Arquitetura de Redes Avançadas

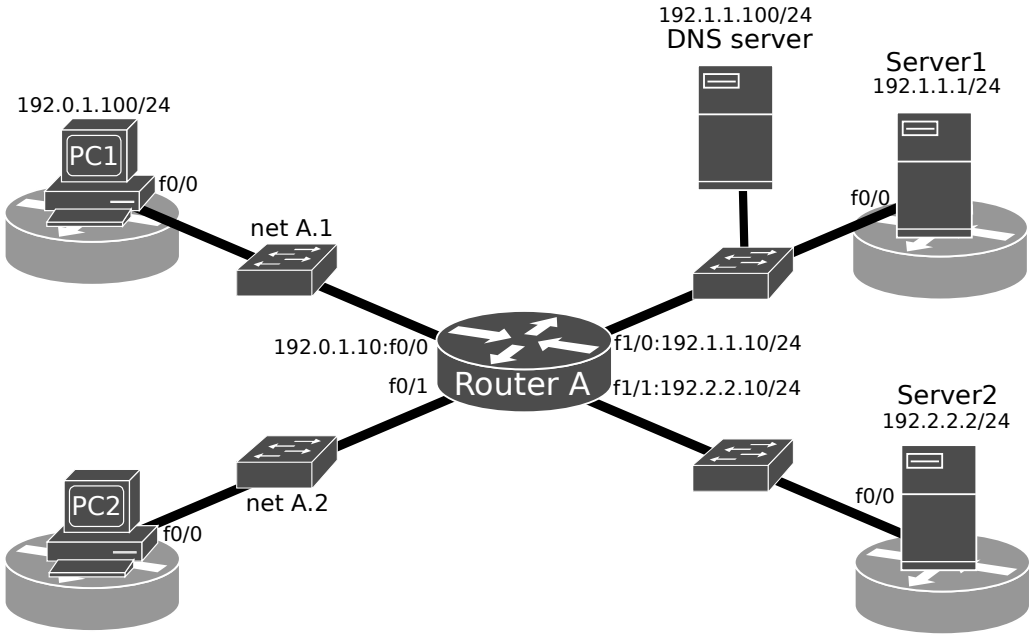
Laboratory Guide

CDN Deployment with Conditional DNS

Objectives

- Understand the basic mechanisms for CDN deployment.
- Conditional DNS.

Geographically aware DNS Server for CDN Deployment



1. Set up and configure the above depicted network. PC1, PC2 (with no IP configuration for now), Server1, and Server2 may be emulated **with routers with disable routing capabilities**. Example for PC1 configuration:

```
PC1(config)#no ip routing                      !disables IPv4 routing
PC1(config)#ip default-gateway 192.0.1.100     !defines default gateway
PC1(config)#ip name-server 192.1.1.100         !defines DNS server
PC1(config)#ip domain lookup                   !domain names should be resolved using DNS
PC1(config)#int f0/0
PC1(config-if)#ip address 192.0.1.100 255.255.255.0
PC1(config-if)#no shutdown
```

Make similar configurations for Server1 and Server2.

The DNS server must be implemented as a virtual Ubuntu/Debian server. A VirtualBox appliance (with all required services and files) is available to download here (login/password: ubuntu/reverse). Connect VM to a host-only adapter (e.g., vboxnet0 in Linux), if none available create a new host-only network (File → Preferences → Network). On GNS3 use a Cloud with the chosen host-only adapter.

Configure DNS Server IPv4 address and gateway:

```
# sudo su
# ifconfig eth0 192.1.1.100/24
# route add default gw 192.1.1.10
```

Note: Any other server should have installed a DNS server (bind9) with default configuration.

Note 2: With your own server image, download the file GeoIP.acl (from the UC page at elearning.ua.pt) and place it on the /etc/bind/ folder of your server.

Test full connectivity, and analyze the contents of the file GeoIP.acl. The file contains a set of BIND Access Control Lists that map each IPv4 network of the world with a country prefix. It was constructed based on the GeoIP tools and database from MaxMind.

