

## FreeBSD Administration - Support #329

### Adding a Jail to FreeNAS

02/08/2014 01:42 PM - Daniel Curtis

<b>Status:</b>	Closed	<b>Start date:</b>	02/08/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Daniel Curtis	<b>% Done:</b>	100%
<b>Category:</b>	Jails / Containers	<b>Estimated time:</b>	1.50 hour
<b>Target version:</b>	FreeBSD 9	<b>Spent time:</b>	3.00 hours

#### Description

This section describes how to use the Jails method, which allows users who are comfortable using the command line to have more control over software installation and management.

While the Plugins method automatically created a FreeBSD jail for each installed PBI, the Jails method allows the user to create as many jails as needed and to specify the type of jail. Unlike the Plugins method, one is not limited to installing only one application per jail.

Essentially, a FreeBSD jail provides a very light-weight, operating system-level virtualization. Consider it as an independent FreeBSD operating system running on the same hardware, without all of the overhead usually associated with virtualization. This means that any software and configurations within a jail are isolated from both the FreeNAS® operating system and any other jails running on that system. During creation, some jail types provide a VIMAGE option which provides that jail with its own, independent networking stack. This allows the jail to do its own IP broadcasting, which is required by some applications.

The following types of jails can be created:

- 1. Plugin jail:** this type of jail provides the most flexibility for software installation. Similar to the Plugins method, this type of jail supports the installation of FreeNAS® PBIs, which integrate into the FreeNAS® GUI. In addition to FreeNAS® PBIs, you can also install the following types of software within a plugin jail: FreeBSD ports and FreeBSD pkgng packages. However, only FreeNAS® PBIs can be managed from the GUI as the other types of software are managed from the command line of the jail. Further, the other types of jails do not support the ability to install FreeNAS® PBIs. If you plan to install FreeNAS® PBIs, create a plugin jail.
- 2. Port jail:** 2. this type of jail supports the installation of FreeBSD ports and FreeBSD pkgng packages. It does not support the installation of FreeNAS® PBIs, meaning that any software installed in this type of jail must be managed from the command line of the jail.
- 3. Standard jail:** this type of jail is functionally the same as a port jail. A distinction is made for those users who prefer to separate network services, such as DHCP or DNS servers, from other installed software.
- 4. Linux jail:** due to the [FreeBSD linux binary compatibility layer](#), Linux can be installed into a jail and software can be installed using the package management system provided by the installed Linux distro. At this time, the Linux distro must be a 32-bit version and any applications installed into the jail must be available as a 32-bit binary.

NOTE: the software which can be installed into a Linux jail is limited to the command line package management tool provided by that Linux distribution. If you install software into a Linux jail, install the 32-bit version of the software.

The ability to create multiple jails and multiple types of jails offers great flexibility and application separation to the administrator. For example, one could create a separate plugin jail for each FreeNAS® plugin, a separate port jail for each application that is not available as a FreeNAS® plugin, and a separate standard jail for each installed network server. Alternately, one has the ability to create one jail and to mix and match how the software is installed into that jail.

## Jails Configuration

Before you can create any jails, you must first configure which volume or dataset will be used to hold the jails. To do this go to [Jails -> Configuration](#) to access the the jails configuration screen.

NOTE: if you have already used the Plugins method, all of the fields in this screen will automatically be filled in. You should still double-check that the pre-configured values are appropriate for your jails.

While a jail can be installed on a UFS volume, it is recommended to use ZFS and to create a dataset to use for the Jail Root. As jails are created on a ZFS system, they will automatically be installed into their own dataset under the specified path. For example, if you configure a Jail Root of /mnt/volume1/dataset1 and create a jail named jail1, it will be installed into its own dataset named /mnt/volume1/dataset1/jail1.

- **Jail Root:** mandatory as you cannot add a jail until this is set

- **IPv4 Network:** see explanation below table; format is IP address of network / CIDR mask
- **IPv4 Network Start Address:** see explanation below table; format is IP address of host / CIDR mask
- **IPv4 Network End Address:** see explanation below table; format is IP address of host / CIDR mask

When selecting the "Jail Root", ensure that the size of the selected volume or dataset is sufficient to hold the number of jails to be installed as well as any software, log files, and data to be stored within each jail. At a bare minimum, budget at least 2GB per jail and do not select a dataset that is less than 2GB in size.

NOTE: if you plan to add storage to a jail, be aware that path size is limited to 88 characters. Make sure that the length of your volume name plus the dataset name plus the jail name does not exceed this limit.

