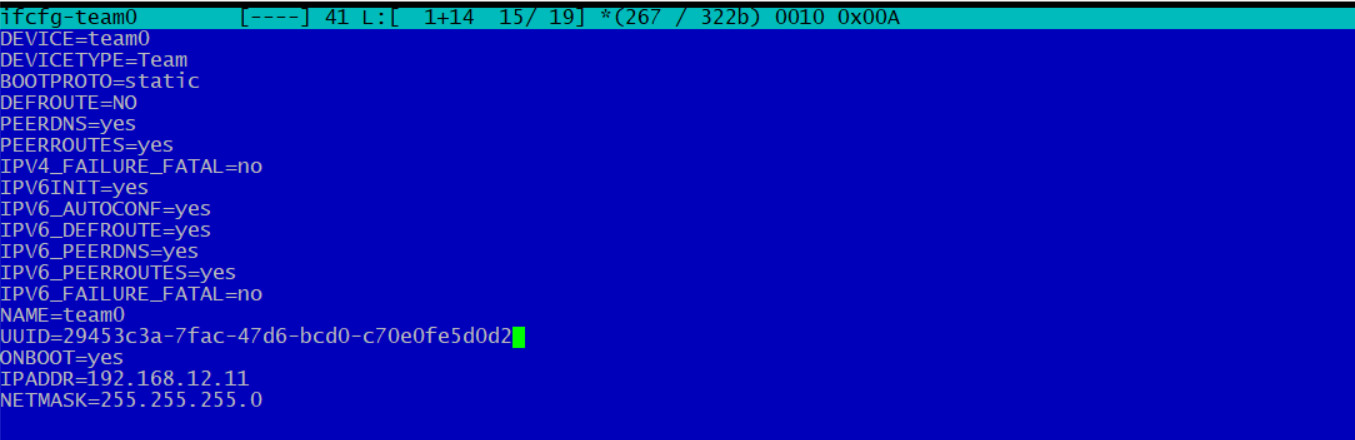
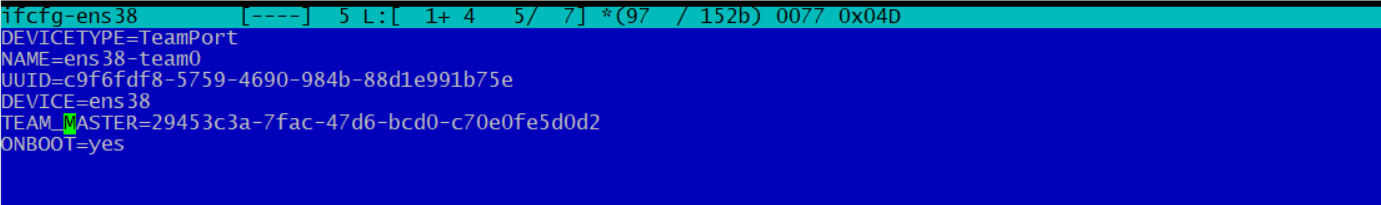
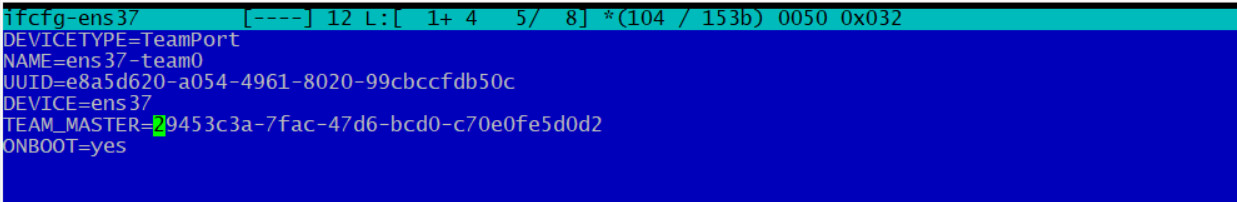
1. **Настроить nic teaming между двумя интерфейсами — server1 и server2. Подсеть 192.168.12.0/24 будет находиться теперь на team0-интерфейсе.**

