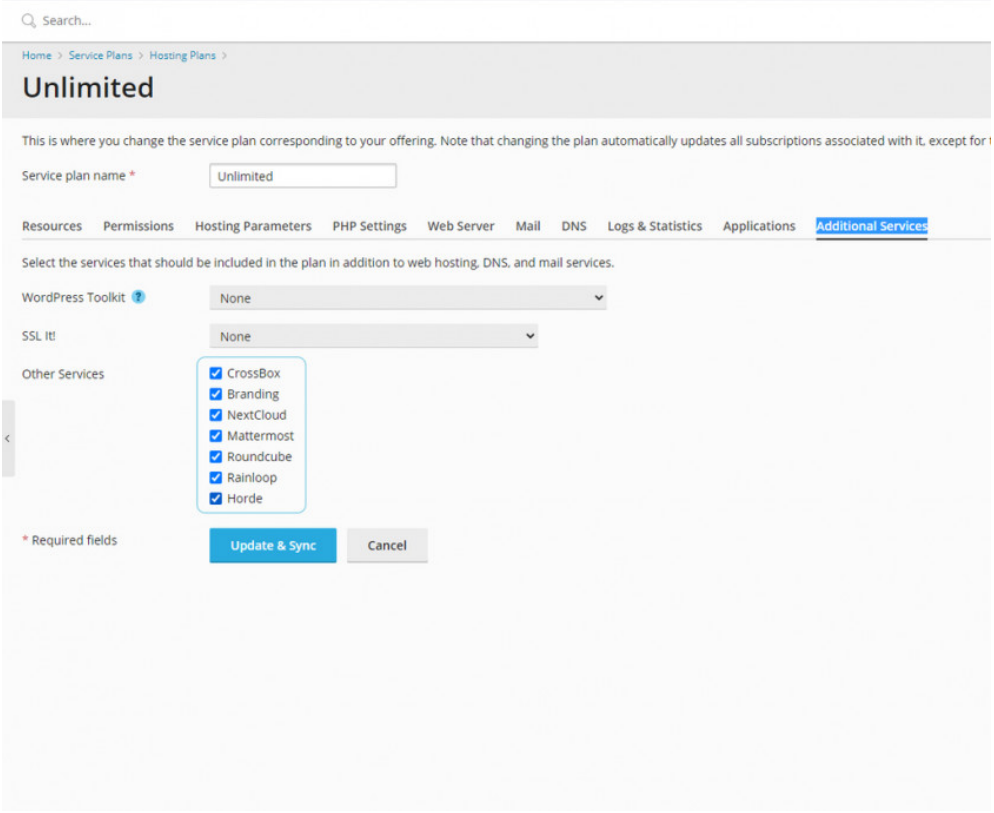
Enabling and disabling apps per hosting package/subscription

This feature is currently supported on cPanel and Plesk and allows server administrator to enable or disable specific apps at the hosting package (service plan) level.

For cPanel, this is done using WHM's feature manager:



For Plesk, this is done using the Additional Services manager:



Control Panel Plugins

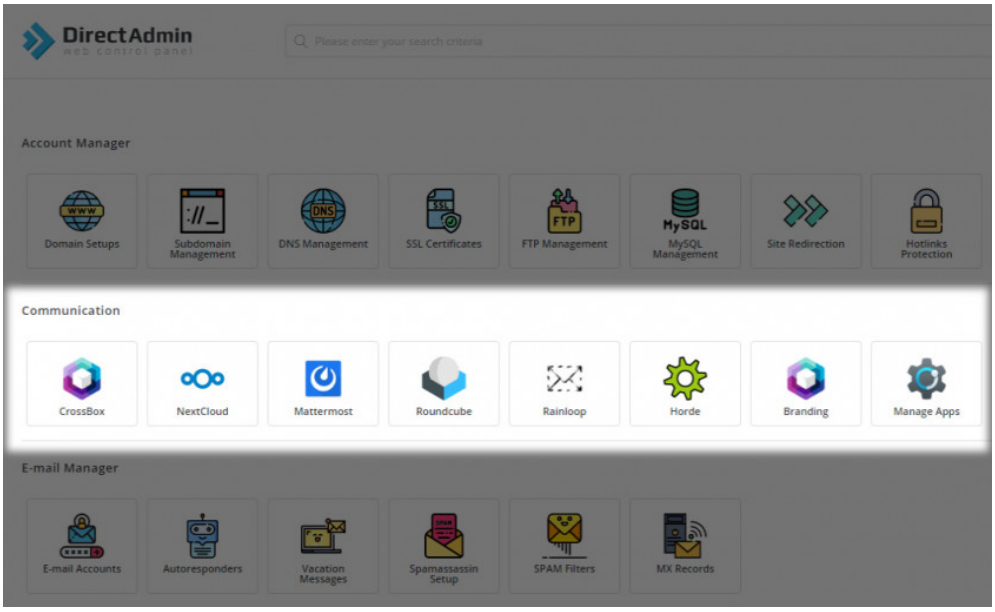
The app connector comes with native plugins for the most popular control panels:

