Public Facing DNS Resolver for NITK

Chirag R - 201CS170
Akash Prasad - 201CS205
Attada Ramprasad - 201CS210

Introduction

- Name to Address Resolution
- Hierarchy of Name Servers
- Authoritative Name Servers
- Domain Name Servers
- Working of a DNS Server

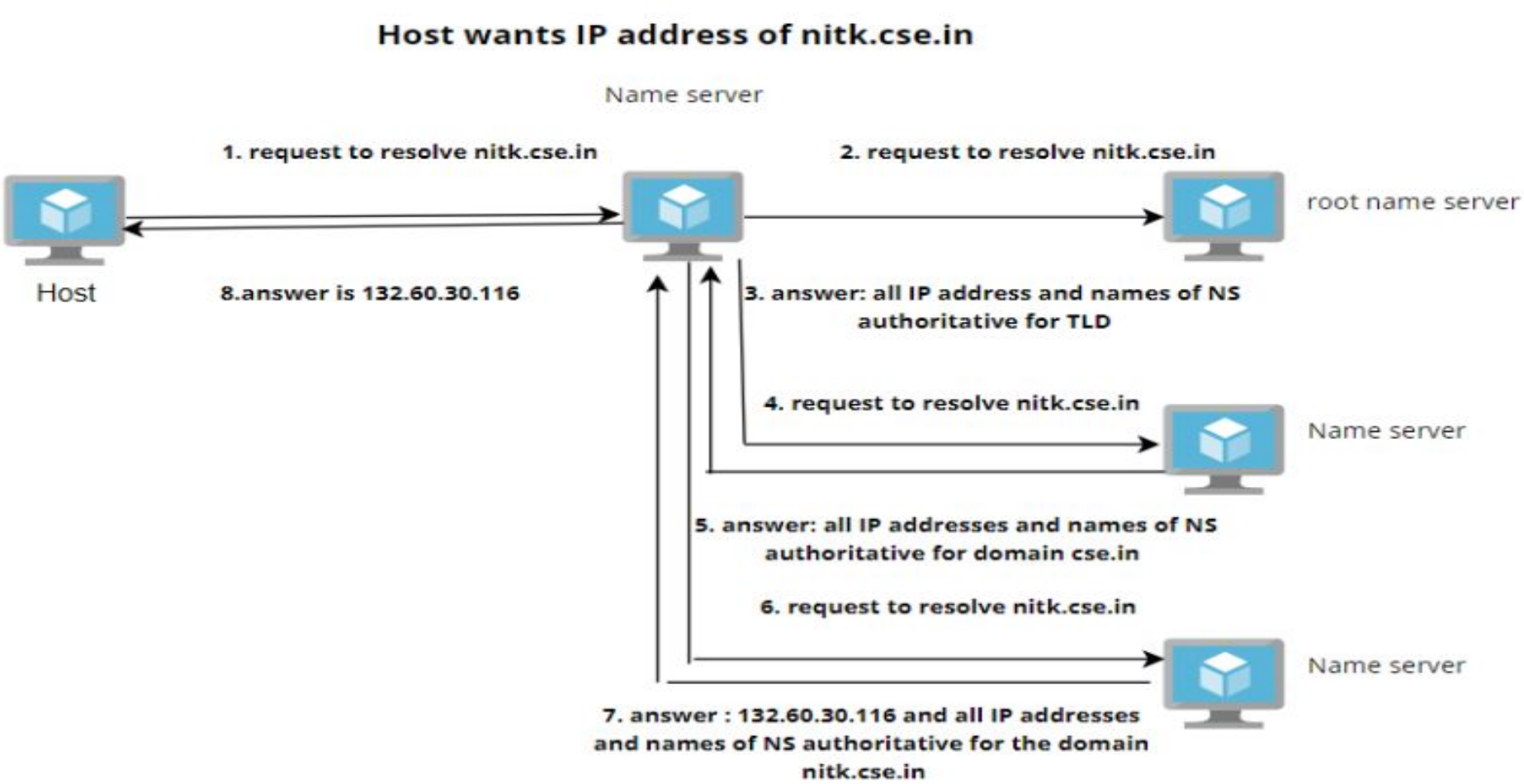


Figure 1.2: Domain Name Server

Problem Statement

Task - Develop a public facing DNS resolver for NITK
Faster DNS Resolution: By caching DNS records locally, the DNS server can reduce the time it takes to resolve domain names to IP addresses for frequently accessed websites.
Can customize according to the needs of the institute at least in the initial phases
Great learning experience for all those involved to understand DNS, networking and security

Note -
Security - A DNS resolver is vulnerable to many kinds of attacks.
Consider legal aspects such as data privacy laws, copyright issues, terms of service provided by public DNS resolvers etc.
Compliance - The working of DNS should be understood in depth to know relevant standards and best practices. Communicate with CCC to know how the same can be implemented at our institute.

Experimental Setup

Virtual machines to be set up on remote system through SSH

Ubuntu servers unreachable

- deb and deb-src => latest mirror files in source.list file
- wget -v --post-data "mode=191&username=201272&password=Rsy&a=1707411321129&producttype=0" https://nac.nitk.ac.in:8090/login.xml

Vagrantfile script to create VMs with required specifications

Name	Location in virtual box	Used as	Fully qualified Domain name	Private IP address
Pdns	Machine1	Primary dns server	machine1.nitk.example.com	192.168.56.2
sdns	Machine2	Secondary dns server	machine2.nitk.example.com	192.168.56.3
host1(Client)	Machine3(used as client)	Generic host1	machine3.nitk.example.com	192.168.56.10
host2(Client)	Machine4(used as client)	Generic host 2	machine4.nitk.example.com	192.168.56.11