FreeNAS® will automatically detect and display the "IPv4 Network" that the administrative interface is connected to. This setting is important as the IPv4 network must be pingable from the FreeNAS® system in order for your jails and any installed software to be accessible. If your network topology requires you to change the default value, you will also need to configure a default gateway, and possibly a static route, to the specified network. If you change this value, ensure that the subnet mask value is correct as an incorrect mask can make the IP network unreachable. When in doubt, keep the default setting for "IPv4 Network". If you are using VMware, make sure that the vswitch is set to promiscuous mode.

Review the default values of the "IPv4 Network Start Address" and "IPv4 Network End Address" to determine if that range is appropriate for the number of jails that you will create. If there is a DHCP server on the network, make sure that this range of addresses is excluded from the scope of the DHCP server. As jails are created, they will automatically be assigned the next free IP address within the range specified by these two values.

NOTE: these 4 fields are necessary for the proper operation of Jails. If you are unable to add, start, or access the software installed into jails, double-check the values in these fields. In particular, make sure that the specified IPv4 settings are reachable by clients and that the specified addresses are not in use by any other clients in the network.

## Adding Jails

To create a jail, click [Jails -> Add Jails](#) to access the jails screen.

NOTE: the "Add Jails" menu item will not appear until after you configure [Jails -> Configuration](#).

- **Jail Name:** mandatory; can only contain letters and numbers
- **type:** default choices are pluginjail, portjail, standard, debian, gentoo, ubuntu, suse, and centos; on a 64-bit system, options are also available for creating the 32-bit versions of a plugin, port, or standard jail
- **IPv4 address:** will be automatically assigned the next free address from the range specified in Jails Configuration; if you change the default address, make sure it is reachable within the FreeNAS® system's network and is not in use by any other host on the network
- **IPv4 netmask:** select the subnet mask associated with IPv4 address
- **IPv4 bridge address:** see NOTE below; will be greyed out for Linux jails or if VIMAGE is unchecked
- **IPv4 bridge netmask:** select the subnet mask associated with IPv4 bridge address; will be greyed out for Linux jails or if VIMAGE is unchecked
- **IPv4 default gateway:** used to set the jail's default gateway IPv4 address; will be greyed out for Linux jails or if VIMAGE is unchecked
- **IPv6 address:** if IPv6 has been configured, will be automatically assigned the next free address from the range specified in Jails Configuration
- **IPv6 prefix length:** select the prefix length associated with IPv6 address
- **IPv6 bridge address:** see NOTE below; will be greyed out for Linux jails or if VIMAGE is unchecked
- **IPv6 bridge prefix length:** select the prefix length associated with IPv6 address; will be greyed out for Linux jails or if VIMAGE is unchecked
- **IPv6 default gateway:** used to set the jail's default gateway IPv6 address; will be greyed out for Linux jails or if VIMAGE is unchecked
- **MAC:** if a static MAC address is needed, input it here; requires VIMAGE to be checked
- **Sysctls:** comma-delimited list of sysctls to set inside jail (e.g. allow.sysvipc=1,allow.raw\_sockets=1)
- **Autostart:** uncheck if you want to start the jail manually
- **VIMAGE:** gives a jail its own virtualized network stack; requires promiscuous mode to be enabled on the interface; does not apply to Linux jails
- **NAT:** enables Network Address Translation for the jail; will be greyed out for Linux jails or if VIMAGE is unchecked
- **vanilla:** uncheck this box if you plan to install FreeBSD packages into a portjail or standard jail

NOTE: The IPv4 and IPv6 bridge interface is used to bridge the epair(4) device, which is automatically created for each started jail, to a physical network device. The default network device is the one that is configured with a default gateway. So, if em0 is the FreeBSD name of the physical interface and three jails are running, the following virtual interfaces will be automatically created: bridge0, epair0a, epair1a, and epair2a. The physical interface em0 will be added to the bridge, as well as each epair device. The other half of the epair will be placed inside the jail and will be assigned the IP address specified for that jail. The bridge interface will be assigned

an alias of the default gateway for that jail, if configured, or the bridge IP, if configured; either is correct.

A "traditional" FreeBSD jail does not use VIMAGE or NAT. If you uncheck both of these boxes, you need to configure the jail with an IP address within the same network as the interface it is bound to, and that address will be assigned as an alias on that interface. To use a VIMAGE jail on the same subnet, disable NAT, and configure an IP address within the same network. In both of these cases, you only configure an IP address and do not configure a bridge or a gateway address.

After making your selections, click the OK button. The jail will be created and will be added to the tree under Jails. By default, a plugin jail will be created and automatically started, unless you specify otherwise.