- DirectAdmin
- cPanel
- Plesk

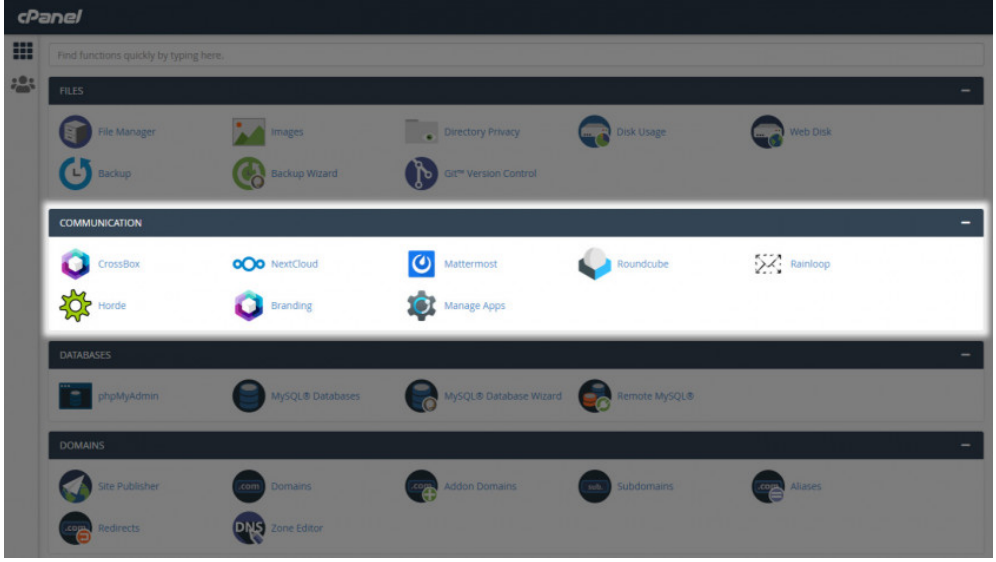
These plugins enable panel users to access all apps quickly from the dedicated communication section:

Quick Access

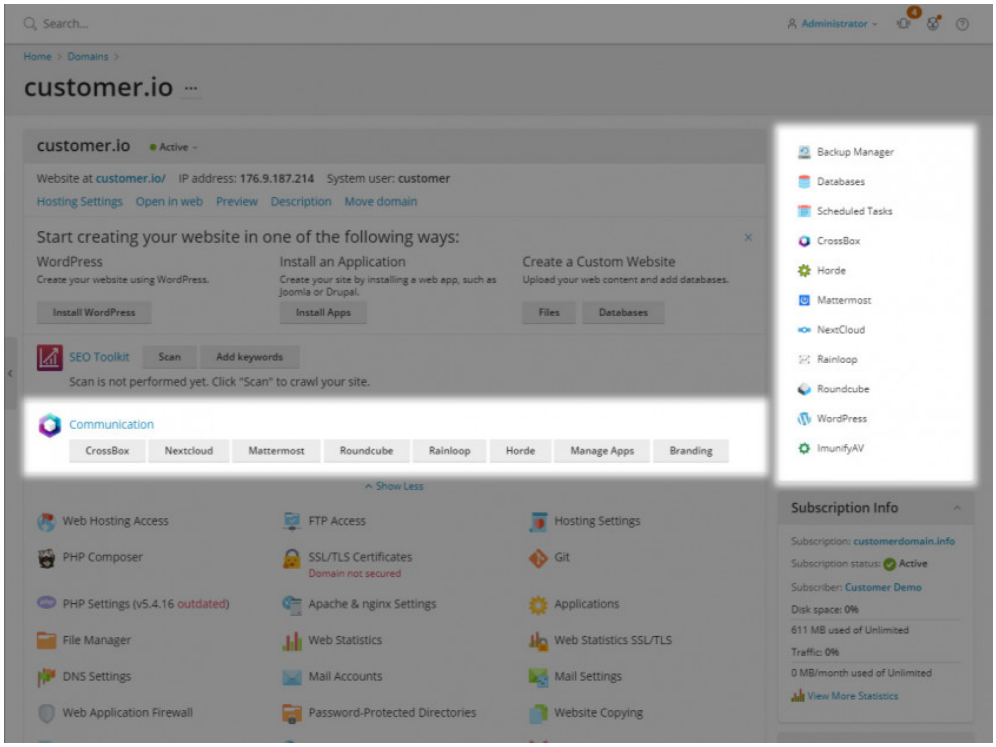
DirectAdmin



cPanel



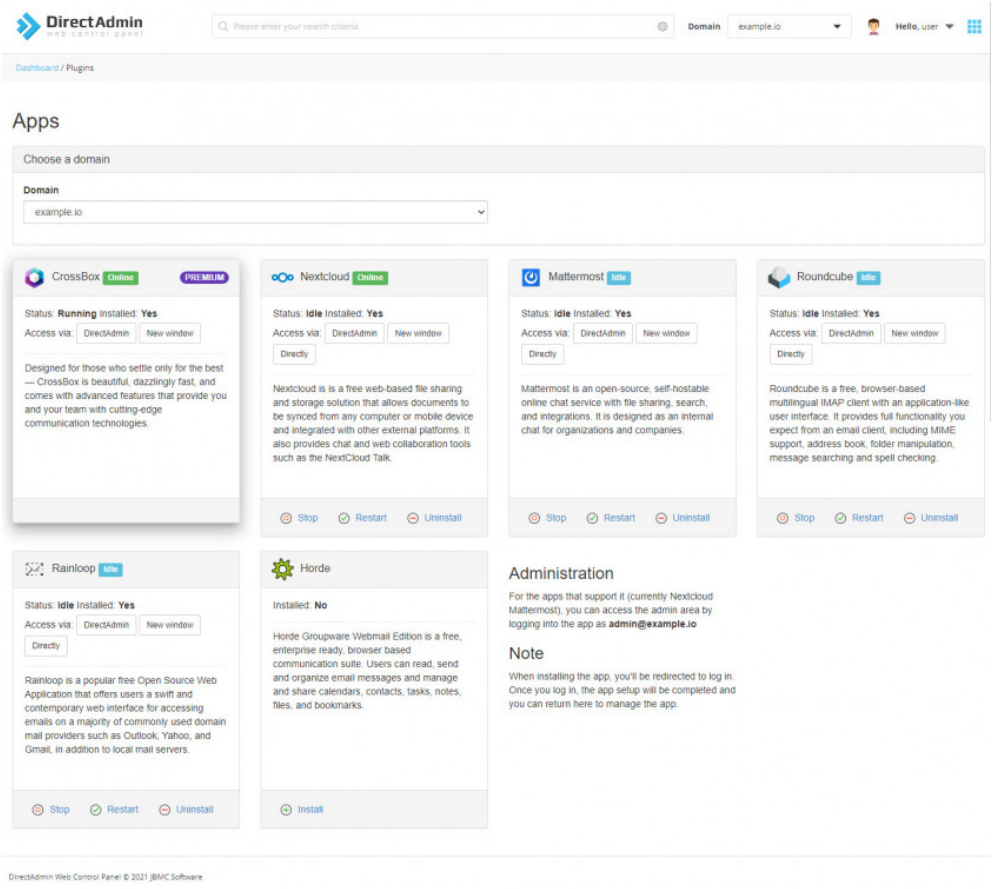
Plesk



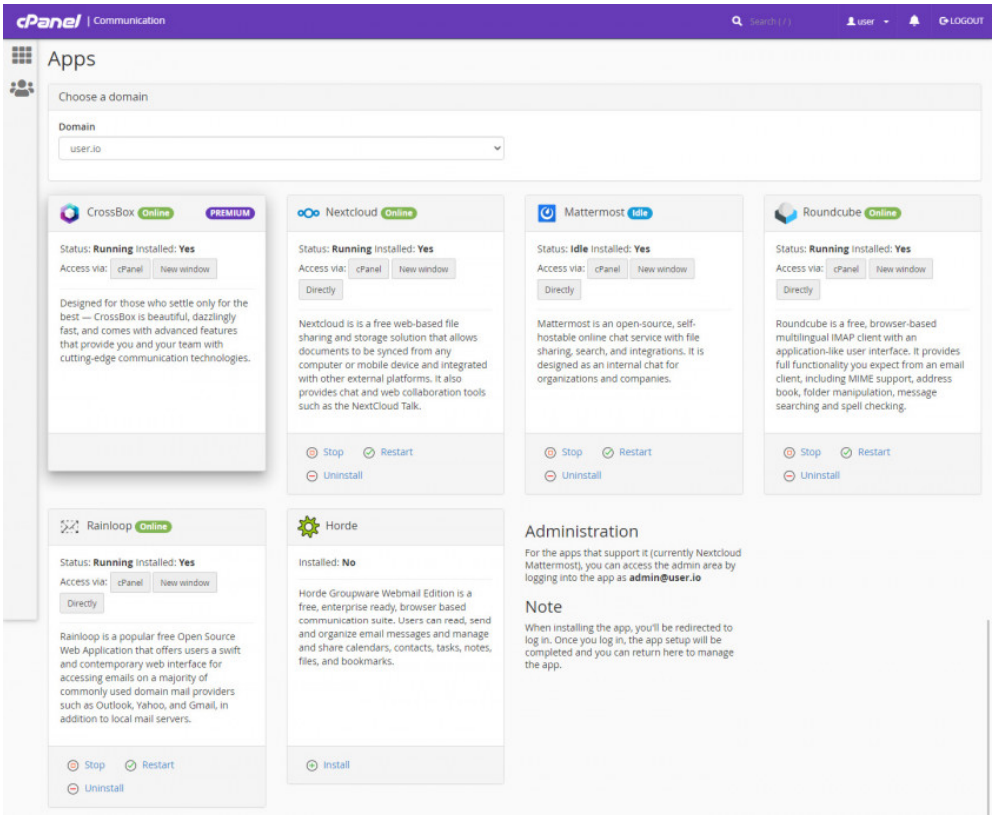
Application Management

By clicking the "Manage Apps" item, users are presented with an intuitive and easy to use interface for starting, stopping, restarting, and uninstalling apps:

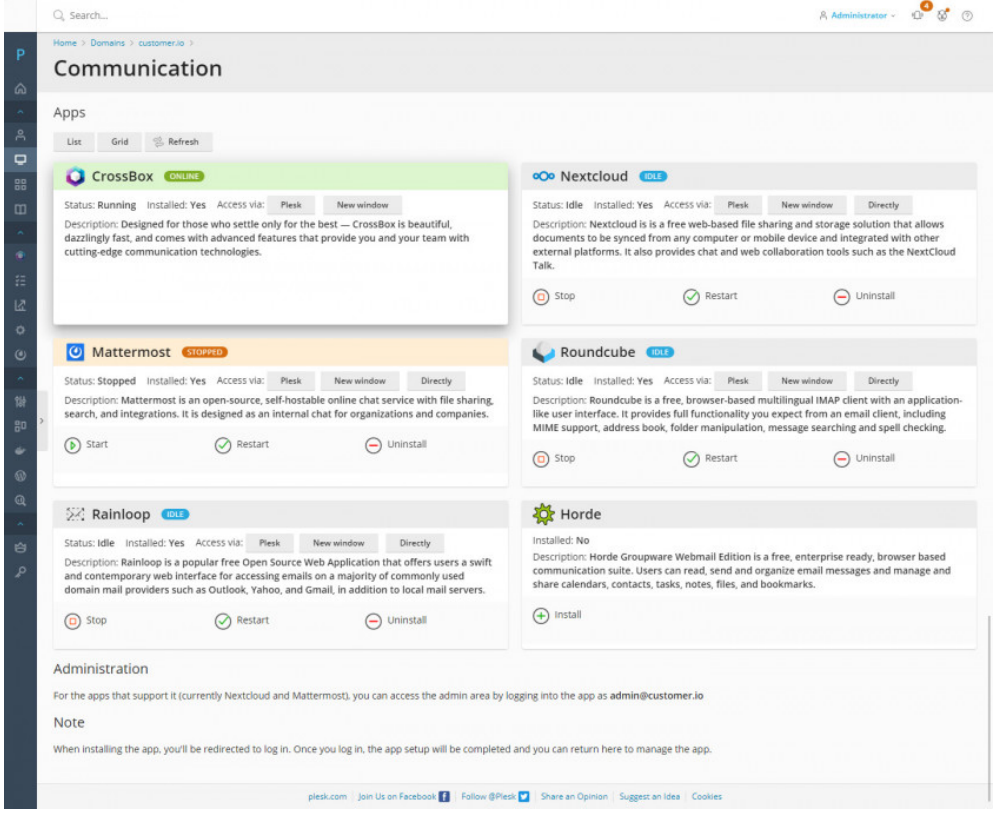
DirectAdmin



cPanel

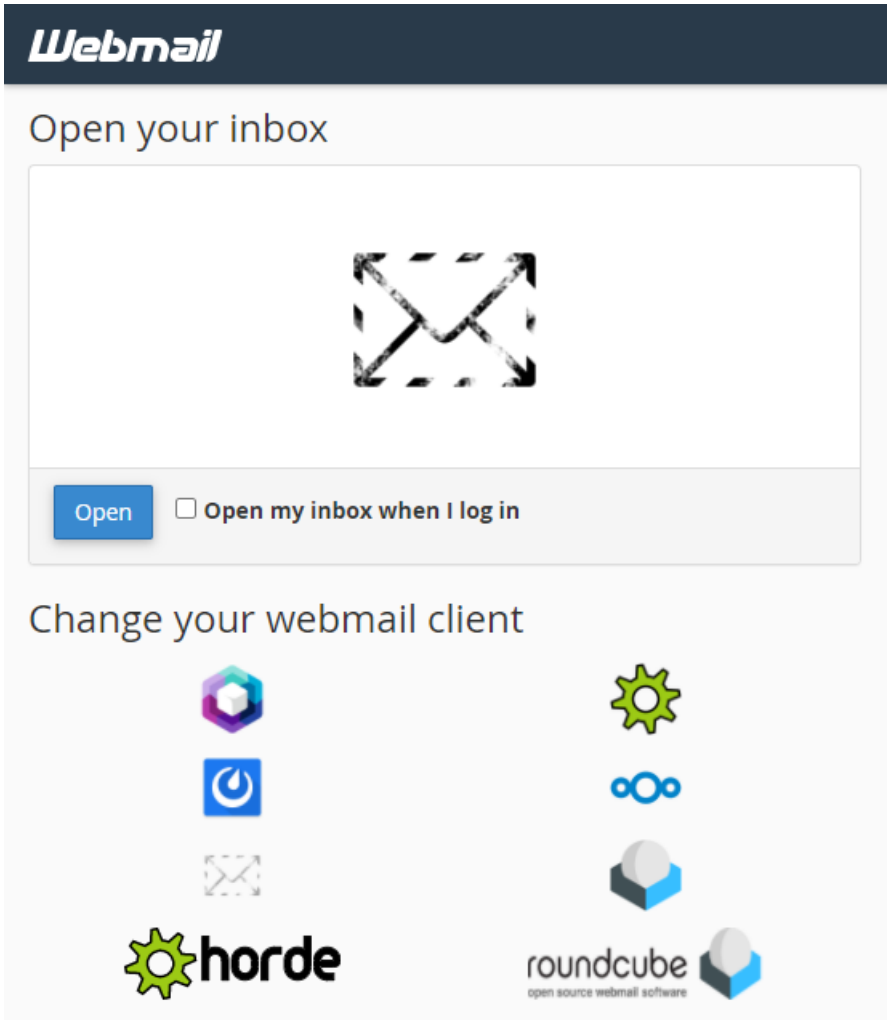


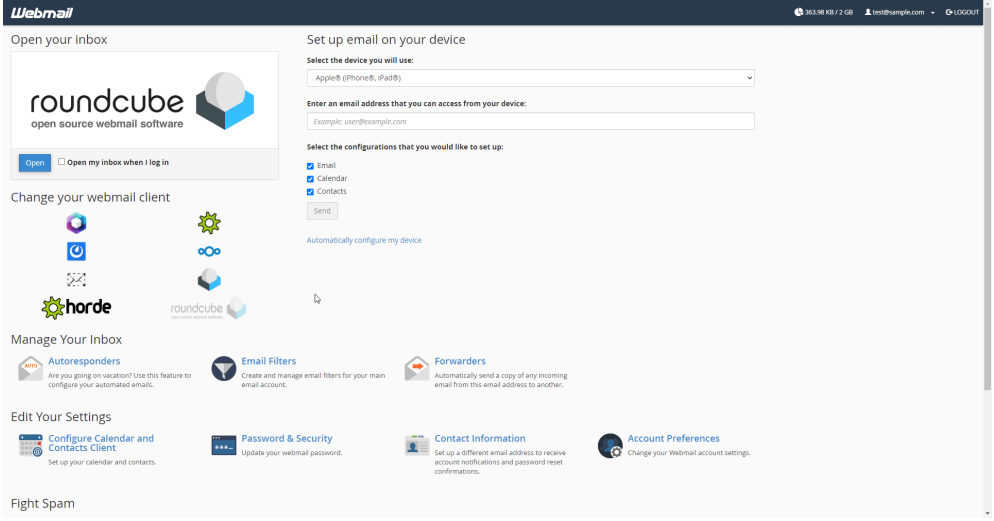
Plesk



cPanel Webmail

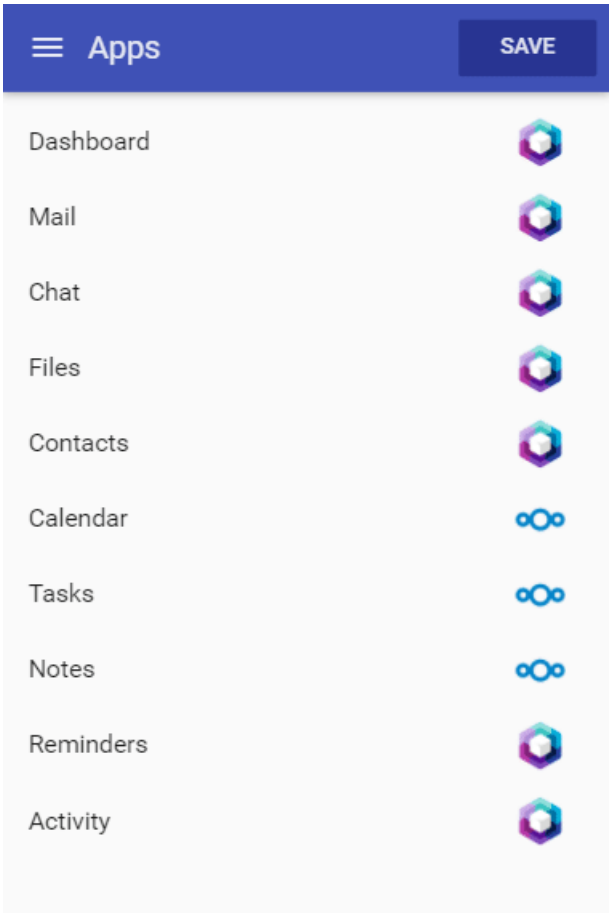
Apps are available for use through the cPanel's Webmail System. The auto login feature is also fully supported:





Module Mapping

Using the module mapping interface located in Settings -> Apps, each user can freely map modules to dedicated apps. For example, a user can choose CrossBox for email but Nextcloud or Horde for the calendar. Users can also choose to disable certain modules entirely.



Known Issues

"Manage Apps" section within DirectAdmin may not work if the user has 2FA enabled for the account.