Configuring DNS

- BIND: UNIX Name Service
  - Configuring the BIND name server (named)
  - Configuring the BIND resolver
  - Constructing the name server database files
    - Zone: a collection of domain information contained in a DNS database file.
      - Basic set of standard records defined in RFC1033
      - More new DBS records in many RFCs, not widely implemented.

BIND Configuration

- Four level service
  - Resolver-only system
  - Master servers
  - Slave servers.
  - Caching-only servers
- Resolver-only system
  - Only need to setup /etc/resolv.conf
- Master
  - Zone files for Forward-mapping zone
  - Zone files for reverse-mapping zone
  - Conf file
  - Root hint file
  - Loopback file

BIND Configuration

- Slave server
  - Transfer zone files
  - boot file
  - Cache file
  - Lookbackfile
- Caching-only
  - Boot file
  - Cache file
  - Possible loopback

Configuring the Resolver

- /etc/resolv.conf
  - Identify up to three name servers
  - Default domain
  - Search list
  - Other options.
- name server address
  - Use IP address
  - Name servers are queried in the order
  - Use official IP for localhost or with 0.0.0.0, not loopback address
- domain name
  - Append the default domain name to any hostname that does not contain a dot. This is configurable.
  - Env LOCALDOMAIN overrides domain entry

Configuring the Resolver (Cont)

- search domain …
  - Defines a series of domains that is searched when a hostname does not contain a dot.
  - Use either search statement or domain statement. Never use both.
  - LOCALDOMAIN overrides the search entry.
- sortlist network[/network] …
  - Address from networks listed on the sortlist command are preferred.
  - For multi-homes host or a router.
  - Rarely used.

Configuring the Resolver (Cont)

- options option …
  - debug *works only if compiled with -DDEBUG*
  - ndots:n *determine if search list is appended*
  - timeout:n *initial query timeout*
  - attempts:n *# of times will try a query with each server.*
  - rotate *share the workload among all servers.*
  - no-check-names *if you must work with hostname with -*
  - inet6 *query fro IPv6*
- All lab machines are configured as resolver-only.

Exercise

- Use 'man resolv.conf' as a resource
- Configure your dns client, so you can
  - Ping wopr (return wopr.csl.mtu.edu)
  - Ping wopr.csl (return wopr.csl.mtu.edu)
  - Ping tamarack (return tamarack.cs.mtu.edu)
  - Ping tamarack.cs (return tamarack.cs.mtu.edu)
  - Ping mail (return mail.mtu.edu)

Configuring named

- The configuration file.
  - General named parameters
  - Usually called named.conf
- The root hints file
  - Points to root domain servers.
  - Common names are: named.ca, db.cache, named.root or root.ca
- Local host file
  - Resolve loopback address
  - Generally named.local

Configuring named(cont)

- The forward-mapping zone file
  - Maps hostname to IP
  - This is the file contains bulk of info about a zone, typically named domain.hosts, amherst.hboc.com.hosts
- The reverse-mapping zone file
  - Maps IP addresses to hostnames
  - Reverse zone are generally use network.rev, like 172.16.rev
- You can use any name, but following the convention and use descriptive names for zone files make maintaining easier.

The Configuration file named.conf

- Points named to the sources of DNS information: *local files or remote servers.*
- Named.conf controls what kind of nameserver (master, slave, or caching-only) for what domain.
- The structure is like C programming language.
  - Statement end with semicolon (;)
  - Literals are enclosed in quotes (“”)
  - Related items are grouped in braces ({})
  - Comment can be enclosed between /* and */, or after //, or after #

Some named.conf configuration commands.

Command	Function
acl	Defines an access control list of IP addresses
include	Includes another file into the configuration file
key	Defines security keys for authentication
logging	Defines what will be logged and where it will be stored
options	Defines global configuration options and defaults
server	Defines a remote server's characteristics
zone	Defines a zone.

Notes; See Appendix C for more detailed info.

A caching-only server configuration

- /etc/named.conf

```
options {
        directory "/var/named"
};

 zone "." {
          type hint;
          file "named.ca"
};

zone "0.0.127.in-addr.arpa" {
           type master;
          file "named.local";
}
```

Default directory for named

Name server use it to locate the root server

Make itself the master for localhost

- Options can be more complex.
  - forwarders
  - forward only

Master server configuration

- /etc/named.conf

```
options {
        directory "/var/named"
};
 zone "." {
         type hint;
         file "named.ca"
};
zone "0.0.127.in-addr.arpa" {
         type master;
         file "named.local";
}
 zone "wrotethebook.com" {
         type master;
         file "wrotethebook.com.hosts"
}
 zone "16.172.in-addr.arpa" {
         type master;
         file "172.16.rev";
}
```