2. On the DNS server, load the ACL file to BIND configuration by adding the following line to /etc/bind/named.conf (before the other include directives):

```
include /etc/bind/GeoIP.acl;
```

If present, comment the line

```
//include /etc/bind/named.conf.default-zones;
```

Restart your DNS server:

```
service bind9 restart
```

Note: If the service restart fails, check the syslog file for reason:

```
tail /var/log/syslog
```

3. Assuming that you own the domain **ARACDN.com** configure your DNS server to act as a master server (zone) for that domain. Start by creating the definition of the zones conditioned by the views (which are dependent of the client geographic position obtain from the ACL) with the associated *statements* (zone specific parameters), edit the file /etc/bind/named.conf.local (with root privileges) and add the following definitions:

```
view "europe" {
  match-clients { PT; ES; FR; GB; };
  recursion no;
  zone "aracdn.com" {
    type master;
    file "/etc/bind/aracdn.com-europe.db";
  };
};

view "north_america" {
  match-clients { US; CA; MX; };
  recursion no;
  zone "aracdn.com" {
    type master;
    file "/etc/bind/aracdn.com-north_america.db";
  };
};

view "other" {
  match-clients { any; };
  recursion no;
  zone "aracdn.com" {
    type master;
    file "/etc/bind/aracdn.com-other.db";
  };
};
```

4. Create the files /etc/bind/aracdn.com-*.db (with root privileges) and add distinct contents.
Example for aracdn.com-europe.db:

```
$TTL   604800
$ORIGIN aracdn.com.
@      IN     SOA    ns1.arxcdn.com. adm.arxcdn.com. (
                          2             ; Serial
                     604800             ; Refresh
                      86400             ; Retry
                    2419200             ; Expire
                     604800 )     ; Negative Cache TTL
       IN     NS     ns1.aracdn.com.
       IN     A      192.1.1.1
ns1    IN     A      192.1.1.100
```

Example for aracdn.com-north_america.db:

```
$TTL   604800
$ORIGIN aracdn.com.
@      IN     SOA    ns1.aracdn.com. adm.aracdn.com. (
                          2             ; Serial
                     604800             ; Refresh
                      86400             ; Retry
                    2419200             ; Expire
                     604800 )     ; Negative Cache TTL
       IN     NS     ns1.aracdn.com.
       IN     A      192.2.2.2
ns1    IN     A      192.1.1.100
```

Example for aracdn.com-other.db:

```
$TTL   604800
$ORIGIN aracdn.com.
@      IN     SOA    ns1.aracdn.com. adm.aracdn.com. (
                          2             ; Serial
                     604800             ; Refresh
                      86400             ; Retry
                    2419200             ; Expire
                     604800 )     ; Negative Cache TTL
       IN     NS     ns1.aracdn.com.
       IN     A      192.3.3.3
ns1    IN     A      192.1.1.100
```

Verify if your zone files are correctly defined:

```
named-checkzone aracdn.com aracdn.com-europe.db
named-checkzone aracdn.com aracdn.com-north_america.db
named-checkzone aracdn.com aracdn.com-other.db
```

Restart your DNS server:

```
service bind9 restart
```

Start a packet captures on RouterA's f0/0 interface.

Using PC1, test the configuration of your DNS by forcing a DNS query with the following ping command:

```
ping aracdn.com
```

Analyze the the content of the DNS packets (server response) and PC1sho 's DNS cache (`show hosts`).

5. Configure the PC2 and the router as belonging to different IPv4 networks in different world locations, e.g:

US: 12.111.196.0/24, 64.20.253.0/24;
PT: 176.124.252.0/24; 192.112.45.0/24;
BR: 65.205.133.0/24; 192.207.204.0/23;

After each address change, clear PC2's DNS cache (`clear host *`) to force new DNS queries.

Start a packet captures on RouterA's f0/1 interface.

Using PC2, test the configuration of your DNS by forcing a DNS query with the following ping command:

```
ping aracdn.com
```

Analyze the the content of the DNS packets (server response) and PC2's DNS cache (`show hosts`) and correlate them with PC2 network's "world location".

Use the GeoIP.acl file to identify more networks in different countries.