Module 1 – Authoritative DNS

This module contains two sections that must be completed in order. In section 1 we will discuss configuring Authoritative Name Servers, DNS Replication, and we'll finish with a review of the top 10 errors that can occur when configuring your Authoritative DNS infrastructure. In section two, you will apply the principles discussed and introduced in section 1 in a live laboratory environment. When you have completed your lab assignments, be sure to submit them to your instructor for evaluation. If your instructor indicates that you have successfully completed your assignment, you can then move on to the next module in this course.

Stub Resolver and the DNS

Before we jump into our content for this lesson, let's briefly recap some of the Introduction to Network Operation course principles. That course provided an overview of how the stub resolver works and how to set up a caching DNS server.

Remember, the caching DNS server, in summary, allows you to cache Resource Records obtained from DNS so you can have a faster resolution of domain names. The caching server returns the answer to the requested information if already known. Otherwise, the server searches for the correct authoritative server with the information and then caches the result for future queries.

Authoritative DNS

An Authoritative name server contains the actual information published in the DNS by the domain owner. It holds the original source files of a domain's zone file which contains the domains information and its resource records. Remember, a zone is part of a domain for which administrative responsibility has been delegated to a single manager.

An Authoritative name server contains the actual information published in the DNS by the domain owner. This information is provided in the form of Resource Records. For example, an "A" Record or Address Record is a Resource Record that provides the IPv4 address for a hostname.

Authoritative name servers are responsible for providing answers to recursive/caching name servers of which Resource Records it has stored. These answers contain important information for each domain, like IP addresses and the appropriate response to provide to caching name servers that request information. Once the answers are provided, this allows web browsers to locate Internet services like websites.

An authoritative DNS server performs two important tasks:

- First, it stores Resource Records for a zone
- Second, it responds to requests from a recursive/caching DNS server about the correct IP address assigned to a domain name.

Authoritative name servers also hold the original source files of a domain's zone files which contain the domain's information and its Resource Records... and remember, a zone is part of a domain for which administrative responsibility has been delegated to a single manager. When configuring software applications, you will be using the term “zone” when adding and updating domain resource records.

DNS Replication

Let's now take a look at DNS replication and the importance of zone reliability. DNS replication creates the same instance of a DNS server either in a different geographic location or network. The geographic placement and the diversity of network connectivity exhibited by the set of DNS servers for a zone can increase the reliability of that zone and improve overall network performance and access characteristics.

So, when replication is done and configured, the mirrored server will have the same IP4 and IP6 addresses and data, so that any authorized name server will be able to provide an answer for the domain. For example, all DNS root servers mirrors have the same IPv4 and IPv6 addresses.

DNS Replication Process

So how does replication work?

A domain or zone will have a Primary DNS Server and at least one Secondary DNS server. The replication process requires the Primary DNS server to be configured to allow the Secondary server to download a copy of the records stored by the Primary Server.

Once the connection is made, the Secondary servers periodically poll from the Primary server to check for new data. This is referred to as the "Refresh Interval". A consistent practice of checking the refresh interval is important because notification of changes which occurs when the Primary DNS server notifies the Secondary DNS server of an update to the resource records for a domain, can be unreliable, especially in instances when a network experiences packet loss since UDP could be used to send the “DNS Notify” packets.

In the replication process, a Secondary DNS server obtains an entire copy or part of a zone from a Primary DNS server. This is known as a zone transfer process and is also known as a “AXFR” type DNS query. A zone transfer uses the Transmission Control Protocol (TCP) and takes the form of a client-server transaction. . Normally, the Primary server does not push data to the Secondary servers, and it is the Secondary servers that request for data from the Primary server.

Originally, this process was the only mechanism for replication. However, with modern software, replication can also occur when the Primary server notifies the Secondary server when data changes, which results in quicker updates.

DNS Zones

A DNS zone can be divided into several zones. A zone is a section of the DNS namespace that is managed by an administrator. Every zone file has a serial number. The recommended format for every serial number is: the 4-digit year, the 2-digit month, 01 to 12, the 2-digit day, from 01 to 31, and a 2-digit number, from 00-99, representing the number of changes today. For example, if you

change the file on 23rd of August 2021, the serial number will be 2021082300. If you change it again on the same day, it will be 2021082301. The Secondary server uses a periodic UDP query to check the serial number and it will copy transfer zone data, using TCP, when this number increases.