Data for wrotethebook.com domain
Is in file /var/named/wrotethebook.com.hosts

Data for IP address to host from 172.16.0.0
Is in file /var/named/172.16.rev

Slave server configuration

- /etc/named.conf

```
options {
        directory "/var/named"
};
 zone "." {
         type hint;
         file "named.ca"
};
zone "0.0.127.in-addr.arpa" {
         type master;
         file "named.local";
}
 zone "wrotethebook.com" {
         type slave;
         file "wrotethebook.com.hosts"
         masters {172.16.12.1 };
}
 zone "16.172.in-addr.arpa" {
         type slave;
         file "172.16.rev";
         masters {172.16.12.1 };
}
```

Define itself as slave server,
specify where is master server
and which file to keep a local copy

- Slave server only transfer the zone when the data changes.

We've looked at different named.conf for all three types of servers.

Now let's take a look at other files, the zone file and zone reverse file, named.local and named.ca
  - They all have the same basic format
  - and use the same type of database records.

Standard Resource Records

Resource Record Text Name	Record Type	Function
Start of Authority	SOA	Marks the beginning of a zone's data, and defines parameters that affect the entire zone.
Nameserver	NS	Identifies a domain's nameserver.
Address	A	Converts a hostname to an address.
Pointer	PTR	Converts an address to a hostname.
Mail Exchange	MX	Identifies where to deliver mail for a given domain name.
Canonical Name	CNAME	Defines an alias hostname.
Text	TXT	Stores arbitrary text strings.

Standard Resource Records

- The format of DNS record is
  [*name*] [*ttl*] class *type data*

  name: domain object
  - host or entire domain. Relative to current domain
  - if empty, the record applied to domain object that was named last.

  ttl: time to live
  - How long should be kept in remote's cache
  - Default $TTL for entire zone

  class: address class of the resource record
  - IN: internet DNS resource records.
  - HS: in Hesoid Name Server developed by MIT

  type: kind of resource record.
  - SOA, A, PTR, MX, CNAM, …

  data: The information specific to this type of resource record.

Zone File Directives

- $TTL
- $ORIGIN
- $INCLUDE
- $GENERATE

Start of Authority Record (SOA)

- Each zone has only one SOA record
- The format of SOA record is

  [zone] [ttl] IN SOA *origin contact (*
  Serial
  Refresh
  Retry
  Expire
  negative_cache_ttl)

- For example:

```
@ IN SOA crab.wrotethebook.com. David.crab.wrotethebook.com. (
    2005061801  ; serial
    21600       ; refresh four times a day
    1800        ; retry every half hour
    604800      ; expire after 1 week
    900         ; negative cache ttl is 15 minutes
 )
```


Name Server Record (NS)

- Identify the authoritative servers for a zone
  - Link domain hierarchy together
    - NS record in top-level domain point to the servers for the second-level domains,
    - Which in turn contain NS records that point to the servers for their subdomains
- The format is

  [domain] [ttl] IN NS *server*

- *Example:*

```
;       Define sub-domains
plant           IN      NS      pack.plant.nuts.com.
                IN      NS      pecan.nuts.com.
sales           IN      NS      acorn.sales.nuts.com.
                IN      NS      pack.plant.nuts.com.
```

Address Record (A)

- Convert hostnames to IP addresses
- Format

  [host] [ttl] IN A address
  - Host name is often written relative to the current domain.
- Example:

```
pecan           IN      A       172.16.12.3
walnut          IN      A       172.16.12.4
filbert         IN      A       172.16.1.2
```


Mail Exchanger Record (MX)

- Redirect mail to a mail server
  - For individual computer
  - For an entire domain
- The Format:

  [name] [ttl] IN MX preference host
  - Name: the value after @ in the mailing address
  - Preference:
    - more than one MX records.
    - Low preference are tried first
  - The name of the mail server
- Example:

```
peanut                  IN      MX       5  almond.nuts.com.
Wrotethebook.com        IN      MX      10 crab.wrotethebook.com
*.wrotethebook.com      IN      MX      10 crab.wrotethebook.com
```

Canonical Name Record (CNAME)

- Defines an alias for the official name of a host
  - Nick name for convenience of users
  - Easy the transition from an old hostname to new
- Format

  Nickname [ttl] IN CNAME host
- Example

```
goober          IN      CNAME   peanut.nuts.com.
```


Domain Name Pointer Record(PTR)

- Convert numeric IP address to hostnames
  - Some program use reverse domain map IP addresses to hostnames to show status
  - Some service provides use the reverse domain to track who is using their services
- Format

  Name [ttl] IN PTR host
  - Name: a number, relative to the current in-addr.arpa domain.
- Example

```
1.12            IN      PTR     almond.nuts.com.
2.12            IN      PTR     peanut.nuts.com.
```

Zone File Directives

- Four Directives
  - Simplify zone file construction
    - $INCLUDE
    - $GENERATE
  - Define values used by the resource records
    - $ORGIN
    - $TTL

Zone File Directives

- $ORIGIN
  - Set the current origin
    - used to complete any relative domain names.
  - Default to domain name defined on the zone statements.

    zone "wrotethebook.com" {

    …

    }
- $INCLUDE
  - Read external file and includes as part of zone.

