

Creating NTP+DNS Nodes on linode.com

version 1.0.1

Dr. Simon A. Roberts

[simonxaies@gmail.com](mailto:simonxaies@gmail.com)

This document is the instructions to creating **powerdns** and **ntp services** on the **linode.com exchange services**. It covers both the primary node as well as basis of sub-nodes for **DNS + NTP services** on Ubuntu Server.

Table of Contents

[Creating the DNS Primary Service 2](#__RefHeading___Toc113_1456449258)

[Installing Zones-API-PHP Library 2](#__RefHeading___Toc115_1456449258)

[Cronjobs to Set on the Primary DNS Service 3](#__RefHeading___Toc117_1456449258)

[Start-on-boot to Set on the Primary DNS Service 3](#__RefHeading___Toc287_4118114418)

[SH Shell Batch Scripts to Create on Primary DNS Service 3](#__RefHeading___Toc289_4118114418)

[SuperMaster’s to Set on the Primary DNS Service 4](#__RefHeading___Toc291_4118114418)

[Creating the DNS Secondary Service 4](#__RefHeading___Toc306_4118114418)

[Cronjobs to Set on the Secondary DNS Service 5](#__RefHeading___Toc117_14564492581)

[Start-on-boot to Set on the Secondary DNS Service 5](#__RefHeading___Toc287_41181144181)

[SH Shell Batch Scripts to Create on Secondary DNS Service 5](#__RefHeading___Toc289_41181144181)

# Creating the DNS Primary Service

This is the primary dns service ie. william.snails.email / hostmaster.snails.email + 0.ntp.snails.email – first propagate your Ubuntu Node to the specifications you need and want to provide for the primary service!

Run the following on the shell:

$ apt-get install tasksel ntp git nano -y

After they have install run ‘tasksel’ and install LAMP services (Do not install DNS Services).

Now run the following on the shell:

$ apt-get install pdns-server pdns-backend-mysql -y

$ unlink /etc/powerdns/pdns.d/bind.conf

**Change local-address as well as local-address-ipv6 in /etc/powerdns/pdns.conf to the public IP Address for the DNS. As well as Answer Any tc=0 to true and supported!**

**Check the wiki for any other help references:** <https://github.com/Chronolabs-Cooperative/Zones-API-PHP/wiki>

## Installing Zones-API-PHP Library

Say we want the REST API on [http://zones.william.snails.email](http://zones.william.snails.email/) then this is where the example is going to be placed.

Firstly goto the shell and execute the following command

$ git clone <https://github.com/Chronolabs-Cooperative/Zones-API-PHP.git> /var/www/zones.william.snails.email

Now we have to configure apache 2 for the site execute the following command

$ nano /etc/apache2/sites-available/zones.william.snails.email.conf

and now insert the following configuration

<VirtualHost \*:80>

ServerName zones.william.snails.email

ServerAdmin simon@snails.email

DocumentRoot /var/www/zones.william.snails.email

ErrorLog ${APACHE\_LOG\_DIR}/zones.william.snails.email-error.log

CustomLog ${APACHE\_LOG\_DIR}/zones.william.snails.email-access.log combined

<Directory /var/www/zones.william.snails.email>

Options Indexes FollowSymLinks MultiViews

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

Now execute the following commands at the shell:

$ a2ensite zones.william.snails.email

$ service apache2 reload

You may want to make a username rather than use root all the time with mysql; but the target database is ‘pdns’ always! You want to now goto [http://zones.william.snails.email](http://zones.william.snails.email/) and run the install into the target database.

## Cronjobs to Set on the Primary DNS Service

Editing the crontabs file allows for commands to be executed on scheduled times like restarting services. Now we have to configure apache 2 for the site execute the following command

$ crontab -e

Add the following lines to the crontab

\*/41 \*/3 \* \* \* service ntp restart

\*/11 \*/7 \* \* \* sh /var/replace-ntp.conf.sh

\*/13 \*/13 \* \* \* service pdns restart

## MySQL Host Address: localhost – User: root – Password: PuppyD0ggy

\*/15 \* \* \* \* mysqldump --add-drop-database --add-locks --single-transaction --extended-insert --host=localhost -uroot -pPuppyD0ggy --result-file=/var/www/zones.william.snails.email/pdns.sql pdns

## Start-on-boot to Set on the Primary DNS Service

Editing the rc.local file allows for commands to be executed on boot like starting services. Now we have to configure apache 2 for the site execute the following command

