DNS Firewall & DNS Sinkhole

DNS Firewall is a Security technology solution, protecting against volumetric, exploit and stealth attacks for both public and private DNS infrastructures.

DNS Firewall is the root of the Internet and protects from the attack at the root.

DNS Firewall governs/inspects DNS Queries at port 53.

DNS Firewall - **Mitigate At The Source** - Thwart initial infection and phishing.

DNS Firewall - **Adapt To Evolving Threat Landscape** - Threat Intelligence services to keep pace with malicious domains/IPs.

DNS Firewall - **Proactively Prevent New Attacks** - Detect and block malware communication with C&C (Command and Control) server.

DNS Firewall Installation Procedure:

```
dnf update -y
dnf -y install bind bind-utils
```

```
systemctl start named
systemctl enable named
systemctl status named
```

===Download the BIND_Configs Zip file from https://drive.google.com/file/d/18yksx8CF6hJH7ACPi0DThho8FuYj-WBU/view?usp=share_link
===From the downloaded zip file, you will have named.conf,urlwhitelist and urlblacklist db files.
===Upload named.conf file to /etc/ and overwrite the existing named.conf
===Upload urlwhitelist and urlblacklist db files to /var/named/ directory
===Edit /etc/named.conf file to configure the listening DNS IP, AWS instance public/private IP address in this case
===Edit /etc/named.conf file to configure the urlwhitelist and urlblacklist db mapping Zone files.

```
systemctl restart named
```

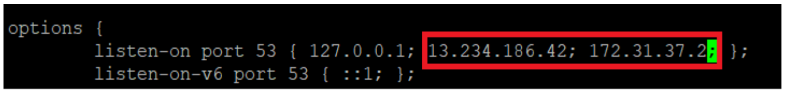
systemctl status named

Note : urlwhitelist is allowing (whitelist) DNS query file
Note : urlblacklist is blocking DNS query file

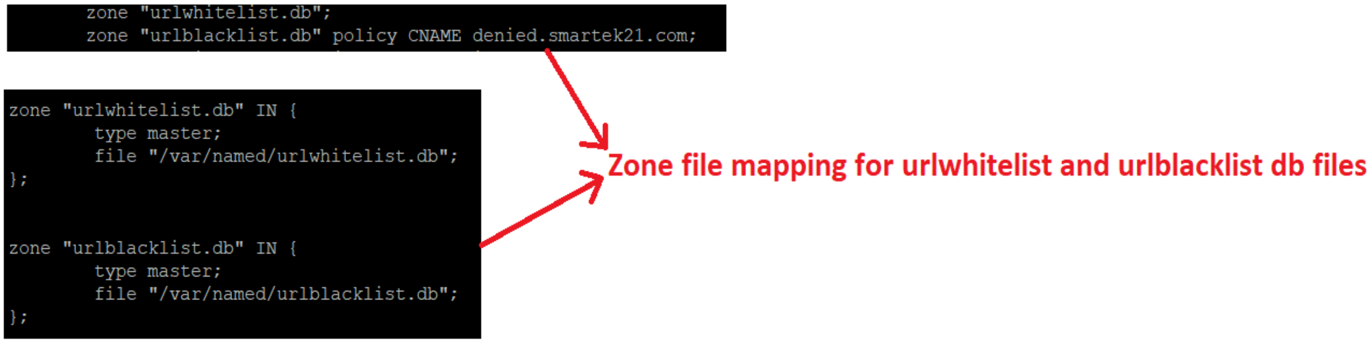
===Allow zone file query pattern domain.com IN CNAME rpz-passthru.
===Deny zone file query pattern domain.com IN CNAME @

Please refer the below for better understanding,

Edit the /etc/named.conf file



```
options {
        listen-on port 53 { 127.0.0.1; 13.234.186.42; 172.31.37.2; };
        listen-on-v6 port 53 { ::1; };
```

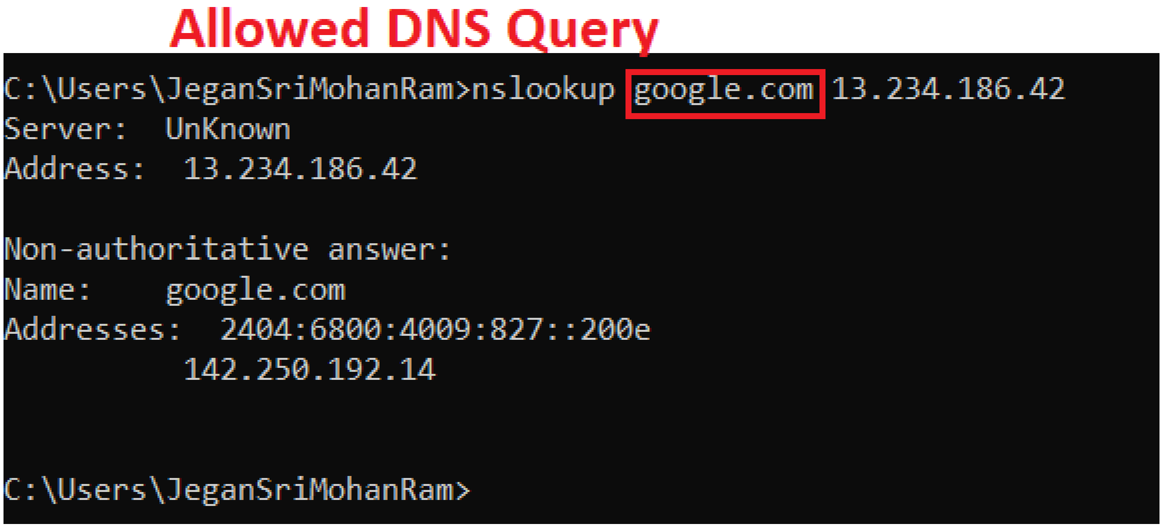


```
        zone "urlwhitelist.db";
        zone "urlblacklist.db" policy CNAME denied.smartek21.com;
```

```
zone "urlwhitelist.db" IN {
        type master;
        file "/var/named/urlwhitelist.db";
};

zone "urlblacklist.db" IN {
        type master;
        file "/var/named/urlblacklist.db";
};
```