Zone File Directives

- $GENERATE
  - Create a series of RR.
  - For example:
    - $GENERATE 1-4 $ CNAME $.1to4

      1-4: The iterator value

      $ CNAME $: template of the resource records to be generated, which $ replaced with current iterator value
    - This will produce:

      1 CNAME 1.1to4

      2 CNAME 2.1to4

      3 CNAME 3.1to4

      4 CNAME 4.1to4
    - Given that the current origin is 20.16.172.in-addr.arpa, the above generate statement are same as:

      1.20.16.172.in-addr.arpa CNAME 1.1to4.20.16.172.in-addr.arpa

      2.20.16.172.in-addr.arpa CNAME 2.1to4.20.16.172.in-addr.arpa

      … …

Zone File Directives

- $TTL
  - Defines the default TTL for RR
  - Use a number of seconds or combination of numbers and letters.
    - w, d, h, m ,s
  - Example: define TTL to be one week.
    - $TTL 604800
    - $TTL 2w

Cache Initialization file

- Zone statement that has its type set to **hints** points to the cache initialization file.

```
zone "." {
        type hint;
        file "named.ca"
};
```

- Root domain is indicated by "**.**" on zone statement
- **named.ca** contains hints to initialize the cache.
  - Names and addresses of the root servers.
  - Help the local server locate a root server during startup
  - Get a ready to use copy from ftp.rs.internic.net every couple of months
- If you system is not connected to the Internet, use the major name servers on your local network instead.


```
... ...
;       last update:    Jan 29, 2004
;       related version of root zone:   2004012900
;
;
; formerly NS.INTERNIC.NET
;
.                       3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
;
; formerly NS1.ISI.EDU
;
.                       3600000      NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.     3600000      A     192.228.79.201
;
; formerly C.PSI.NET
;
.                       3600000      NS    C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.     3600000      A     192.33.4.12
;
; formerly TERP.UMD.EDU
;
.                       3600000      NS    D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET.     3600000      A     128.8.10.90
;
; formerly NS.NASA.GOV
;
.                       3600000      NS    E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET.     3600000      A     192.203.230.10
```

The named.local file

- Convert the address 127.0.0.1 to name localhost.

```
$TTL        86400
@           IN  SOA     almond.nuts.com. jan.almond.nuts.com. (
                        1               ; serial
                        360000          ; refresh every 100 hours
                        3600            ; retry after 1 hour
                        3600000         ; expire after 1000 hours
                        360000          ; default ttl is 100 hours
                        )
            IN  NS      almond.nuts.com.
0           IN  PTR     loopback.
1           IN  PTR     localhost.
```


The Reverse Zone File

- Similar to named.local
- Translate IP address to hostname, so contains PTR records
- In named.conf, we have

```
zone "16.172.in-addr.arpa" {
        type master;
        file "172.16.rev";
}
```

File 172.16.rev

```
;
;        Address to hostname mappings.
;
@       IN      SOA     almond.nuts.com. jan.almond.nuts.com. (
                                10099   ;   Serial
                                43200   ;   Refresh
                                3600    ;   Retry
                                3600000 ;   Expire
                                2592000 ) ; Minimum
                IN      NS      almond.nuts.com.
                IN      NS      filbert.nuts.com.
                IN      NS      foo.army.mil.
1.12            IN      PTR     almond.nuts.com.
2.12            IN      PTR     peanut.nuts.com.
3.12            IN      PTR     pecan.nuts.com.
4.12            IN      PTR     walnut.nuts.com.
2.1             IN      PTR     filbert.nuts.com.
6               IN      NS      salt.plant.nuts.com.
                IN      NS      pecan.nuts.com.
```


The Reverse Zone file

- @ in SOA: allows the zone statement to define the zone file domain
- NS record
- PTR record
  - If not ends in dots, relative to current domain

  For example: 1.12 is interpreted as 1.12.16.172.in-addr-arpa
- Additional NS records for sub domain
  - NS records for subdomain must be placed in the higher-level domain before subdomain can be used.
- Four bytes of in-addr.arpa domain are treated as four instinct pieces of a name.
  - Byte boundaries make less flexible than subnet which is bit-oriented.
  - Use $GENERATE helps to create more flexible reverse domain delegations.