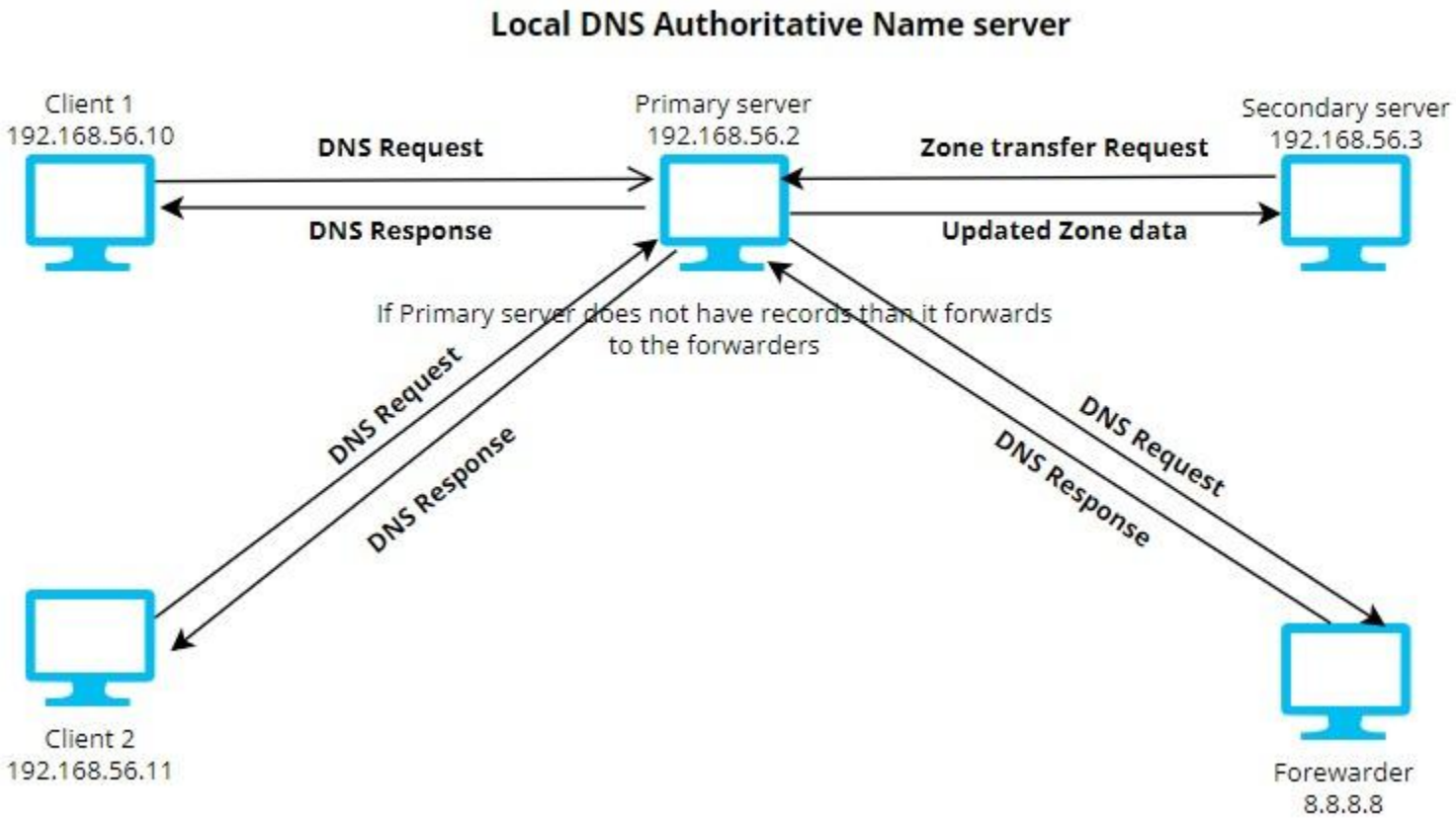
```
 GNU nano 6.2                                  /etc/bind/named.conf.options
acl "trusted" {

        192.168.56.2;   # machine1(primary server)
        192.168.56.3;   # machine2(secondary server)
        192.168.56.10;  # machine3(host1 client)
        192.168.56.11;  # machine4(host2 client)


};
```

```
# enables recursive queries
recursion yes;
allow-recursion { trusted; };   # allows recursive queries from "trusted" clients
allow-query { trusted; };       #for caching of the dns server
listen-on { 192.168.56.2; };    # machine1 private IP address- listen o private network only
allow-transfer { none; };       # disable zone transfers bydefault


 forwarders {
        8.8.8.8;                # Google public DNS Server for query resolution
        8.8.4.4;
 };
```


Local DNS Authoritative Name server
Client 1
192.168.56.10
DNS Request
DNS Response
Primary server
192.168.56.2
Zone transfer Request
Updated Zone data
Secondary server
192.168.56.3
If Primary server does not have records than it forwards
to the forwarders
DNS Request
DNS Response
DNS Request
DNS Response
Client 2
192.168.56.11
Forewarder
8.8.8.8

```
 GNU nano 6.2                                  /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 "nitk.example.com" {
        type primary;
        file "/etc/bind/zones/db.nitk.example.com";     # zone file path
        allow-transfer { 192.168.56.3; };               # machine2 private IP address -secondary server machine2 IP address
};

zone "56.168.192.in-addr.arpa" {
        type primary;
        file "/etc/bind/zones/db.192.168.56";           # 192.168.56.0 subnet
        allow-transfer { 192.168.56.3; };               #secondary server machine2 private IP address
};
```

```
 GNU nano 6.2                                    /etc/bind/zones/db.nitk.example.com
$TTL    604800
@       IN      SOA     machine1.nitk.example.com. admin.nitk.example.com. (
                              7         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