Zone file mapping for urlwhitelist and urlblacklist db files

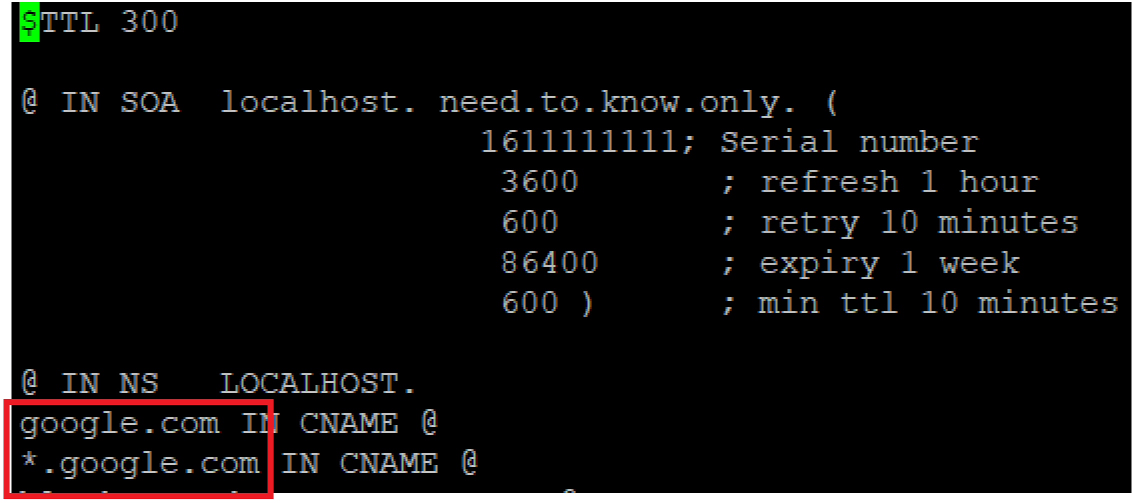
Testing the DNS Firewall,



Allowed DNS Query

```
C:\Users\JeganSriMohanRam>nslookup google.com 13.234.186.42
Server:  UnKnown
Address:  13.234.186.42

Non-authoritative answer:
Name:    google.com
Addresses:  2404:6800:4009:827::200e
          142.250.192.14


C:\Users\JeganSriMohanRam>
```

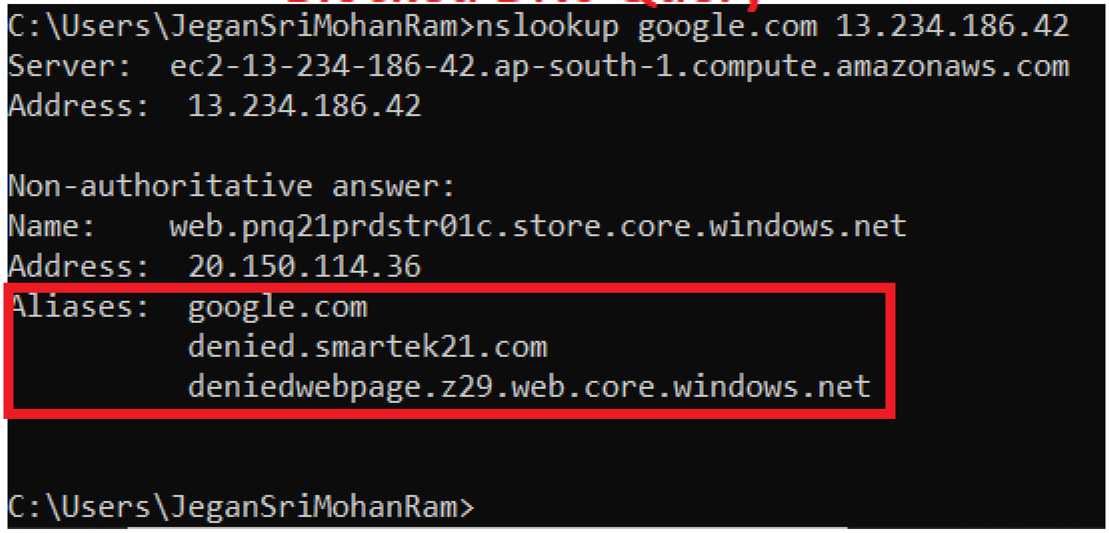
Edit the /var/named/urlblacklist.db file to block google.com,



urlblacklist db file

```
$TTL 300

@ IN SOA  localhost. need.to.know.only. (
                      1611111111; Serial number
                       3600       ; refresh 1 hour
                       600        ; retry 10 minutes
                       86400      ; expiry 1 week
                       600 )      ; min ttl 10 minutes

@ IN NS   LOCALHOST.
google.com IN CNAME @
*.google.com IN CNAME @
```