The Reverse Zone file

Example:
Assume there are four subdomains
192.168.30.0 – 192.168.30.63
192.168.20.64 – 192.168.30.127
192.168.20.128 – 192.168.30.191
192.168.20.192 – 192.168.30.255
How to define the reverse subdomains?
Recall that the reverse domain name in zone statement and $ ORIGIN are byte bounded. Use $GENERATE to create ranges.

$ORIGIN 30.168.192.in-addr.arpa
$GENERATE 0-63 $ CNAME $.1ST64
$GENERATE 63-127 $ CNAME $.2nd64
$GENERATE 128-191 $ CNAME $.3rd64
$GENERATE 192-255 $ CNAME $.4th64

Which map the 256 numeric names to four other domains
1st64.30.168.192.in-addr.arpa domain
2nd64.30.168.192.in-addr.arpa domain
3rd64.30.168.192.in-addr.arpa domain
4th64.30.168.192.in-addr.arpa domain
When a request of finding PTR of 52.30.168.192.in-addr.arpa in,
1. it found the canonical name 52.1st64.30.168.192.in-addr.arpa
2. see the PTR from server fro 1st64.30.168.192.in-addr.arpa domain

The forward-mapping zone file

- Covert hostnames to IP addresses, which are A records
- Also contains MX, CNAME, and other records.
- MX record
  - Identify a mail server for the entire domain.
  - Identify a mail server for a a host
  - Preference number. The lower the number, the more desirable .
- CNAME
  - Alias for the official hostname, which is called canonical name
  - Alias should not be used in other resource records.

Forwarding map zone file

- Need the file specified in named.conf zone statement.

```
zone "nuts.com" {
        type master;
        file "nuts.com.hosts"
}
```



```
;
;       Addresses and other host information.
;
@       IN      SOA     almond.nuts.com. jan.almond.nuts.com.
(
                                10118      ; Serial
                                43200      ; Refresh
                                3600       ; Retry
                                3600000    ; Expire
                                2592000 )  ; Minimum
;       Define the nameservers and the mail servers
                IN      NS      almond.nuts.com.
                IN      NS      filbert.nuts.com.
                IN      NS      foo.army.mil.
                IN      MX      10 almond.nuts.com.
                IN      MX      20 pecan.nuts.com.
;
;       Define localhost
;
localhost       IN      A       127.0.0.1
;
```

```
;       Define the hosts in this zone
almond          IN      A       172.16.12.1
                IN      MX      5 almond.nuts.com.
loghost         IN      CNAME   almond.nuts.com.
peanut          IN      A       172.16.12.2
                IN      MX      5 almond.nuts.com.
goober          IN      CNAME   peanut.nuts.com.
pecan           IN      A       172.16.12.3
walnut          IN      A       172.16.12.4
filbert         IN      A       172.16.1.2
;       host table has BOTH host and gateway entries for 10.104.0.19
mil-gw          IN      A       10.104.0.19
;
;    Glue records for servers within this domain
pack.plant      IN      A       172.16.18.15
acorn.sales     IN      A       172.16.6.1
;
;       Define sub-domains
plant           IN      NS      pack.plant.nuts.com.
                IN      NS      pecan.nuts.com.
sales           IN      NS      acorn.sales.nuts.com.
                IN      NS      pack.plant.nuts.com.
```


Controlling the named process

- After configuration, start process named
- Named is usually started at boot time.
  - On Solaris, /etc/init.d/inetsvc
  - On Red Hat Linux, /etc/rc.d/init.d/named
    - # /etc/rc.d/init.d/named start
    - # /etc/rc.d/init.d/named stop
- Name Control program ndc and rndc

Using nslookup

- Debugging
- Query a name server directly and retrieve the info
  - Interactively or directly from command line

```
[ruihong@cslserver ~]$ /usr/sbin/nslookup google.com
Server:  dns1.cec.mtu.edu
Address:  141.219.152.253

Name:    google.com
Addresses:  216.239.57.99, 216.239.39.99, 216.239.37.99
```


More about nslookup

- Control the server to be used
  server *.*.*
- Control the domain name used in the session
  set domain=*.*
- Control records type to be queried
  set type=NS / A /any / *
- down load an entire domain from authoritative server
  ls *.* > filename
  For security reason, most sites do not implement ls.
- Check for more features in nslookup using help command in interactive mode.

Exercise

- Use "set type=XX" to:
  - Find the NS for mtu.edu
  - Find the NS for csl.mtu.edu
  - Find the MX record for
    - username@mtu.edu
    - username@yahoo.com
    - usename@csl.mtu.edu
- Run nslookup a_secret_host twice. Any difference of the output?
  - Ex: nslookup pku.edu.cn

Summary

- DNS is a important user service
- BIND
  - Client resolver: /etc/resolv.conf
  - Server named: /etc/named.conf
    - Type of server
    - Where are the database file
      - Zone file
        - » Resource Records
- Use nslookup to test the implementation