

# CS 498 Lecture 2

## Setting up DNS and HTTP



Jennifer Hou  
Department of Computer Science  
University of Illinois at Urbana-Champaign

# DNS

## Reading:

1. Red Hat Linux 9: Red hat Linux Reference Guide, Chapter 12: Berkeley Internet Name Domain (BIND).
2. DNS HOWTO  
<http://www.linux.org/docs/ldp/howto/DNS-HOWTO.html>

# Overview

In /etc/named.conf

```
zone "example.com" IN {  
    type master;  
    file "example.com.zone";  
    allow-update { none; };  
};
```

Under /var/named/ directory

example.com.zone

```
$ORIGIN example.com  
$TTL 86400  
@ IN SOA dns1.example.com.  
hostmaster.example.com. (  
    2001062501 ; serial  
    21600    ; refresh after 6 hours  
    3600    ; retry after 1 hour  
    604800  ; expire after 1 week  
    86400 ) ; minimum TTL of 1 day  
  
IN NS dns1.example.com.  
IN MX 10 mail.example.com.  
IN MX 20 mail2.example.com.
```

# Acl Statement in /etc/named.conf

- acl (access control)


```
acl black-hats {  
    10.0.2.0/24;  
    192.168.0.0/24; };  
acl red-hats {  
    10.0.1.0/24; };  
options {  
    blackhole { black-hats; };  
    allow-query { red-hats; };  
    allow-recursion { red-hats; }; }
```

Diagram annotations:

  - Line pointing to `black-hats`: acl-name
  - Line pointing to `10.0.2.0/24`: match-element



# Option Statement in /etc/named.conf

- **allow-query** — Specifies which hosts are allowed to query this nameserver. By default, all hosts are allowed to query. An access control list, or collection of IP addresses or networks may be used here to only allow particular hosts to query the nameserver.
  - **allow-recursion** — Similar to allow-query, this option applies to recursive queries. By default, all hosts are allowed to perform recursive queries on the nameserver.
  - **blackhole** — Specifies which hosts are not allowed to query the server.
  - **directory** — Changes the named working directory to something other than the default value, /var/named/.
- 


# Option Statement in /etc/named.conf

- **forward** — Controls forwarding behavior of a forwarders directive.
  - **first** — Specifies that the nameservers specified in the forwarders directive be queried before named attempts to resolve the name itself.
  - **only** — Specifies that named not attempt name resolution itself in the event queries to nameservers specified in the forwarders directive fail.
- **forwarders** — Specifies a list of valid IP addresses for nameservers where requests should be forwarded for resolution.
- **listen-on** — Specifies the network interface on which named listens for queries. By default, all interfaces are used.

```
options {  
    listen-on { 10.0.1.1; };  
};
```



# Zone Statement in `/etc/named.conf`

- `allow-query` — Specifies the clients that are allowed to request information about this zone. The default is to allow all query requests.
  - `allow-transfer` — Specifies the slave servers that are allowed to request a transfer of the zone's information. The default is to allow all transfer requests.
  - `allow-update` — Specifies the hosts that are allowed to dynamically update information in their zone. The default is to deny all dynamic update requests.
  - `file` — Specifies the name of the file in the named working directory that contains the zone's configuration data.
  - `masters` — The `masters` option lists the IP addresses from which to request authoritative zone information. Used only if the zone is defined as type `slave`.
- 



# Example Zone Statements


```
zone ``example.com" IN {  
    type master;  
    file ``example.com.zone";  
    allow-update { none; };  
};
```

```
zone ``example.com" {  
    type slave;  
    file ``example.com.zone";  
    masters { 192.168.0.1; };  
};
```





# Zone Statement in /etc/named.conf

- notify — Controls whether named notifies the slave servers when a zone is updated.
    - yes — Notifies slave servers.
    - no — Does not notify slave servers.
    - explicit — Only notifies slave servers specified in an *also-notify* list within a zone statement.
  - type — Defines the type of zone.
    - forward — Forwards all requests for information about this zone to other nameservers.
    - hint — A special type of zone used to point to the root nameservers which resolve queries when a zone is not otherwise known. No configuration beyond the default is necessary with a hint zone.
    - master — Designates the nameserver as authoritative for this zone. A zone should be set as the master if the zone's configuration files reside on the system.
    - slave — Designates the nameserver as a slave server for this zone. Also specifies the IP address of the master nameserver for the zone.
- 



## Zone Statement in /etc/named.conf

- For each zone statement

```
zone "example.com" IN {  
    type master;  
    file "example.com.zone";  
    allow-update { none; };  
};
```


there is a corresponding zone file,  
example.com.zone in /var/named/





# Zone Files

## Contains

- Directives
  - Resource records
    - Define the parameters of the zone
    - Assign identities to individual hosts
- 




# Zone File Directives

- **\$INCLUDE** — Configures named to include another zone file in this zone file at the place where the directive appears. This allows additional zone settings to be stored apart from the main zone file.
- **\$ORIGIN** — Appends the domain name to unqualified records, such as those with the hostname and nothing more.


**\$ORIGIN** example.com

Any names used in resource records that do not end in a trailing period (.) will have example.com appended to them.





# Zone File Directives

- \$TTL — Sets the default *Time to Live (TTL)* value for the zone. This is the length of time, in seconds, a zone resource record is valid. Each resource record can contain its own TTL value, which overrides this directive.
- 



# Zone File Resource Records

- A — Address record, which specifies an IP address to assign to a name, as in this example:

*<host> IN A <IP-address>*

- CNAME — Canonical name record, maps one name to another. This type of record is also known as an alias record.