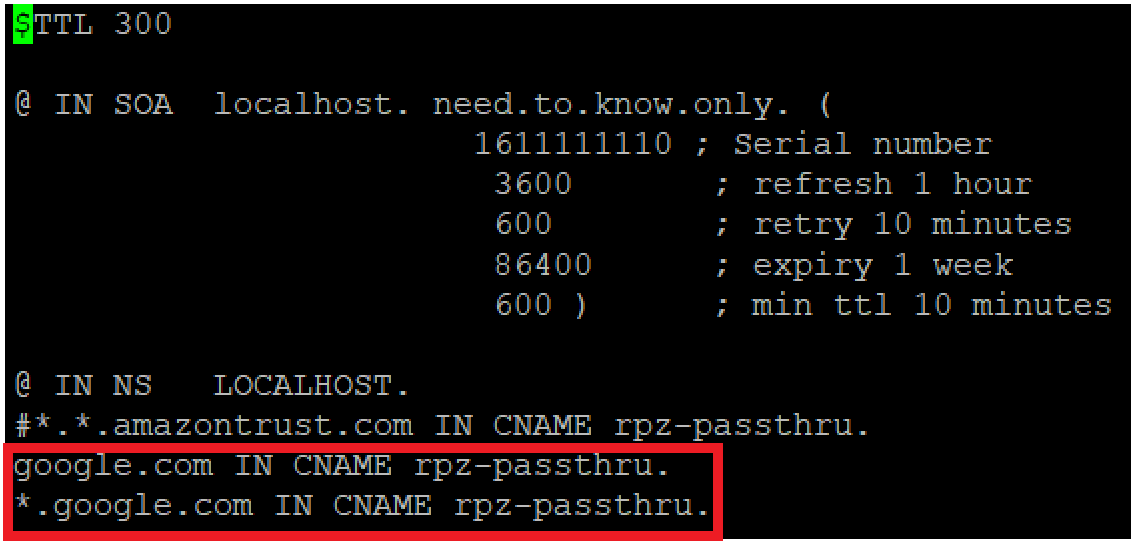
Now again test the DNS Firewall for google.com,it should be blocked



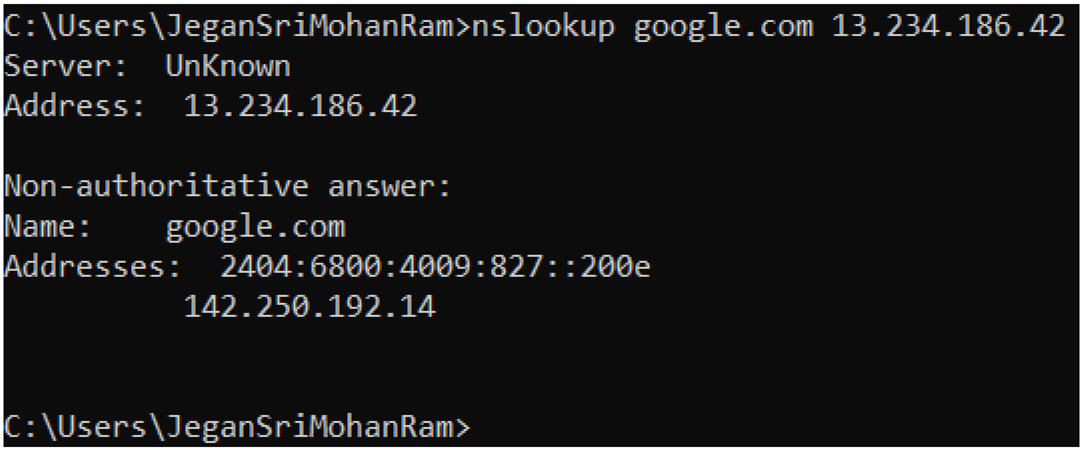
Blocked DNS Query

```
C:\Users\JeganSriMohanRam>nslookup google.com 13.234.186.42
Server:  ec2-13-234-186-42.ap-south-1.compute.amazonaws.com
Address:  13.234.186.42

Non-authoritative answer:
Name:    web.pnq21prdstr01c.store.core.windows.net
Address:  20.150.114.36
Aliases:  google.com
          denied.smartek21.com
          deniedwebpage.z29.web.core.windows.net


C:\Users\JeganSriMohanRam>
```

Edit the /var/named/urlwhitelist.db file to unblock google.com, urlwhitelist db zone file



```
$TTL 300

@ IN SOA  localhost. need.to.know.only. (
                      1611111110 ; Serial number
                       3600        ; refresh 1 hour
                       600         ; retry 10 minutes
                       86400       ; expiry 1 week
                       600 )       ; min ttl 10 minutes

@ IN NS   LOCALHOST.
#*.*.amazontrust.com IN CNAME rpz-passthru.
google.com IN CNAME rpz-passthru.
*.google.com IN CNAME rpz-passthru.
```

