

How to set up an infrastructure with a 4PSA DNS Manager server and several hosting servers

This article describes how to set up a redundant DNS infrastructure for multiple hosting servers.

Infrastructure architecture

The infrastructure described in this example has the following architecture:

- One 4PSA DNS Manager server will act as the DNS slave server for all the hosting servers.
- The hosting servers (Plesk, Cpanel, Ensim, etc.) will act as the master DNS servers.

The 4PSA DNS Manager will automatically fetch the zones list from every server, so the domains added from the control panel interface will be replicated on the 4PSA DNS Manager server and will have a slave zone.

Requirements

The recommended 4PSA DNS Manager configuration for this example is:

- **Server OS:** Redhat Linux, Redhat Enterprise Linux, Fedora Core, FreeBSD 4.9
- **Server location:** another network
- **Server hardware:** a low end server or a VPS that can work with more than 50,000 zones
- **Control panel:** no control panel installed

The exemplification will be made using a Plesk for *nix server.

Assumptions

For this example, the following assumptions are made:

- **DNS1** is the 4PSA DNS Manager slave server and has the `<slave_ip_addr>` IP address.
- **CLIENT1** is a Plesk 7 for *nix server and has the `<plesk_client_ip_addr>` IP address.

Set up the Plesk server to dump the zones list

The steps that have to be followed to set up the Plesk server to dump the zones list are:

STEP 1: Log in as `root` to the **CLIENT1** server using your favorite SSH console (e.g.: Putty).

STEP 2: Copy the 4PSA DNS Manager integration tools for the specified control panel (in this case, Plesk for *nix) by running:

```
client1# mkdir -p /usr/local/dnsmanager
client1# scp -r -p <slave_ip_addr>:/usr/local/dnsmanager/remote/plesk/plesk_export.sh /usr/local/dnsmanager
```

STEP 3: Provide the **DNS1** root password in order to complete the transfer:

```
client1# chmod 700 /usr/local/dnsmanager/plesk_zones.sh
```

STEP 4: Set up the zones dump cron job. This cronj job will dump the list of the zones on your server to a HTTP accessible location. To do so, run:

```
client1# crontab -e
```

STEP 5: Add the following line to cron:

```
* /50 * * * * /usr/local/dnsmanager/plesk_zones.sh path_to_web_srv_root_dir/c11_dump_zones.txt >/dev/null 2>&1
```

where `path_to_web_srv_root_dir` is the path where the Apache web server stores its public files plus the directory on your domain where you want the dump to be made.

Example: if the domain where you want to save the dump is `my_plesk.com`, then `path_to_web_srv_root_dir` can be `'/home/httpd/vhosts/my_plesk.com/httpdocs/dnsmanager_dump'`. This directory might be different on your Plesk server.

Make sure that the `/home/httpd/vhosts/my_plesk.com/httpdocs/dnsmanager_dump` directory exists. Otherwise, the script will not be able to save the dump file.

Do not forget to save the cron file!

STEP 6: Edit the `named.conf` file to permit zone transfers from the 4PSA DNS Manager server. In `common-allow-transfer` named ACL, add the 4PSA DNS Manager server's IP address, like in the example below:

```
acl common-allow-transfer {
    <slave_ip_addr>
};
```

STEP 7: Since Plesk uses its database to write the `named.conf` file, the best solution is to add the 4PSA DNS Manager server's IP address to the 4PSA database as well. In this way, you can be sure that the `named.conf` file will remain correct. To do so, run:

```
client1# mysql -uadmin -p`cat /ec/psa/.psa.shadow` dnsmanager -e"INSERT INTO misc VALUES
('DNS_Allow_TransferXX\','<slave_ip_addr>');"
```

where:

- `xx` is a unique number (increment it to add more IP addresses).
- `<slave_ip_addr>` is the 4PSA DNS Manager server's IP address.

STEP 8: Restart the `named` service:

```
/etc/init.d/named restart
```

Setting up the 4PSA DNS Manager server to grab the zones list from the Plesk server

The steps that have to be followed to set up the 4PSA DNS Manager server to grab the zones list previously dumped by the Plesk server are:

STEP 1: Using the 4PSA DNS Manager interface, you have to add a **Remote update** source for the dump location you have just set up. To do so, follow the next steps:

STEP 2: Go to the **Clients > {client_name} > Remote Update Locations** page.

STEP 3: Using the controls available in the **New Remote Update Location** fieldset, define a location with the following properties:

- **Remote update location** - Fill in http://my.plesk.com/path_to_dump/c11_dump_zones.txt. This is the location where the cron job you have previously set up dumps the zones list file. You must verify that the zones list actually exists at this location.
- **Keep minimum <x> minutes between updates** - Use the drop-down list and select 60.
- **Remote URL priority** - If you wish, you can select the priority of the zone definitions downloaded from this URL over other (duplicate) zone definitions.

If the specified location is password protected, then you must provide the appropriate HTTP, HTTPS or FTP authentication **Username** and **Password**.

STEP 4: Click **OK** to add the remote location.

In order to integrate multiple servers, you must repeat the above process for all your servers.

Related articles

- [Primary and secondary server setup for 4PSA DNS Manager](#)
- [How to install a LetsEncrypt SSL certificate in VoipNow](#)
- [How to remove the country code field from the CallNow interface](#)
- [How to use the CallNow feature](#)
- [How to debug 504 gateway timeout](#)