The first time you add a type of jail, the GUI will automatically download the necessary components from the Internet. If it is unable to connect to the Internet, the jail creation will fail. Otherwise, a progress bar will indicate the status of the download and provide an estimated time for the process to complete. Once the first jail is created, subsequent jails of that type will be added instantaneously as the downloaded base for creating that type of jail is saved to the Jail Root.

## Managing Jails

To view and configure the added jails, click [Jails -> View](#) all Jails.

Click a jail's entry to access its configuration icons. In order, from left to right, these icons are used to:

- **Edit Jail**: edit the jail's settings as described in the next section.
- **Add Storage**: configure the jail to access an area of storage as described in Adding Storage.
- **Upload Plugin**: only available in a plugin jail. Used to install plugins as described in Installing FreeNAS® PBIs.
- **Start/Stop**: this icon will vary, depending upon the current running status of the jail. If the jail is currently stopped, the icon will be green and can be used to start the jail. If the jail is currently running, the icon will be red and can be used to stop the jail. A stopped jail and its applications are inaccessible until it is restarted.
- **Shell**: used to access a root command prompt in order to configure the selected jail from the command line.
- **Delete**: deleting the specified jail also deletes any software that was installed in that jail. The GUI will display a warning which requires you to click the Yes button, indicating that you are sure that you want to delete the jail, before this operation is performed.

## Accessing a Jail Using SSH Instead of its Shell Icon

If you prefer to use ssh to access a jail you will need to first start the ssh service and create a user account for ssh access. Since this configuration occurs on a jail-by-jail basis, click the **Shell** icon for the jail you wish to configure ssh access to.

To start the SSH service on a non-Linux jail, look for the following line in that jail's /etc/rc.conf:

```
vi /etc/rc.conf
```

Then select the following string, change the NO to **YES** and save the file:

```
sshd_enable="NO"
```

```
:s/NO/YES/
```

```
:wq
```

Then, start the SSH daemon:

```
service sshd start
```

The host RSA key pair should be generated and the key's fingerprint and random art image displayed.

For a Linux jail, refer to the documentation for that Linux distribution for instructions on how to start the SSH service. Depending upon the distribution, you may have to first install a SSH server.

## Add a user account

If you want the user to have superuser privileges to a non-Linux jail, make sure the user is placed in the **wheel** group when it is created. Type adduser and follow the prompts:

```
mkdir /home
adduser
```

When you get to this prompt, do not press enter but instead type **wheel**:

```
Login group is user1. Invite user1 into other groups? []: wheel
```

## Set the root password

Once the user is created, set the root password so that the new user will be able to use the su command to gain superuser privilege. To set the password, type passwd then input and confirm the desired password.

```
passwd
```

## Create root SSH key

Most access to the jail will be done remotely over SSH, it is good practice to create a strong SSH key pair. Like so:

```
ssh-keygen -t ecdsa
```

NOTE: If password-less logins are going to be used, make sure to NOT put a password in when prompted.

For a Linux jail, you will need to create a user account using the software that comes with the Linux distribution. Since Linux does not use the wheel group, if you wish to give this user superuser privileges, instead install and configure the sudo application.

Finally, test from another system that the user can successfully ssh in and become the superuser. In this example, a user named user1 uses ssh to access the non-Linux jail at 192.168.2.3. The first time the user logs in, they will be asked to verify the fingerprint of the host:

```
ssh user1@192.168.2.3
```

```
The authenticity of host '192.168.2.3 (192.168.2.3)' can't be established.
ECDSA key fingerprint is 6f:93:e5:36:4f:54:ed:4b:9c:c8:c2:71:89:c1:58:f0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.2.3' (ECDSA) to the list of known hosts.
Password: SuperSecretPassword
```

NOTE: each jail has its own user accounts and service configuration. This means that you will need to repeat these steps for each jail that requires SSH access.

Most of these settings were previously described and can be changed using this screen after jail creation. The following settings differ between the "Add Jail" and "Edit Jail" screens:

- **Jail Name:** this setting is read-only once the jail has been created.
- **IPv4 aliases:** once a jail has been created, this field can be used to add additional IPv4 addresses, which are known as aliases. When adding multiple aliases, use a comma delimited list.
- **IPv6 aliases:** once a jail has been created, this field can be used to add additional IPv6 addresses. When adding multiple aliases, use a comma delimited list.

NOTE: if you need to modify the IP address information for a jail, use it's **Edit Jail** button instead of the associated networking commands from the command line of the jail.

