Linux DNS

Setting up Coredns and writing a zone file

Type of Systems: Linux
Necessary Skills: Linux admin, (basic) Docker
Complete Goal: Create a simple DNS server.

Writing the zone file

A common format for the name of zone files is “example.com.db” where example.com is your domain name. Create the directory /opt/Coredns and create the file “example.com.db” Begin editing it in your preferred CLI text editor.

The most basic form of a zone file is as follows:

```
$ORIGIN example.com.
$TTL 3600
example.com. IN SOA ns.example.com. admin.example.com. (
                                                  1       ; serial
                                                  3600    ; refresh
                                                  600     ; retry
                                                  1209600 ; expire
                                                  3600 )  ; negative caching TTL
example.com. www IN CNAME example.com.
example.com. IN NS ns
ns IN A 10.55.0.2
example.com. MX 10 mail
mail IN A 10.55.0.3
```

The first line, $ORIGIN, denotes the domain which will be used for the current section. Note that whenever this domain or other addresses are referenced in the zone file, they end in a dot. The next line, $TTL, essentially tells clients how long to cache the DNS information for.

The third item is the most important part of a (functioning) zone file and is required before any rules surrounding the previously referenced domain. The format of this line is as follows:

> <domain name> IN SOA ns.<domain name> <hostmaster>.<domain name> (a bunch of numbers)

IN refers to the internet record class, and can usually be omitted (but don’t omit it). SOA stands for “start of authority.” The hostmaster name is essentially just an email address. If you have an actual email address available for an admin, use that. Otherwise, just use a filler like admin. The numbers in the parentheses afterwards are parameters surrounding the timing of various functions of a DNS.

Each line after this one will either start with the domain name, or the name of some service. The format for the last two lines beginning with the domain name is:

<domain name> <record class> <parameter> <service name>

For the ns service, it is in the IN record class, and takes parameter NS, and is named ns. For the mail service, it is in the MX record class, and the parameter is the priority (relevant for systems with multiple mail services), with name mail.

Following each of these lines is a line which routes these services to a specific host. For IPv4, the parameter following in is A, and for IPv6 it is AAAA. The first part of the line is the name given previously for the service, and the last is the IP address of the host.

Spin up the DNS server:

(This section is taken from “Wicked Quick Backup DNS”)
To bring up the dns server with the following bash on the Docker server, using the zone file you made, and the Corefile below with the zone section repeated for each of your zone files:

```
docker run --restart=always --volume=<Config directory>:/root/ -p 53:53/udp coredns/coredns -conf /root/Corefile
```

Note that if you have issues running this command because it says port 53 is already in use, you should bind that port mapping to a specific address on the host by using `-p ADDRESS:53:53/udp`
Also note that <Config directory> must be a fully specified directory to your Coredns configuration directory, preferably at /opt/Coredns/
Corefile:

```
.:53 {
      forward . 1.1.1.1
      log
      errors
}
example.com:53 {
      file /root/<Zonefile name>
      log
      errors
}
```