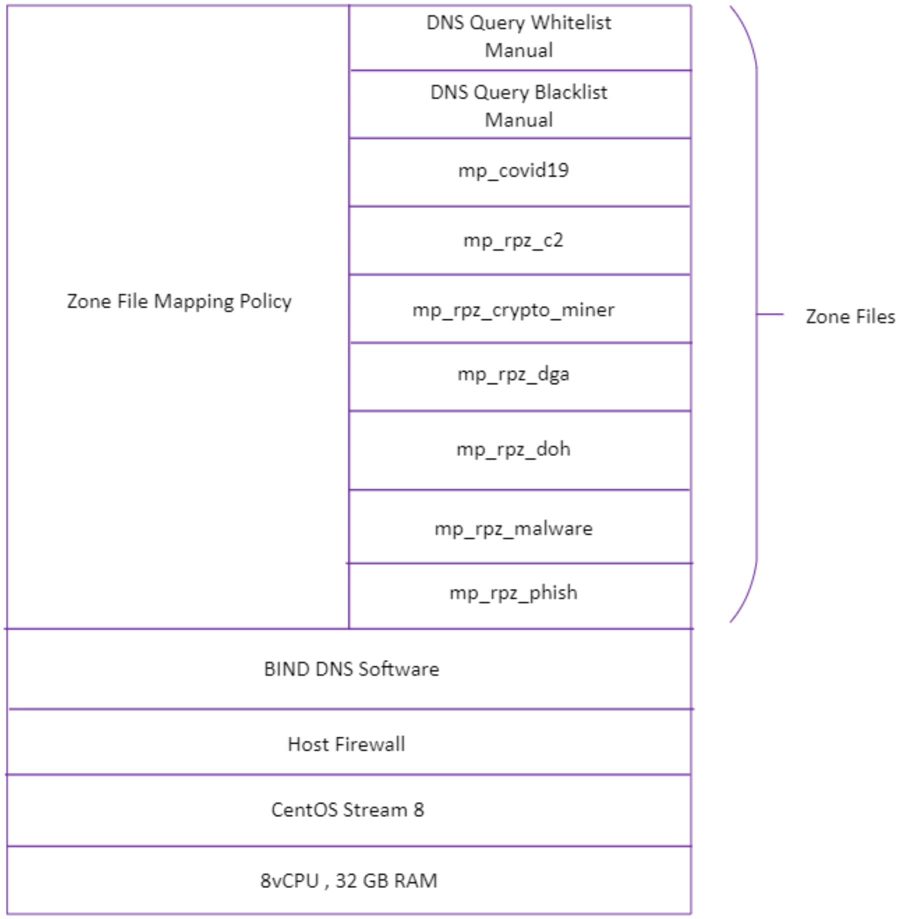


```
C:\Users\JeganSriMohanRam>nslookup google.com 13.234.186.42
Server:  UnKnown
Address:  13.234.186.42

Non-authoritative answer:
Name:    google.com
Addresses:  2404:6800:4009:827::200e
          142.250.192.14


C:\Users\JeganSriMohanRam>
```

Done!!!

To make DNS firewall more stronger, we can inherit third party Threat Intelligence services to write Zone files into the DNS firewall using AXFR technology (Master Slave Zone transfer technology)