Adding Storage

It is possible to give a jail access to an area of storage on the FreeNAS® system. This is useful if you install an application that stores a large amount of data or if an installed application needs access to the data stored on the FreeNAS® system. An example would be transmission, which stores torrents. The storage is added using the mount\_nullfs(8) mechanism which links data that resides outside of the jail as a storage area within the jail.

To add storage, click the **Add Storage** button for a highlighted jail's entry. This screen can also be accessed by expanding the jail's name in the tree view and clicking Storage -> Add Storage.

Browse to the "Source" and "Destination", where:

- \* Source: is the directory or dataset on the FreeNAS® system you would like to gain access to from the jail. This directory must reside outside of the volume or dataset being used by the jail. This is why it is recommended to create a separate dataset to store jails, as the dataset holding the jails will always be separate from any datasets used for storage on the FreeNAS® system.
- \* Destination: select the directory within the jail which will be linked to the "Source" storage area.

When you are adding storage, it is typically because the user and group account associated with an application installed inside of a jail needs to access data stored on the FreeNAS® system. Before selecting the "Source", it is important to first ensure that the permissions of the selected directory or dataset grant permission to the user/group account inside of the jail. This is typically not the default, as the users and groups created inside of a jail are totally separate from the users and groups of the FreeNAS® system.

This means that the workflow for adding storage is usually as follows:

1. Determine the name of the user and group account used by the application. For example, the installation of the transmission application automatically creates a user account named transmission and a group account named transmission. When in doubt, check the files /etc/passwd (to find the user account) and /etc/group (to find the group account) inside of the jail. Typically, the user and group names are similar to the application name. Also, the UID and GID are usually the same as the port number used by the service.
2. On the FreeNAS® system, create a user account and group account to match the name of the user and group used by the application in the jail.
3. On the FreeNAS® system, determine if you want the jail to have access to existing data or if you want to set aside an area of storage for the jail to use.
4. If the jail should access existing data, edit the permissions of the volume or dataset so that the user and group account has the desired read and write access. If multiple applications or jails are to have access to the same data, you will need to create a separate group and add each needed user account to that group.
5. If you are instead setting aside an area of storage for that jail (or individual application), create a dataset. Then, edit the permissions of that dataset so that the user and group account has the desired read and write access.
6. Use the **Add Storage** button of the jail and select the configured volume/dataset as the "Source".

If you wish to prevent writes to the storage, check the box "Read-Only".

By default, the "Create directory" box is checked. This means that the directory will automatically be created for you under the specified "Destination" path if the directory does not already exist.

Once a storage has been added, it will be added to the tree under the specified jail. In the example shown in Figure 10.2e, a dataset named volume1/data has been chosen as the "Source" as it contains the files stored on the FreeNAS® system. When the storage was created, the user browsed to volume1/jails/pluginjail/usr/local in the "Destination" field, then typed in test as the directory. Since this directory did not already exist, it was created as the "Create directory" box was left as checked. The resulting storage was added to the pluginjail entry in the tree as /usr/local/test. The user has clicked this /usr/local/test entry in order to access its edit screen.

By default, the storage is mounted as it is created. To unmount the storage, uncheck its "Mounted?" box.

NOTE: a mounted dataset will not automatically mount any of its child datasets. While the child datasets may appear browsable inside the jail, any changes will not be visible. Since each dataset is considered to be its own filesystem, each child dataset must have its own mount point, meaning that you need to create a separate storage for any child datasets which need to be mounted.

To delete the storage, click its **Delete** button.

**DANGER!** It is important to realize that an added storage is really just a pointer to the selected storage directory on the FreeNAS® system. It does not create a copy of that data within the jail. This means that if you delete any files from the "Destination" directory located in the jail, you are really deleting those files from the "Source" directory located on the FreeNAS® system. However, if you delete the storage, you are only deleting the pointer, not the data itself.

## Resources

<http://doc.freenas.org/index.php/Jails>

[http://doc.freenas.org/index.php/Jails\\_Configuration](http://doc.freenas.org/index.php/Jails_Configuration)

[http://doc.freenas.org/index.php/Adding\\_Jails](http://doc.freenas.org/index.php/Adding_Jails)

## History

#1 - 02/10/2014 04:17 PM - Daniel Curtis

- Status changed from Resolved to Closed

#2 - 06/11/2014 08:46 AM - Daniel Curtis

- Description updated

**#3 - 02/15/2015 08:50 PM - Daniel Curtis**

- Project changed from 81 to FreeBSD Administration

- Category set to Jails / Containers

- Target version set to FreeBSD 9