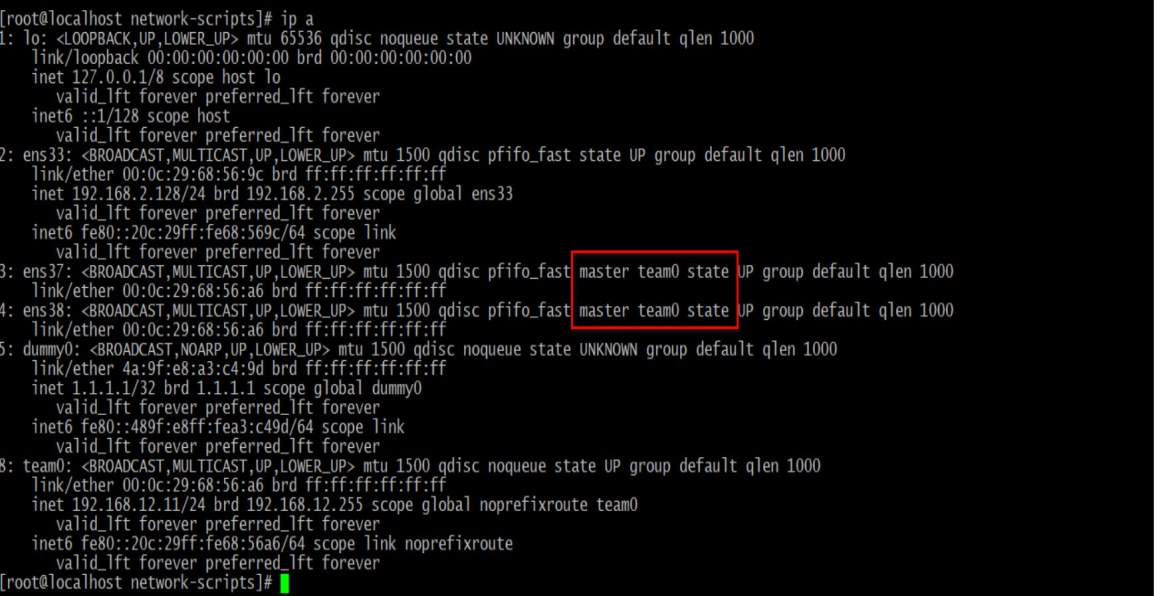
Добавляем интерфейс team0 на оба сервера



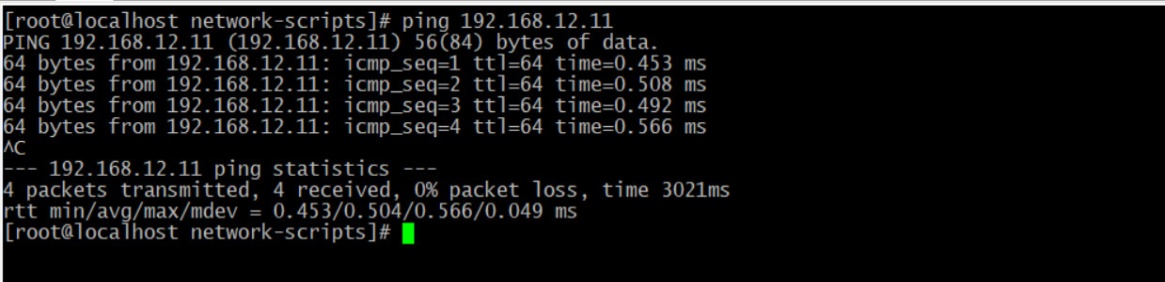
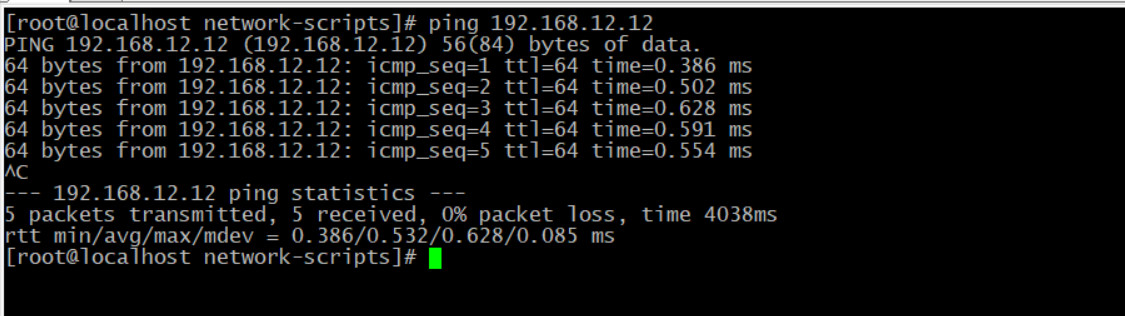
Меняем настройки ens37 и ens38 добавляем в team0