Example check the below,



Zone File Mapping Policy

DNS Query Whitelist Manual

DNS Query Blacklist Manual

mp_covid19

mp_rpz_c2

mp_rpz_crypto_miner

mp_rpz_dga

mp_rpz_doh

mp_rpz_malware

mp_rpz_phish

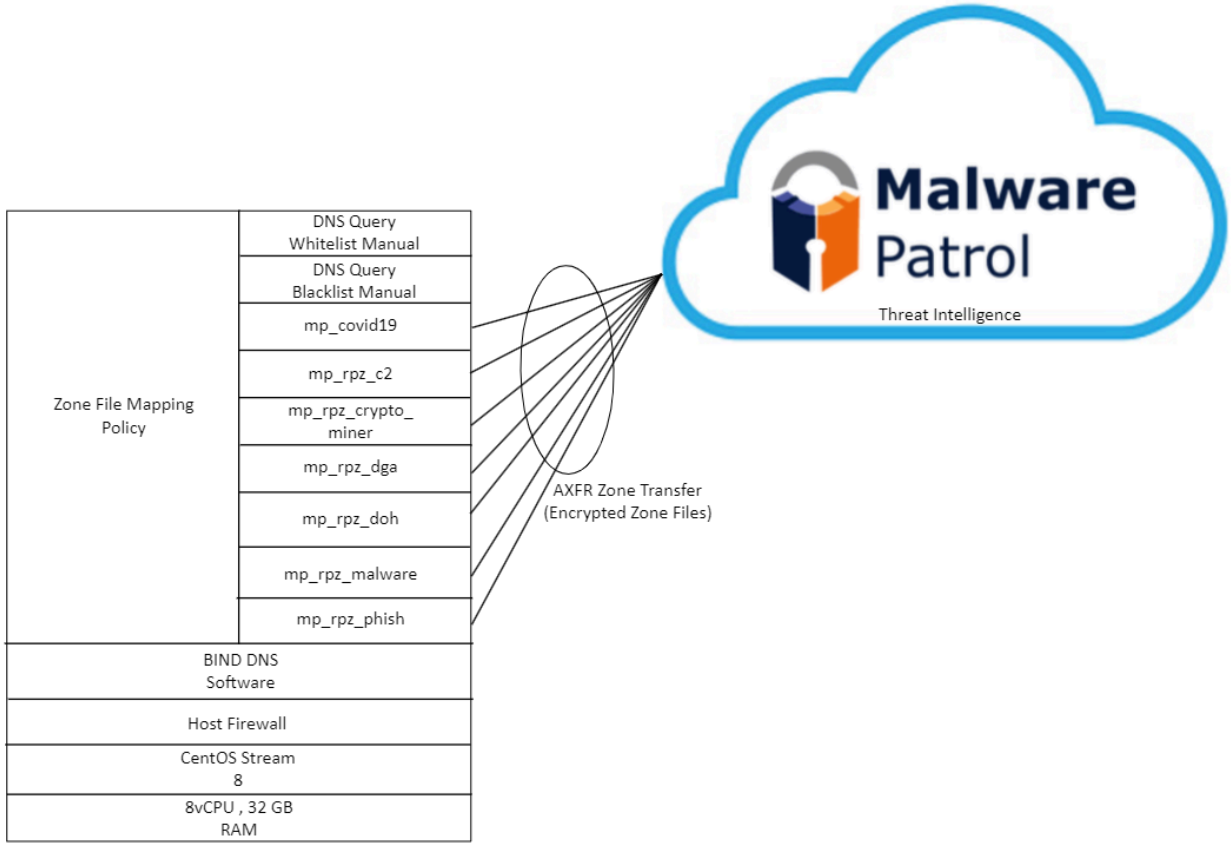
Zone Files

BIND DNS Software

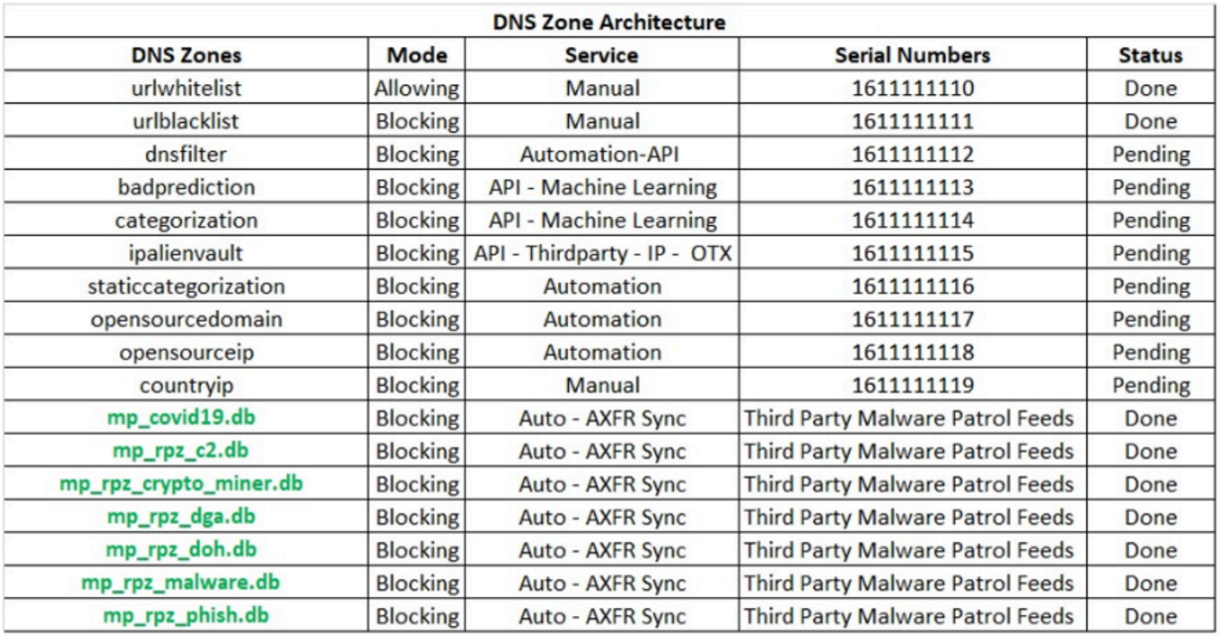
Host Firewall

CentOS Stream 8

8vCPU , 32 GB RAM



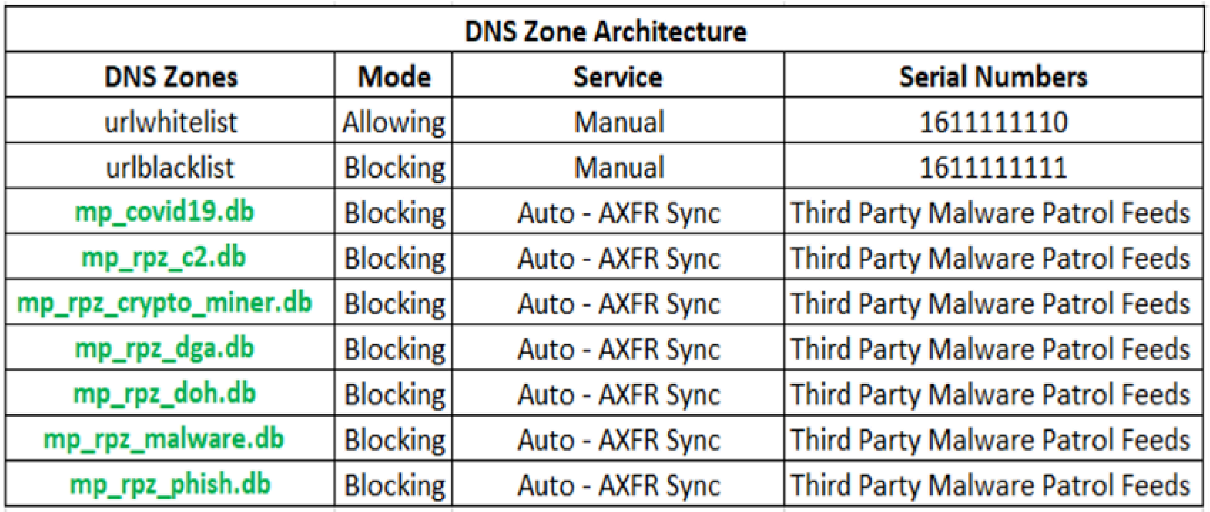
DNS Firewall Zones ST21 Design



DNS Zone Architecture

DNS Zones	Mode	Service	Serial Numbers	Status
urlwhitelist	Allowing	Manual	1611111110	Done
urlblacklist	Blocking	Manual	1611111111	Done
dnsfilter	Blocking	Automation-API	1611111112	Pending
badprediction	Blocking	API - Machine Learning	1611111113	Pending
categorization	Blocking	API - Machine Learning	1611111114	Pending
ipalienvault	Blocking	API - Thirdparty - IP - OTX	1611111115	Pending
staticcategorization	Blocking	Automation	1611111116	Pending
opensourcedomain	Blocking	Automation	1611111117	Pending
opensourceip	Blocking	Automation	1611111118	Pending
countryip	Blocking	Manual	1611111119	Pending
mp_covid19.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_c2.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_crypto_miner.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_dga.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_doh.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_malware.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done
