DNS Server Objectives Tasks in PDF file

DNS Objectives

1. Configure cache only DNS server on **server.example.com**
   a> DNS should allow access to all Hosts on **192.168.122.0/24** only.
   b> Configure the firewall to accept DNS queries
   c> Bind the DNS server to listen on eth0 interface
   d> DNS should forward the queries to ipaserver.example.com in case entry in cache is missing.

Commands: (On server.example.com)

yum install unbound (To install unbound Service to implement cache only DNS server)

systemctl start unbound (To start Service)

systemctl enable unbound (To enable service)

systemctl status unbound (To check the status service)

vim /etc/unbound/unbound.conf **(To make changes in config file to configure Cache only DNS Server)**

interface 192.168.122.10 (To bind the DNS server to listen on eth0 interface)

access-control: 192.168.122.0/24 allow (To allow access for clients on 192.168.122.0/24 network)

forward-zone: (To forward the DNS queries to DNS Server configured on IPA Server)

name: " . "

forward-addr: 192.168.122.254

domain-insecure: "example.com" (To disable DNS security because DNS security is not configured, Don't forget to set this parameter)

systemctl restart unbound (Restart service to make the changes effective)

unbound-checkconf (To check the config file and ensure there is no mistake)

firewall-cmd --add-service=dns --permanent (To configure firewall to accept inbound DNS traffic)

firewall-cmd –reload (To reload the firewall to make the changes effective)

2. Set the DNS IP as **192.168.122.10** on **client2.example.com** which is IP of server.example.com on which cache only DNS is configured.

a> Perform some DNS queries and check the DNS cache on **server.example.com**

b> Disconnect the **ipaserver.example.com** machine and verify the queries are still successful from cached data.

Complete task 1 first

Commands: (On client2.example.com)

nmcli connection modify client2 ipv4.dns 192.168.122.10 (To set DNS IP of server.example.com on client2 machine)

systemctl restart network (To restart Network Service)

cat /etc/resolv.conf (To verify DNS address is set as 192.168.122.10,server.example.com)

Perform some DNS queries from client2 machine,which will be forwarded to IPA server DNS because server.example.com Cache only DNS does not have cached data yet.

if you will make queries again for same hostnames ,it will not be forwarded to IPA server DNS but will be replied from Cache of server DNS which is built as result of first query for each hostname

dig client.example.com

dig rhce.example.com

dig web.example.com

dig ipaserver.example.com

All above queries will be forwarded to IPA Server DNS and replied but server.example.com DNS will also save the result in cache to reply further queries for same hostname.

Now we will dump cache on server.example.com and verify the output. ….continued on next page

Commands: (On server.example.com)

unbound-control dump_cache **(To dump the cache build as result of DNS queries, You will find data for all the queries you made from client2 and replied from IPA Server)**

Now disconnect the IPA Server from the Network and make all queries again from client2 and they will be successful and will be replied from server.example.com DNS Cache.

dig client.example.com

dig rhce.example.com

dig web.example.com

dig ipaserver.example.com

unbound-control flush rhce.example.com **(To flush the entry for rhce.example.com from server DNS Cache)**

Again make query for rhce.example.com and now it will fail because cached entry is flushed and server DNS is not able to forward the request to IPA Server DNS which is disconnected. Rest queries will be still successful.

This is how Cache only DNS work.

3. Dump the DNS cache to **/root/dns_dump.txt** and verify the output.

a> Restart the unbound service on **server.example.com** and verify the DNS cache.

b> Load the DNS cache from **/root/dns_dump.txt** and again verify the cache

Complete tasks 1 and 2 first-

Commands: (On server.example.com)

unbound-control dump_cache > /root/dns_dump.text (To dump the cache to file)

more /root/dns_dump.txt (To verify the dumped data))

systemctl restart unbound (To restart Service,After restarting the service,all cached data will be lost)

unbound-control dump_cache (Dump the cache but it will be empty file)

vim /root/dns_dump.txt (Modify the record for rhce.example.com and map this to IP 192.168.122.40 replacing 192.168.122.10)

rhce.example.com 192.168.122.40 (This is not exact output snapshot but you will understand when you display the file)

:wq

unbound-control load_cache < /root/dns_dump.txt (To load the cache again to DNS SERVER)

unbound-control dump_cache (You will see restored cache)

Commands: (On client2.example.com)

Make DNS query for rhce.example.com and this time IP 192.168.122.40 will be returned back because we made changes in cached data. So wrong result is returned due to this and this is called cache poisoning.

dig rhce.example.com