Проверяем что получилось

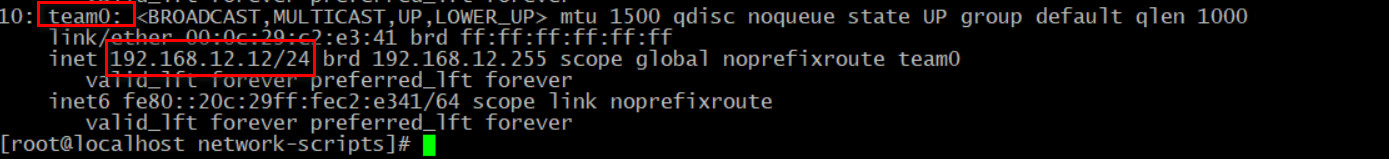


Проверяем на пинг



1. **На интерфейсе team0 сервера server2 назначить статический IP из подсети 192.168.12.0/24.**

Задали IP из целевой подсети



1. **На сервере server2 настроить DHCP-сервер для выдачи динамического IP-адреса интерфейсу team0 сервера server1, а также IP-адрес DNS-сервера 3.3.3.3.**

Устанавливаем yum install dhcp –y

Настраиваем файл /etc/dhcp/dhcpd.conf

subnet 192.168.12.0 netmask 255.255.255.0 {

range 192.168.12.10 192.168.12.12;

option domain-name-servers 3.3.3.3;

# option domain-name "dmosk.local";

# option routers 192.168.0.1;

# option broadcast-address 192.168.0.255;

default-lease-time 600;

max-lease-time 7200;

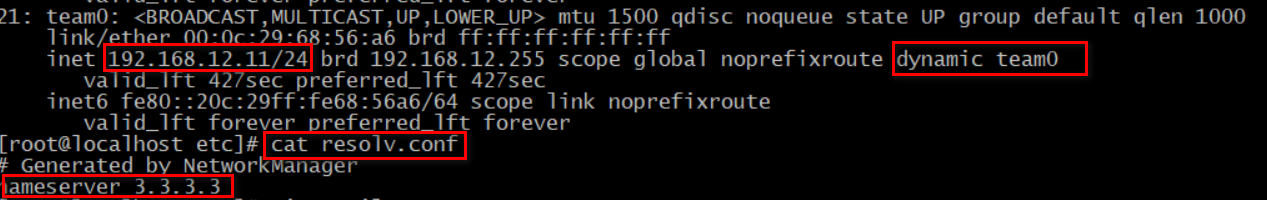
}

Добавляем правила firewall

firewall-cmd --permanent --add-service=dhcp

Меняем настройки для интерфейса team0 bootproto = ‘dhcp’

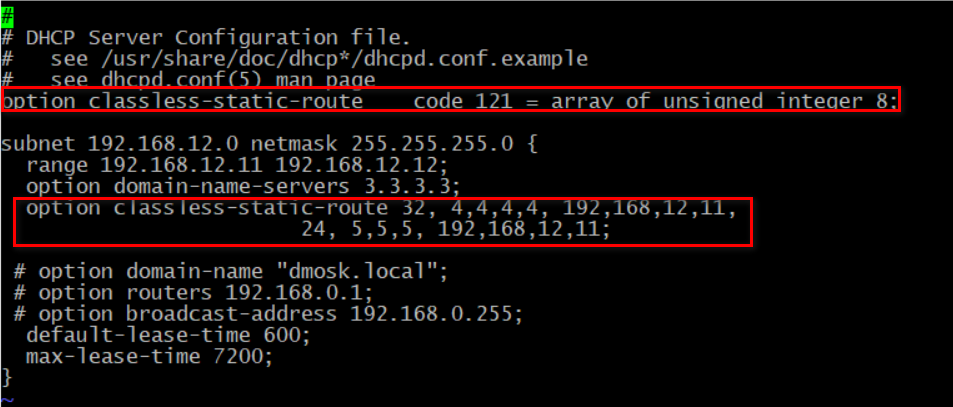
Перезагружаем интерфейс и проверяем



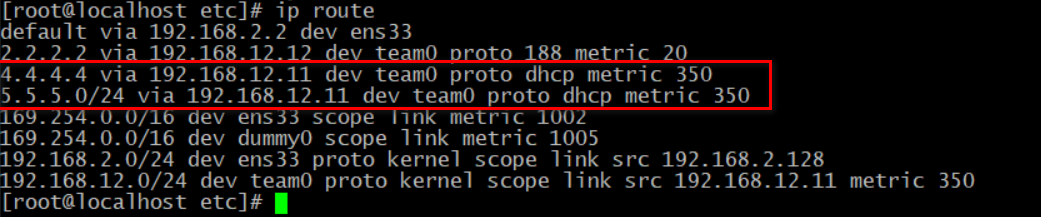
Работает.