mp_rpz_phish.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds	Done

DNS Firewall Zones Suggestion for Clients



DNS Zone Architecture			
DNS Zones	**Mode**	**Service**	**Serial Numbers**
urlwhitelist	Allowing	Manual	1611111110
urlblacklist	Blocking	Manual	1611111111
mp_covid19.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_c2.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_crypto_miner.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_dga.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_doh.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_malware.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds
mp_rpz_phish.db	Blocking	Auto - AXFR Sync	Third Party Malware Patrol Feeds

DNS Sinkhole

DNS Sinkhole is allowing very specific DNS queries and denying the rest by default.

So thereby achieving the easiest and best DNS Security.

In the same scenario, the /var/named/urlblacklist.db zone file needs to be modified/replaced with the below content to achieve DNS Sinkhole effect.

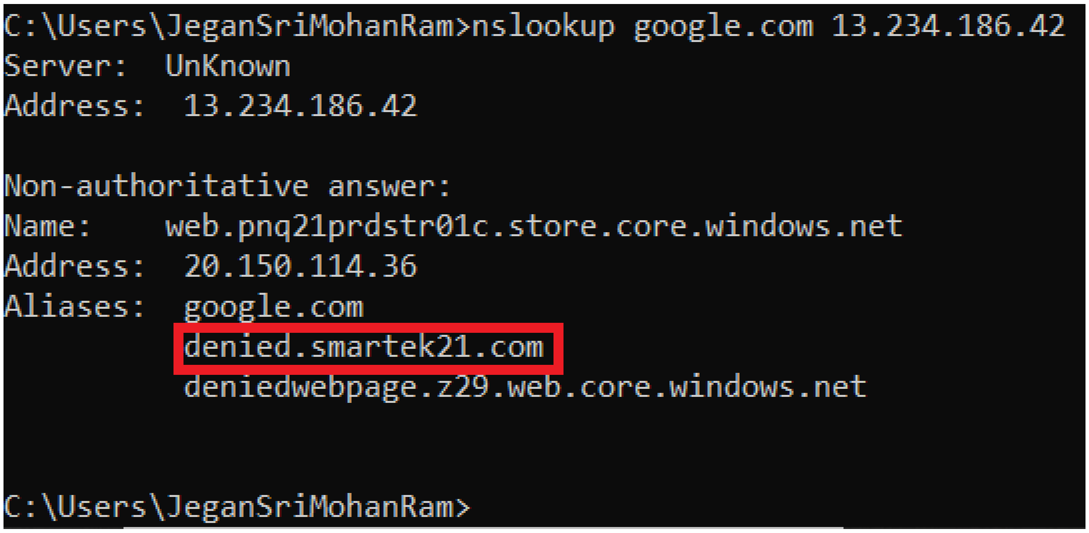
```
 $TTL 300

@ IN SOA  localhost. need.to.know.only. (
                 1623071203; Serial number
                  3600      ; refresh 1 hour
                  600       ; retry 10 minutes
                  86400     ; expiry 1 week
                  600 )     ; min ttl 10 minutes

@ IN NS   LOCALHOST.
*     IN  A   127.0.0.1
*     IN      AAAA    ::1
```

