

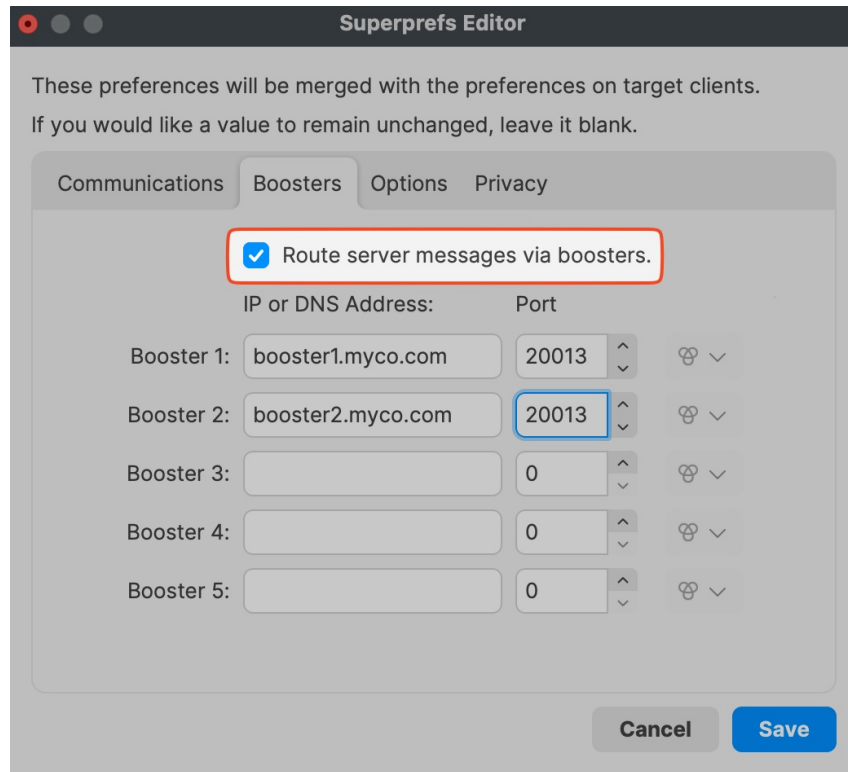
Troubleshooting

- [What are "Server Messages" and why do I want them?](#)
- [Networking - Assign static IP Address for a FileWave Booster Appliance](#)
- [Booster ID Generation](#)
- [Enable core dumps for booster crashes](#)

What are "Server Messages" and why do I want them?

What

Perhaps you have seen the option in [Superprefs Editor](#) or in [Client Monitor](#) about "Route server messages via boosters" and asked yourself what that was?



When/Why

We know that when a client is talking to the FileWave server it does a check-in every so often (2min default). But how does it talk and what is said?

When we take a look at the [Default TCP and UDP Port Usage](#) for FileWave we see 20015 and 20017 (communications between client and server). There is also NATS ports that handle notifications for things like initiating a TeamViewer session.

Note: The default port setting is 20015. However, SSL is now required, and the system will automatically use port 20017 instead when 20015 is entered. Do not manually set the port to 20017. Always enter 20015, and the system will handle the SSL port change for you.

How

Basically, server messages are the communications a client needs to work with the server. Below is a list of messages that could be sent via the publishing and routing services if this option is enabled. You should enable Route Server Messages via boosters to let your Boosters handle these additional communications to take burden off the FileWave Server.

- Checkin
- Fileset properties and status
- Software updates
- Lock / Unlock client
- Kiosk categories and item info

Networking - Assign static IP Address for a FileWave Booster Appliance

For the Linux based Booster if you cannot use the port <https://server:10000> to change network setting please follow the instructions below:

▼ Debian Linux

Debian Linux

Changing the IP address in Debian 12, which uses `systemd-networkd` for network management, involves different steps compared to CentOS. The following guide is tailored for Debian 12 servers using `systemd-networkd` but you could also use [Webmin](#) on your server assuming the server comes online initially with DHCP.

⚠ For Webmin know that you will need to go to Webmin -> Webmin Configuration -> Operating System and Environment and make sure it's set to Debian 12.4 (Or whatever version we are at when you set up your system. You can see this with `cat /etc/debian_version` on the server.

1. Locate Network Interface:

First, identify the network interface you wish to configure. You can list all network interfaces using:

```
networkctl list
```

```
admin@ip-172-30-3-220:/etc/network$ networkctl list
```

IDX	LINK	TYPE	OPERATIONAL	SETUP
1	lo	loopback	carrier	unmanaged
2	ens5	ether	routable	configured

2 links listed.