1. **При помощи DHCP выдать серверу Server1 2 статических маршрута 4.4.4.4/32 и 5.5.5.0/24 с next hop интерфейса team0 на сервере server2**

Для того что бы выдать маршруты клиент добавляем в DHCP строки /etc/dhcp/dhcpd.conf



Перезагрузим интерфейс и проверяем, что маршруты были выданы клиенту



1. **Настроить DNS-сервер для зоны example.com на сервере server3. Создать прямую и обратную зоны, а также несколько записей с разными RR. Убедиться, что только запросы на IP-адрес 3.3.3.3 будут обслуживаться этим DNS-сервером.**

Настраиваем vi /etc/named.conf

На каком адресе будет слушать DNS

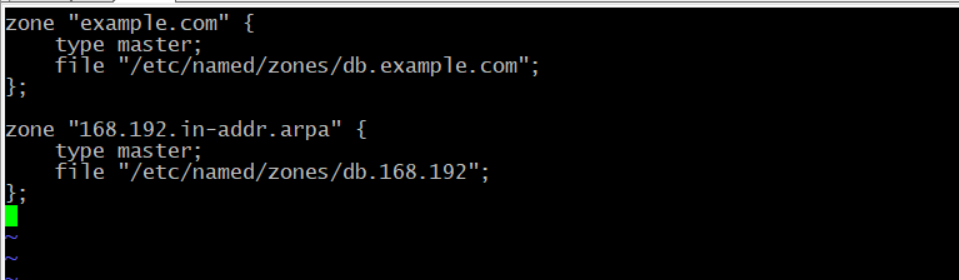
options {

listen-on port 53 { 127.0.0.1;3.3.3.3;};

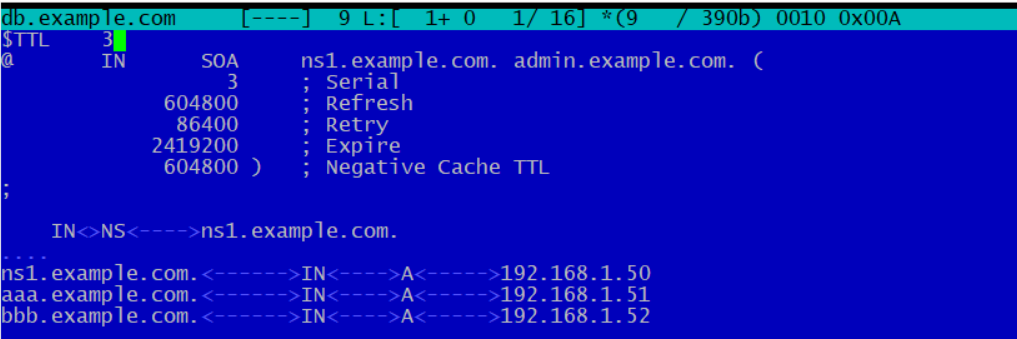
добавляем файл зон

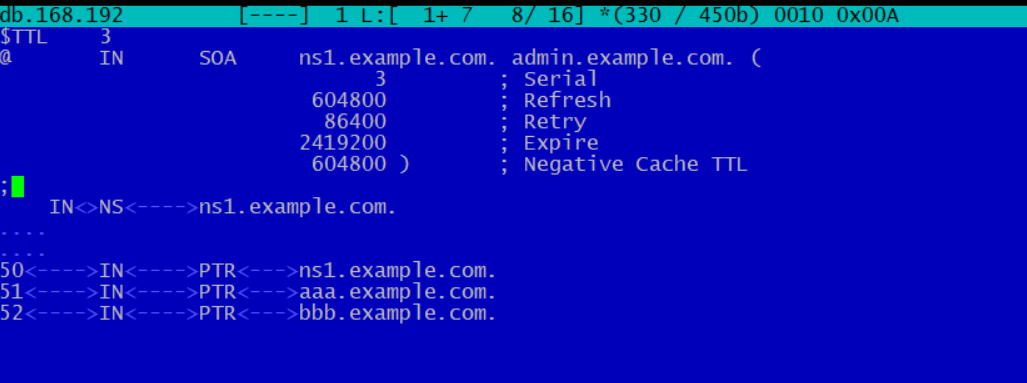
include "/etc/named/named.conf.local"