*<alias-name> IN CNAME <real-name>*

server1 IN A 10.0.1.5

www IN CNAME server1





# Zone File Resource Records

- MX — Mail eXchange record, which tells where mail should go.

IN MX *<preference-value>* *<email-server-name>*

*<preference-value>* allows numerical ranking of the email servers for a namespace.

IN MX 10 mail.example.com.

IN MX 20 mail2.example.com.





# Zone File Resource Records

- NS — NameServer record, which announces the authoritative nameservers for a particular zone.

IN NS *<nameserver-name>*

IN NS dns1.example.com.

IN NS dns2.example.com.



# Sample Zone File

\$ORIGIN example.com

\$TTL 86400

```
@      IN  SOA  dns1.example.com. hostmaster.example.com. (
        2001062501 ; serial
        21600 ; refresh after 6 hours
        3600 ; retry after 1 hour
        604800 ; expire after 1 week
        86400 ) ; minimum TTL of 1 day
      IN  NS   dns1.example.com.
      IN  NS   dns2.example.com.
      IN  MX   10    mail.example.com.
      IN  MX   20    mail2.example.com.
      IN  A    10.0.1.5
server1 IN  A    10.0.1.5
server2 IN  A    10.0.1.7
dns1    IN  A    10.0.1.2
dns2    IN  A    10.0.1.3
ftp     IN  CNAME server1
mail    IN  CNAME server1
mail2   IN  CNAME server2
www     IN  CNAME server2
```

# Sample Reverse Zone File

```
$ORIGIN 1.0.10.in-addr.arpa
```

```
$TTL 86400
```

```
@          IN      SOA     dns1.example.com. hostmaster.example.com. (
                2001062501 ; serial
                21600 ; refresh after 6 hours
                3600  ; retry after 1 hour
                604800 ; expire after 1 week
                86400 ) ; minimum TTL of 1 day

          IN      NS      dns1.example.com.
          IN      NS      dns2.example.com.
20       IN      PTR     alice.example.com.
21       IN      PTR     betty.example.com.
22       IN      PTR     charlie.example.com.
23       IN      PTR     doug.example.com.
```

```
<last-IP-digit>  IN  PTR  <FQDN-of-system>
```



# Using rndc

- BIND includes a utility called rndc which allows command line administration of the named daemon from the localhost or from a remote host.
  - In order to prevent unauthorized access to the named daemon, BIND uses a shared secret key method is used to grant privileges to hosts.
    - This means an identical key must be present in both `/etc/named.conf` and the rndc configuration file, `/etc/rndc.conf`
- 

# Controls Statement in /etc/named.conf

- In order for rndc to connect to a named service, there must be a controls statement in the BIND server's /etc/named.conf file.
- The controls statement below shown allows rndc to connect from the localhost.

- ```
controls {  
    inet 127.0.0.1 allow { localhost; } keys { <key-name>; };  
};
```

This statement tells named to listen on the default TCP port 953 of the loopback address and allow rndc commands coming from the localhost, if the proper key is given.

# Controls Statement in /etc/named.conf

- The *<key-name>* relates to the key statement, which is also in the /etc/named.conf file.

- key "*<key-name>*" {  
    algorithm hmac-md5;  
    secret "*<key-value>*";  
};

In this case, the *<key-value>* is a HMAC-MD5 key. The same key statement should appear in /etc/rndc.conf

# HTTP

## Reading:


1. Red hat Linux 9: Red Hat Linux Reference  
Chapter 10 Apache HTTP Server
2. Red Hat Linux 9: Red Hat Linux Customization  
Guide Chapter 19 Apache HTTP Server  
Configuration



# Directives in Apache Configuration File

- /etc/httpd/conf/httpd.conf

- Directives

- **Listen:** identifies the ports on which the Web server will accept incoming requests. By default, the Apache HTTP Server is set to listen to port 80 for non-secure Web communications and (in the /etc/httpd/conf.d/ssl.conf which defines any secure servers) to port 443 for secure Web communications.
  - **ServerRoot: sets the** top-level directory which contains the server's files. Both the secure server and the non-secure server set the ServerRoot directive is set to "/etc/httpd".
- 




# Directives in Apache Configuration File

## Directives

- **User:** sets the user name of the server process and determines what files the server is allowed to access. By default User is set to apache.
- **Group:** specifies the group name of the Apache HTTP Server processes. By default Group is set to apache.
- **ServerAdmin:** sets the email address of the Web server administrator. This email address will show up in error messages on server-generated Web pages.


A common way to set up ServerAdmin is to set it to `webmaster@example.com`. Then alias webmaster to the person responsible for the Web server in `/etc/aliases` and run `/usr/bin/newaliases`.





# Directives in Apache Configuration File

## ● Directives

- **ServerName:** sets a hostname and port number (matching the Listen directive) for the server. The ServerName does not need to match the machine's actual hostname. However, the value specified in ServerName must be a valid Domain Name Service (DNS) name that can be resolved by the system.
  - **DocumentRoot:** sets the directory which contains most of the HTML files which is served in response to requests. The default is the `/var/www/html` directory.
- 

# Directives in Apache Configuration File

## • Directives

- **Directory** `<Directory /path/to/directory>` and `</Directory>` tags create what is referred to as a *container* and are used to enclose a group of configuration directives meant to apply only to a particular directory and its subdirectories.

```
<Directory />
```

```
Options FollowSymLinks Indexes
```

```
AllowOverride None
```

```
</Directory>
```