It is the DNS admin's responsibility to increase the serial number after every change, otherwise, the secondary and primary servers will be inconsistent

Decreasing a serial number

Let's say you have an instance where you increased the serial number too high and you want to reset it to a lower number. RFC 1912 section 3.1 provides a detailed explanation to address this problem.

- Take the `incorrect' serial number and add 2147483647 to it. If the number exceeds 4294967296, subtract 4294967296.
- Load the resulting number. Then wait 2 refresh periods to allow the zone to propagate to all servers. Repeat step 1 until the resulting serial number is less than the target serial number. Increase the serial number to the target serial number.

Keep in mind, if you ever decrease the serial number, the secondary DNS will never update again until the serial number goes above its previous value.

DNS Configuration File

Now let's take a look at how the BIND DNS software can be configured to serve a zone "example.com"f . The configuration shown can be included in a separate file or within the "named.conf" file. It is common practice to have a separate file containing details of your zones to avoid having a large "named.conf" file with several entries.

Also note that each zone should have a separate configuration file containing entries pertaining to a single zone. So you would have:

- named.conf - which is the main BIND Configuration file setting out which IP addresses the BIND DNS server will listen to requests to. This file would contain entries and paths of other configuration files that BIND should load and it is here where an entry to load "zones.conf" should be included.

- zones.conf - which includes a configuration of all the zones that your BIND server will be primary or secondary for. Each zone entry must contain the path to the zone file. You may call this file “zones.conf” or any other name but it must match the info you include in your “named.conf”
- example.com - which is a zone file - a normal text file - that will contain resource record data for the zone “example.com”

Within the zone configuration - lets call ours “zones.conf”, you will have a configuration section for each zone your server will be authoritative for and declare whether your server will be a primary DNS server for the zone or a secondary DNS server for the zone.
In this example, the zone configuration file declares a zone “example.com” of which this server will be the primary DNS server (master), and which servers will be the secondary servers and allowed to transfer - or make copies - of the entire zone. Note that in the “zones.conf” file, no resource record data is included - just details of which zones are to be served and which servers are allowed to make zone transfers.

Server Configurations

Now that we have the primary and secondary servers configured, there may be instances where one server is the primary for some zones and secondary for other zones. As such, we recommend keeping the files in different directories. Also, the secondary directory needs to have appropriate permissions so that the DNS software daemon, which is a software process that runs in the background, can create the necessary files.

So, what happens when you have a remote machine requesting a transfer of zone contents?

Server Configurations

Now that we have the primary and secondary servers configured, there may be instances where one server is the primary for some zones and secondary for other zones. As such, we recommend keeping the files in different directories. Also, the secondary directory needs to have appropriate permissions so that the DNS software daemon, which is a software process that runs in the background, can create the necessary files.

So, what happens when you have a remote machine requesting a transfer of zone contents?

Allow Transfer

When there is a transfer request from a remote machine, this type of request is fulfilled using the "allow-transfer" function. This function defines the IP addresses that are allowed to transfer or copy the zone information from the primary or secondary for the zone. You should only allow your designated secondary servers to be able to obtain a copy of the zone file from the primary DNS server.

Zone File Structure

So how is a zone file actually structured?

Zone files are simple text files that contain the details of all resource records for that domain. Take a look at this sample zone file for the example.com domain. Click each section of the zone file to learn more about it.

TTL stands for Time To Live, which mentions the time in seconds for which caching name servers can cache the data. The TTL value in this example is 1 day. The TTL is a global option that sets the default TTL for all other records.

The $ORIGIN indicates a DNS node tree and will typically start a DNS zone file. Any host labels below the origin will append the origin hostname to assemble a fully qualified domain name (FQDN). Any host label within a record that uses a fully qualified domain terminating with a dot (.) will not append the origin hostname.