Создаём и добавляем файлы конфига зон "/etc/named/named.conf.local\

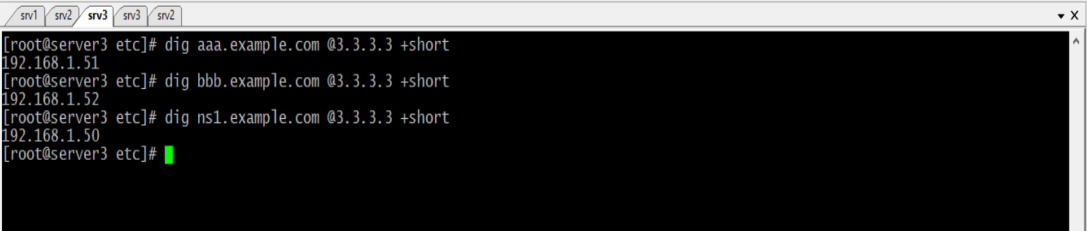


Настраиваем файлы конфига зон прямой и обратный

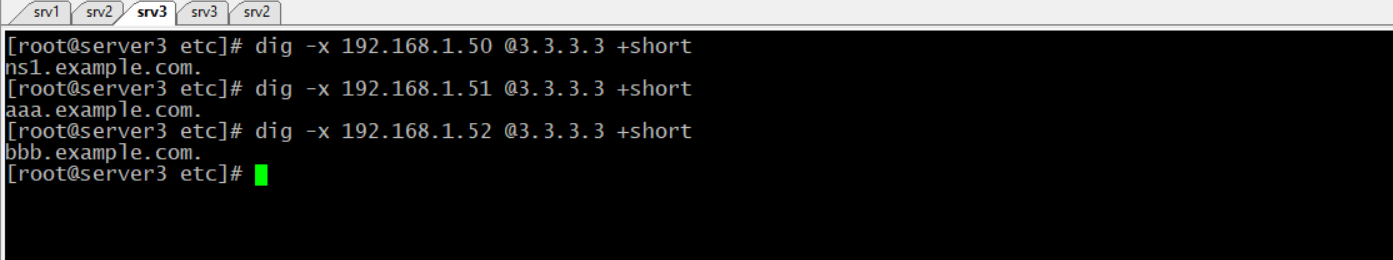




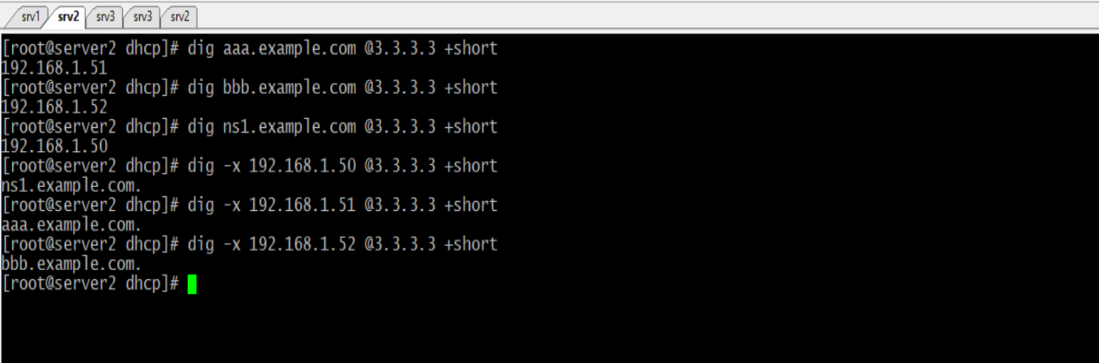
Перезапускаем проверяем что работает с помощью dig



Обратную зону



Пробуем с srv 2 (добавил пару маршрутов так как менялись настойки сети)