Now do the following ,

systemctl restart named
systemctl status named

Nothing will resolve or connect through this DNS Sinkhole IP 13.234.186.42

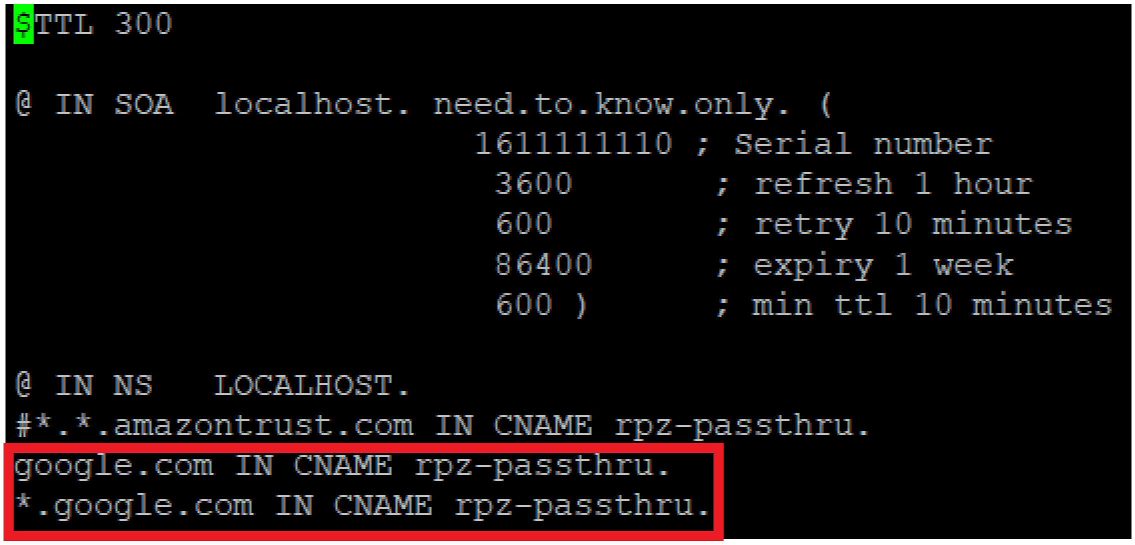
Lets test with google.com,



```
C:\Users\JeganSriMohanRam>nslookup google.com 13.234.186.42
Server:  UnKnown
Address:  13.234.186.42

Non-authoritative answer:
Name:    web.pnq21prdstr01c.store.core.windows.net
Address:  20.150.114.36
Aliases:  google.com
          denied.smartek21.com
          deniedwebpage.z29.web.core.windows.net


C:\Users\JeganSriMohanRam>
```

The only option to make the DNS Sinkhole allow google.com is to whitelist in /var/named/urlwhitelist.db file

urlwhitelist db zone file

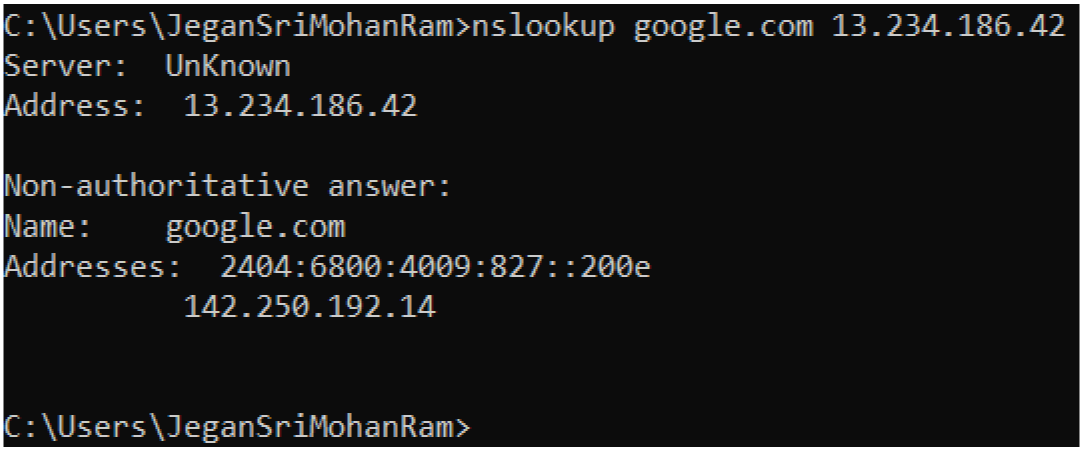


```
$TTL 300

@ IN SOA  localhost. need.to.know.only. (
                     1611111110 ; Serial number
                      3600        ; refresh 1 hour
                      600         ; retry 10 minutes
                      86400       ; expiry 1 week
                      600 )       ; min ttl 10 minutes

@ IN NS   LOCALHOST.
#*.*.amazontrust.com IN CNAME rpz-passthru.
google.com IN CNAME rpz-passthru.
*.google.com IN CNAME rpz-passthru.
```