; name servers - NS records
     IN      NS          machine1.nitk.example.com.
     IN      NS          machine2.nitk.example.com.

; name servers - A records
machine1.nitk.example.com.                      IN      A       192.168.56.2
machine2.nitk.example.com.                      IN      A       192.168.56.3

; 192.168.56.0/21 - A records
machine3.nitk.example.com.      IN      A       192.168.56.10
machine4.nitk.example.com.      IN      A       192.168.56.11
```

```
  GNU nano 6.2                                    /etc/bind/zones/db.192.168.56
$TTL    604800
@       IN      SOA     nitk.example.com. admin.nitk.example.com. (
                              6         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
; name servers
        IN      NS      machine1.nitk.example.com.
        IN      NS      machine2.nitk.example.com.

; PTR Records
2       IN      PTR     machine1.nitk.example.com.      ;192.168.56.2
3       IN      PTR     machine2.nitk.example.com.      ;192.168.56.3
10      IN      PTR     machine3.nitk.example.com.      ;192.168.56.10
11      IN      PTR     machine4.nitk.example.com.      ;192.168.56.11
```

```
 GNU nano 6.2                                  /etc/netplan/00-private-nameservers.yaml
network:
    version: 2
    ethernets:
        eth1:                                       # Private network interface
            nameservers:
                addresses:
                - 192.168.56.2                    # Private IP for ns1
                - 192.168.56.3                    # Private IP for ns2
                search: [ nitk.example.com ]        # DNS zone
```

```
vagrant@machine4:~$ sudo resolvectl
Global
       Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (eth0)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 10.0.2.3
       DNS Servers: 10.0.2.3
        DNS Domain: nitk.ac.in

Link 3 (eth1)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 192.168.56.2
       DNS Servers: 192.168.56.2 192.168.56.3
        DNS Domain: nitk.example.com
```