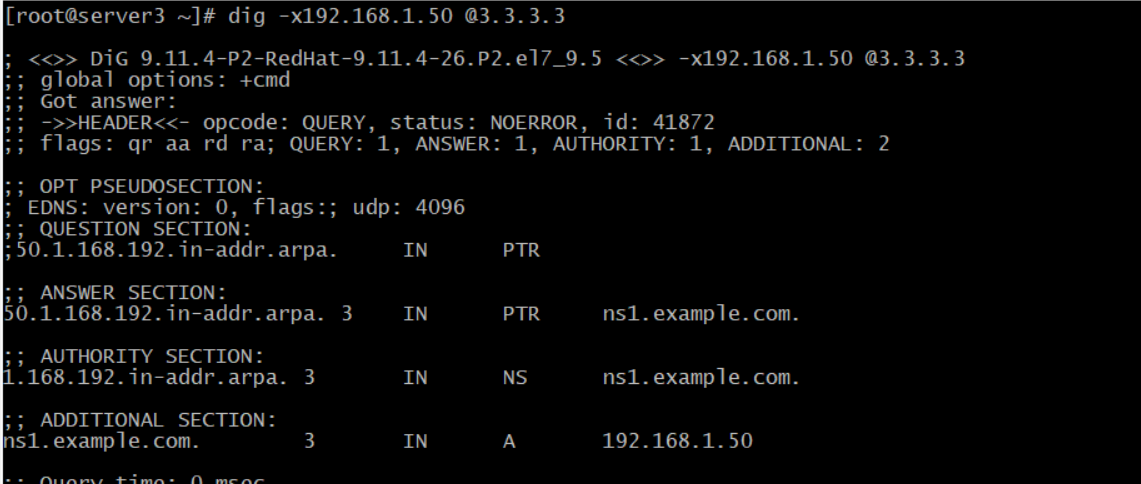


Что бы сервер принимал DNS запросы только на 3.3.3.3 делаем соответствующие настройки в /etc/named.conf, в listen-on port записываем нужные адрес.

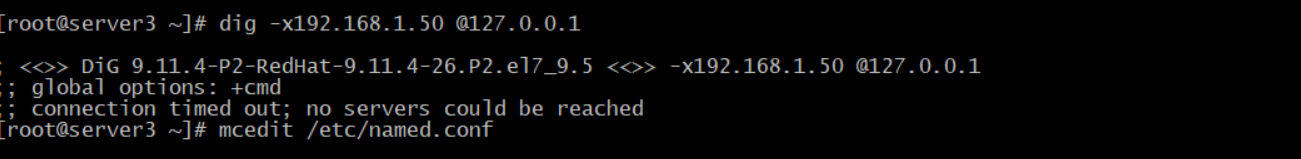


Проверяем через dig

Запрос на 3.3.3.3



Запрос на 127.0.0.1 соответственно не отрабатывает



**6. Настроить фаерволл на серверах server2 и server3, чтобы разрешить только соответствующие запросы (DHCP/DNS).**

Для корректной работы DNS с Firewalld добавляем правила

**firewall-cmd --zone=public --add-service=dns –permanent**

**firewall-cmd --zone=public --add-service=dns**

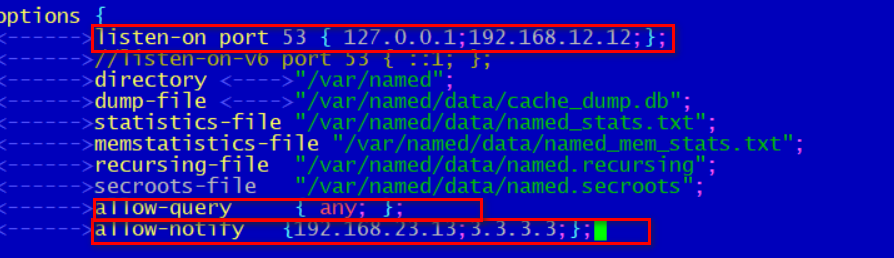
Для корректной работы DHCP с Firewalld добавляем правило

firewall-cmd --permanent --add-service=dhcp

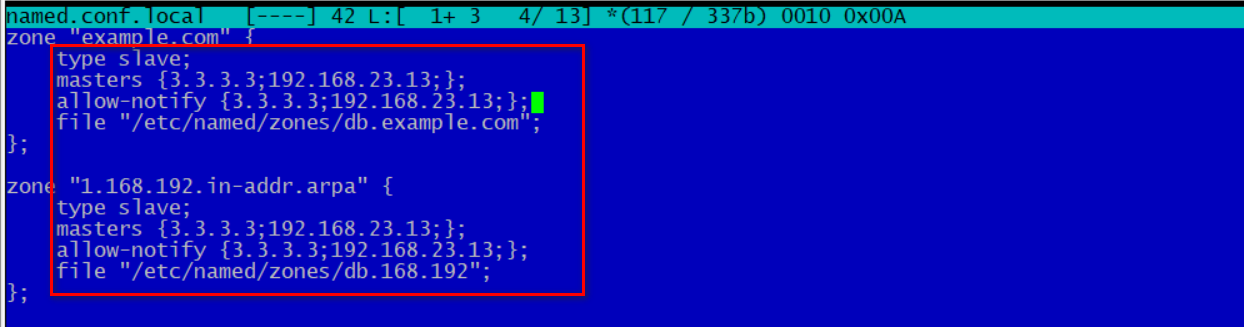
**7. \* Настроить slave для DNS-сервера server3. Убедиться, что репликация записей происходит**