Now do the following ,

systemctl restart named
systemctl status named

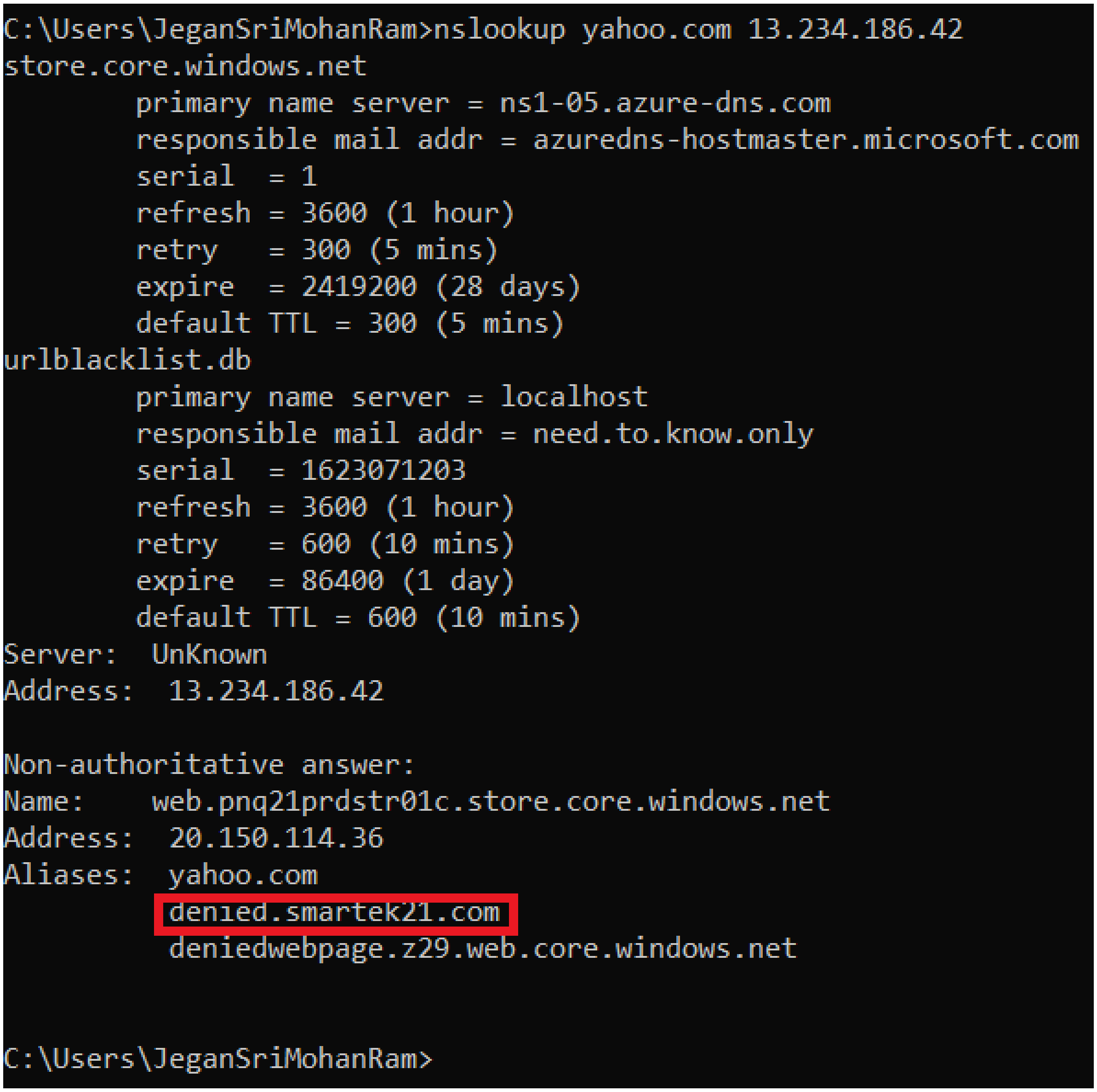
Lets test with google.com, it should work/passthrough this time,



```
C:\Users\JeganSriMohanRam>nslookup google.com 13.234.186.42
Server:  UnKnown
Address:  13.234.186.42

Non-authoritative answer:
Name:    google.com
Addresses:  2404:6800:4009:827::200e
          142.250.192.14


C:\Users\JeganSriMohanRam>
```

In this only google.com will work, else nothing will work.

Lets test with yahoo.com,



```
C:\Users\JeganSriMohanRam>nslookup yahoo.com 13.234.186.42
store.core.windows.net
        primary name server = ns1-05.azure-dns.com
        responsible mail addr = azuredns-hostmaster.microsoft.com
        serial  = 1
        refresh = 3600 (1 hour)
        retry   = 300 (5 mins)
        expire  = 2419200 (28 days)
        default TTL = 300 (5 mins)
urlblacklist.db
        primary name server = localhost
        responsible mail addr = need.to.know.only
        serial  = 1623071203
        refresh = 3600 (1 hour)
        retry   = 600 (10 mins)
        expire  = 86400 (1 day)
        default TTL = 600 (10 mins)
Server:  UnKnown
Address:  13.234.186.42

Non-authoritative answer:
Name:    web.pnq21prdstr01c.store.core.windows.net
Address:  20.150.114.36
Aliases:  yahoo.com
          denied.smartek21.com
          deniedwebpage.z29.web.core.windows.net


C:\Users\JeganSriMohanRam>
```

Hence proved.

Done!!!