Attempts for a Recursive Resolver

Resolve names other than those defined in the zone files (unauthoritative responses)

Forwarders used when DNS records are not available

Not having forwarders lead to resolvers perform recursive resolution querying root servers to find authoritative name servers for the given domain

Tried commenting out the forwarders

Worked on a physical system with similar configurations

Results and Analysis

```
vagrant@machine4:~$ sudo resolvectl
Global
       Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (eth0)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 10.0.2.3
       DNS Servers: 10.0.2.3
        DNS Domain: nitk.ac.in

Link 3 (eth1)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 192.168.56.2
       DNS Servers: 192.168.56.2 192.168.56.3
        DNS Domain: nitk.example.com
```

Shows that eth1 interface is configured with our DNS servers

```
vagrant@machine4:~$ nslookup machine3
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   machine3.nitk.example.com
Address: 192.168.56.10
```

```
vagrant@machine4:~$ nslookup machine3 192.168.56.2
Server:         192.168.56.2
Address:        192.168.56.2#53

Name:   machine3.nitk.example.com
Address: 192.168.56.10
```

```
vagrant@machine4:~$ nslookup machine3 192.168.56.3
Server:         192.168.56.3
Address:        192.168.56.3#53

Name:   machine3.nitk.example.com
Address: 192.168.56.10
```

Shows queries are being resolved as expected

```
vagrant@machine4:~$ nslookup google.com 192.168.56.2
Server:         192.168.56.2
Address:        192.168.56.2#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.77.174
Name:   google.com
Address: 2404:6800:4007:818::200e
```

```
vagrant@machine4:~$ nslookup google.com 192.168.56.2
Server:         192.168.56.2
Address:        192.168.56.2#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.193.110
Name:   google.com
Address: 2404:6800:4007:826::200e
```

Results before and after using commenting out forwarders

```
vagrant@machine4:~$ nslookup google.com 192.168.56.2
Server:         192.168.56.2
Address:        192.168.56.2#53

** server can't find google.com: REFUSED
```

Result after commenting out machine4 IP address from ACL “trusted”

Conclusion and Future Work

- Successfully created authoritative name server that gives responses to queries corresponding to records in the zone files
- Since removing forwarders and client server from the ACL of "trusted" servers being used for allow-recursion does not resolve domain names, we can conclude that the recursive resolver works fine when the client is part of ACL "trusted".
- Future work may include proof of working of recursive resolver,, caching and security aspects and working with CCC to allocate an IP address for the resolver making it public

Thank You