Настройка slave:

Настраиваем etc/named.conf



Настраиваем зоны etc/named/named.conf.local указываем что зоны slave



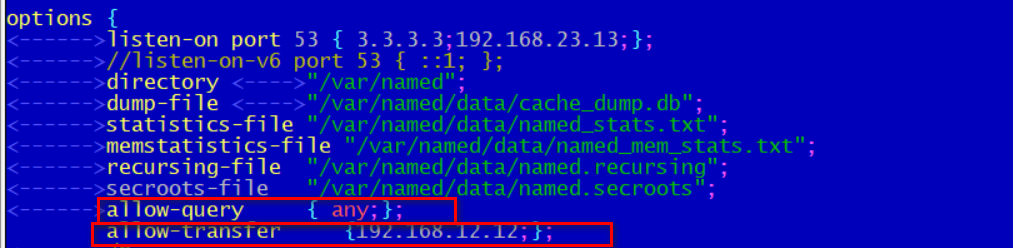
Указываем type slave – означает что зона slave, masters – перечисляются мастер сервера,

Notify – сервера от которых можно принимать оповещение.

Дополнительно нужно создать целевые директории и назначить владельцем пользователя named.

Так же отключить SElinux, так как будут возникать ошибки доступа к файлам.

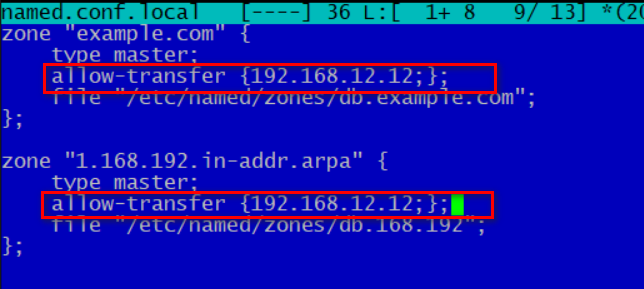
Дополнительные настройки со стороны мастер



Allow-query – не обязательно any разрешает запросы со всех хостов, можно указать отдельные хосты

Allow-transfer – указывает slave куда будет отправляется информация об изменения зоны.

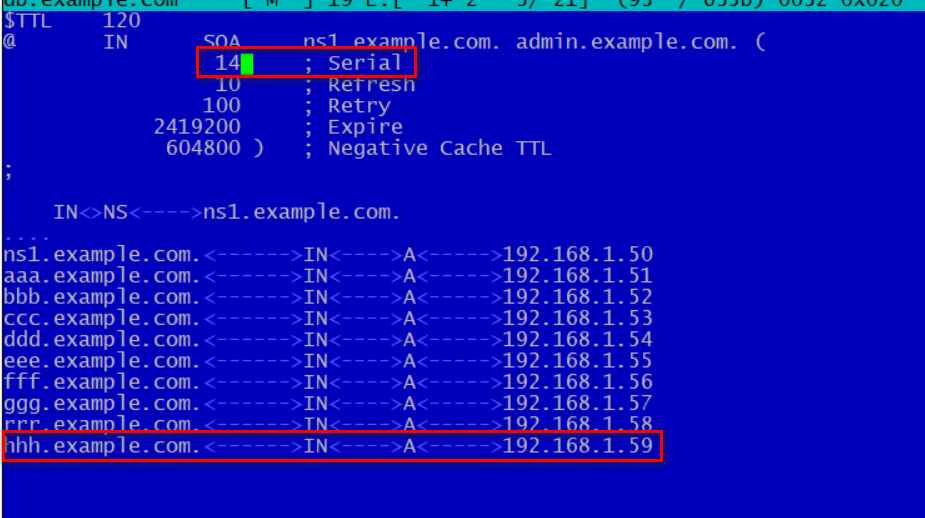
Для файла зон



Аналогично Allow-transfer – указывает slave куда будет отправляется информация об изменения зоны.