2. Configure Network Settings:

`systemd-networkd` uses individual `.network` files for each network interface, located in `/etc/systemd/network/`. Create or edit the network configuration file for your interface, named like `10-eth0.network` (replace `eth0` with your interface name).

```
sudo nano /etc/systemd/network/10-eth0.network
```

3. Configure IP Address:

In the `.network` file, add or modify the following sections:

```
[Match]
Name=eth0

[Network]
Address=192.168.1.100/24
Gateway=192.168.1.1
DNS=8.8.8.8
DNS=8.8.4.4
LinkLocalAddressing=no
IPv6AcceptRA=no
```

Replace `eth0` with your actual network interface name.

Modify the `Address` with your new IP and subnet mask (e.g., `/24` for a 255.255.255.0 netmask).

Set the `Gateway` and `DNS` entries as per your network configuration.

You'll also want to edit `/etc/network/interfaces` because `ens192` is configured there for DHCP. That's how you might have gotten to it via Webmin for instance. Edit the file to put a `#` before the 2 lines that have `ens192` on them. Those 2 lines in the file will look like this after editing:

```
# The primary network interface
#allow-hotplug ens192
#iface ens192 inet dhcp
```

4. Reload and Restart systemd-networkd:

After making changes, enable the Networkd service so interfaces come up at boot time, and reload the daemon and

restart the network:

```
sudo systemctl enable systemd-networkd
sudo systemctl daemon-reload
sudo systemctl restart systemd-networkd
```

5. Verification:

Check the status of your network interface to ensure the new settings are active:

```
networkctl status eth0
```

You can also use `ip addr show eth0` to view the IP configuration.

▼ CentOS Linux

CentOS Linux

Depending if you are using the appliance we offer for a CentOS Linux virtual appliance or a Linux machine you built the steps may be slightly different. The steps shown below will be for the FileWave virtual appliance that we offer.

1. Configure the "ifcfg-ens160" file on the server. (This file will be different if you are not using our Virtual Appliance and will have a different name like "ifcfg-eth1" for example)

```
vi /etc/sysconfig/network-scripts/ifcfg-ens160
```

2. Change/add the following values of the file.
 1. Change BOOTPROTO=none
 2. Add "IPADDR", "NETMASK", "GATEWAY", "DNS1" to the file with your network configurations. I attached a screen shot of a completed file below. (If you want to add more then one DNS server you can add DNS2, etc to the file)

```
DEVICE=ens160
BOOTPROTO=none
ONBOOT=yes
TYPE=Ethernet
IPADDR=10.1.10.188
NETMASK=255.255.0.0
GATEWAY=10.1.0.1
DNS1=10.1.10.25
```

~

3. Save the file using "esc" then ":wq"
3. Now you will need to restart the network services on the server.

```
/etc/init.d/network restart
```

Booster ID Generation

Description

In some circumstances, a Booster may appear as missing or multiple boosters may overwrite a single Booster record in the Admin console. This can occur where each booster is a copy of another booster, e.g. VM source is identical.

Where Booster IDs conflict, it generating a new Booster ID will be necessary

Directions

Running the following command through a Booster shell should force a Booster to configure a new Booster ID.

Linux & macOS

```
fwcontrol booster stop && rm -f /etc/xdg/FileWave/Booster.conf && rm -f /var/FWBooster/*.key && rm -f /var/FWBooster/*.crt && fwcontrol booster start
```

It will then be necessary to 'Enroll Booster(s)' from their contextual menu, even if previously enrolled.

Enable core dumps for booster crashes

For CentOS

Step-by-step guide

1. edit this file:
`vi /usr/local/etc/filewave/supervisor/supervisord-booster.conf`
2. Uncomment this line:
`command=/bin/bash -c "ulimit -c unlimited && /usr/local/sbin/fwbooster"`
3. restart the booster:
`sudo /etc/init.d/fw-booster restart`
4. edit this file:
`vi /etc/sysctl.conf`
5. Add the lines below:
`kernel.core_uses_pid = 1`
`kernel.core_pattern = /tmp/core-%e-%s-%u-%g-%p-%t`
`fs.suid_dumpable = 2`
6. Enable debugging:
`echo "DAEMON_COREFILE_LIMIT='unlimited'" >> /etc/sysconfig/init`
7. Reload the settings in /etc/sysctl.conf:
`sysctl -p`

For testing purpose:
Kill booster process e.g:
(`ps ax | grep fw`)
`kill -6 3014`

Check if a core dump is generated under /tmp.
Should be something like this:
core-fwbooster-6-0-0-8440-1509016449