The SOA or Start of authority is the mandatory record that must be present in all zone files. It specifies the main properties and characteristics of a domain. The characteristics include:

- The Host name of the primary server and the email address of the domain administrator. In this example the hostname is ns1.example.com; the admin's email is admin.example.com. The @ symbol is changd to a dot and a training dot. This will become admin@example.com.

- The serial number of the zone, as used by the secondary name servers to know if there is an update to the zone.
- The Refresh interval indicates how often the secondary DNS server checks the serial number on the Primary server.
- The Retry interval indicates how often the secondary DNS server checks serial number if the Primary server did not respond.
- The Expiry time is triggered if the Secondary server is unable to contact the primary for this period of time, the secondary DNS server will expire (delete) its copy of the zone data.
- The Non-existent RR period is used for negative caching: indicates how long a cache may store the non-existence of a Resource Record

The NS record specifies the Name Servers that are responsible for this domain. It is a list of all the nameservers for the zone, primary and secondary. In this example, we have two name servers (DNS servers) responsible for the example.com domain. One is ns1.example.com and the other is ns2.example.com.

The IP addresses for both your nameservers need to be defined in the zone file with either or both A and AAAA records. These are sometimes referred to as “glue records” and are critical in enabling the resolution of Resource Records in your zone. The "A" record or address record is used to match an IP address for a hostname in a zone file. It is usually the most frequently used resource record in a zone file. When you query for a domain, the default answer you get is an "A" record which is denoted by a capital A.

AAAA resource record , also known or referred to as the "Quad A" record, is used for an IPv6 record. This is because the IPv6 record has 4 times the address space of an IPv4 record.

The MX record or Mail eXchange record defines the mail servers for a domain.

CNAME stands for Canonical Name Record. Canonical names mask one domain name or alias to another. For example, internetsociety.org domain can have the CNAME of isoc.org.

Resource Record Format

Each resource record will be formatted to include the: Domain, TTL or Time To Live, Class, Type, and Data. In most instances there will be one record per line, except the SOA which can extend over several lines.

If the domain name does not end in a dot, the zone's own name ("origin") is appended. A domain name designated as the @ symbol, means the origin itself.

A couple of things to note: TTL shortcuts can be used such as 60s, 30m, 4h, 1w2d. However, if you omit the TTL, the default value will be used. If you omit the Class, it will default to IN.

Lastly, the type and data elements cannot be omitted or left empty.

Verifying Your Configuration

When using BIND, to validate your zone file configuration modification, there are several commands you can use. Click each of the below to learn more.

Use the "named-checkzone" command. This named-checkzone command checks the syntax and integrity of a zone file. It performs the same checks as the named command does when loading a zone.

Use the “named-checkconf” command to identify any errors in the named.conf file. These errors may result in the zone not being served by your primary name server which would lead to a number of resulting errors.

If you have an error in the named.conf file or the zone file, the named service may continue to run, but may serve old data that was loaded before the error was made in the configuration file.

Once all errors are addressed, you can use the command "rndc reload" to load the updates made to a zone file. This command reloads both the configuration file and zones.

If it is the first time you are loading the zone file, use the command "rndc reconfig". This command will load any new zones that you've added and remove any that you no longer have defined, but it won't detect any changes that you've made to zones that are already loaded.

Verifying Your Configuration

So now we have working authoritative name servers. However, the information in the zone files will not be served properly if the preceding zone level does not have appropriate delegation informing it to direct requests for the sub-domain to your authoritative DNS servers.

For example, for "example.com" to work correctly, the authoritative name servers for ".com" must have an entry to delegate control of "example.com" to "ns1.example.com" and "ns2.example.com". Also, the DNS root zone - denoted with a dot - must have an entry to delegate control for ".com" to the nameservers responsible for ".com" thus completing the full delegation process.

Similarly, if you would like to delegate control of "accounting.example.com" to another administrative entity, the authoritative name servers for "example.com" must contain information of this delegation.

Conclusion

This concludes section 1 of this module. In this section we discussed configuring Authoritative Name Servers, DNS Replication, and we finished with a review of the top 10 errors that can occur when configuring your DNS.

Now, it's your turn to put what we have discussed into practice by completing the lab assignments. Click the proceed to lab assignment button to continue.