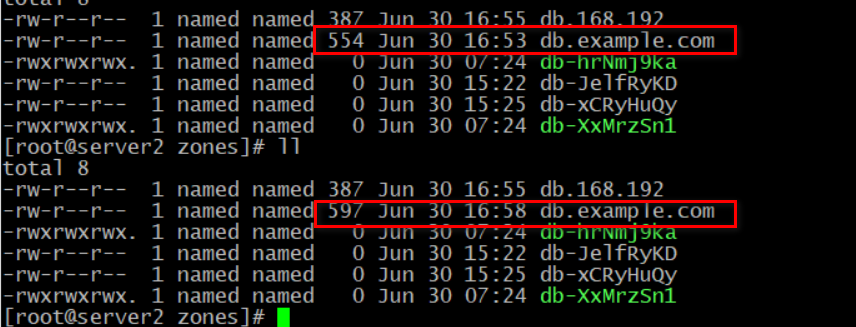
В целом всё проверяем:

В файле RR на мастер добавляем A запись hhh.example.com и меняем serial



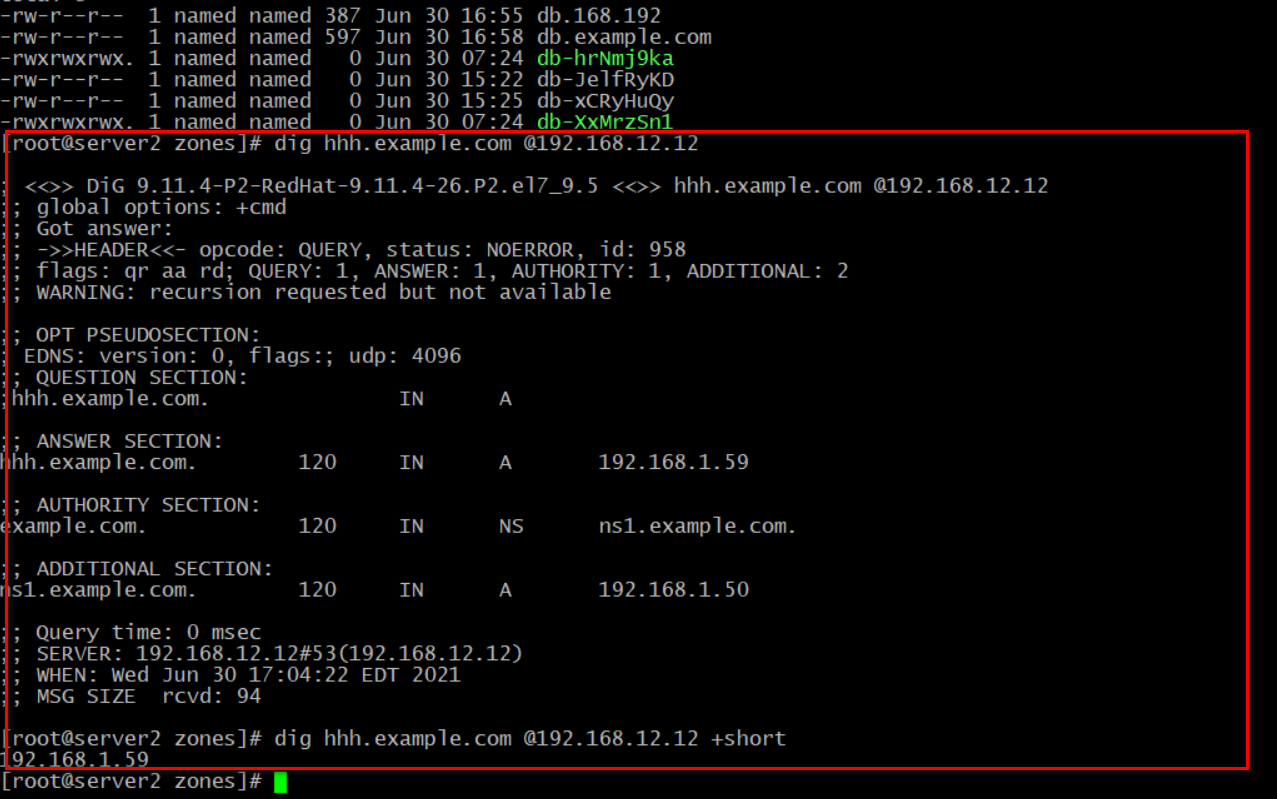
Идём на slave и проверяет, что он резолвит добавленную запись.

Для наглядности на slave можно сравнить предыдущий размер и время файла с размером после обновления



Как видно размер файла на slave изменился после добавления записи

Можно резолвить со slave



Работает.