$ nano /etc/rc.local

Edit the following two lines into rc.local

service pdns start

service ntp start

## SH Shell Batch Scripts to Create on Primary DNS Service

Editing the replace-ntp.conf.sh file allows for commands to be executed in a script. Now we have to create and edit the file \*.sh script; execute the following command

$ nano /var/replace-ntp.conf.sh

Now insert the following lines and save and exit

cd /tmp

rm -vf ntp.conf

wget http://ntp.snails.email:80/v1/ntp.conf

if [ -s "/tmp/ntp.conf" ]

then

rm -vf /etc/ntp.conf

mv /tmp/ntp.conf /etc

chmod -fv 0644 /etc/ntp.conf

else

echo " ntp.conf was returned empty or not existing! "

fi

## End Script

## SuperMaster’s to Set on the Primary DNS Service

You will require several open dns as super master DNS Services, to enter these goto [http://zones.william.snails.email](http://zones.william.snails.email/) and log into generate an ‘authkey’ now go back to the Zones API Forms and refresh the page so the ‘authkey’ populates in the form from the PHP Session Variables; now goto the SuperMaster section and enter the following Ipv4 or your own if you have some:-

* **8.8.8.8**
* **8.8.4.4**
* **7.235.1.174**
* **64.81.79.2**
* **61.9.211.34**
* **61.9.211.21**
* **4.2.2.6**
* **4.2.2.5**
* **4.2.2.4**
* **4.2.2.3**
* **4.2.2.2**
* **4.2.2.1**
* **216.158.234.243**
* **216.158.228.164**
* **212.71.253.5**
* **212.71.252.5**
* **199.231.191.75**

# Creating the DNS Secondary Service

This is the seondary and further dns service ie. jester.snails.email + 1.ntp.snails.email – first propagate your Ubuntu Node to the specifications you need and want to provide for the primary service!

Run the following on the shell:

$ apt-get install ntp nano -y

Now run the following on the shell:

$ apt-get install pdns-server pdns-backend-mysql -y

$ unlink /etc/powerdns/pdns.d/bind.conf

**Change local-address as well as local-address-ipv6 in /etc/powerdns/pdns.conf to the public IP Address for the DNS. As well as Answer Any tc=0 to true and supported!**

**Check the wiki for any other help references:** <https://github.com/Chronolabs-Cooperative/Zones-API-PHP/wiki>

## Cronjobs to Set on the Secondary DNS Service

Editing the crontabs file allows for commands to be executed on scheduled times like restarting services. Now we have to configure apache 2 for the site execute the following command

$ crontab -e

Add the following lines to the crontab

\*/41 \*/3 \* \* \* service ntp restart

\*/11 \*/7 \* \* \* sh /var/replace-ntp.conf.sh

\*/15 \* \* \* \* sh /var/replace-db-pdns.sh

\*/13 \*/13 \* \* \* service pdns restart

## Start-on-boot to Set on the Secondary DNS Service

Editing the rc.local file allows for commands to be executed on boot like starting services. Now we have to configure apache 2 for the site execute the following command

$ nano /etc/rc.local

Edit the following two lines into rc.local

service pdns start

service ntp start

## SH Shell Batch Scripts to Create on Secondary DNS Service

Editing the **replace-ntp.conf.sh** file allows for commands to be executed in a script. Now we have to create and edit the file \*.sh script; execute the following command

$ nano /var/replace-ntp.conf.sh

Now insert the following lines and save and exit

## Start Script

cd /tmp

rm -vf ntp.conf

wget http://ntp.snails.email:80/v1/ntp.conf

if [ -s "/tmp/ntp.conf" ]

then

rm -vf /etc/ntp.conf

mv /tmp/ntp.conf /etc

chmod -fv 0644 /etc/ntp.conf

else

echo " ntp.conf was returned empty or not existing! "

fi

## End Script

Editing the **replace-db-pdns.sh** file allows for commands to be executed in a script. Now we have to create and edit the file \*.sh script; execute the following command

$ nano /var/replace-db-pdns.sh

Now insert the following lines and save and exit

## Start Script

##

## MySQL Host Address: localhost – User: root – Password: PuppyD0ggy

##

cd /tmp

rm -vf pdns.sql

wget http://zones.william.snails.email:80/pdns.sql

if [ -s "/tmp/pdns.sql" ]

then

mysql –-database=pdns --host=localhost -uroot -pPuppyD0ggy < /tmp/pdns.sql

else

echo " pdns.sql was returned empty or not existing! "

fi

## End Script