

# Unix System Administration

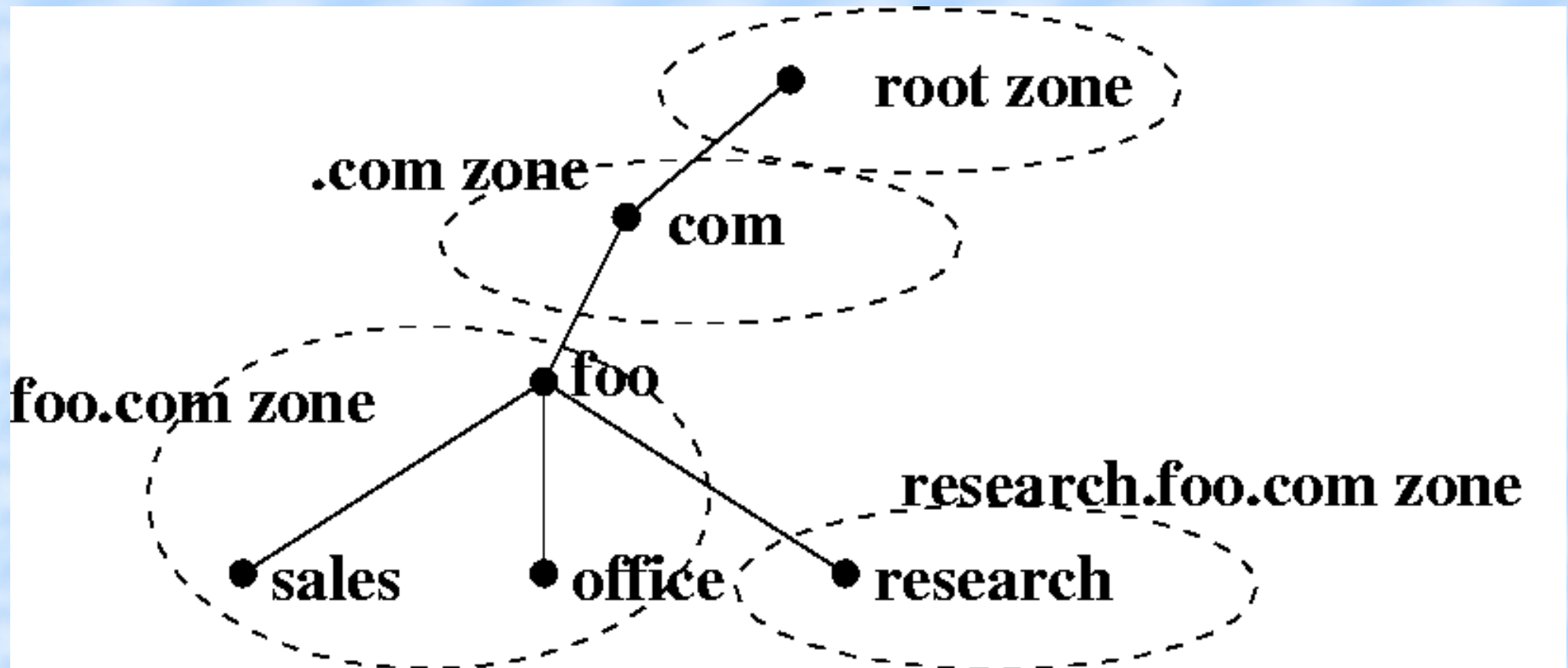
**Chris Schenk**

**Lecture 14 – Tuesday Mar 06**

**CSCI 4113, Spring 2007**

# DNS – Review

- NS records bind the DNS world together
- Every parent knows about its children



# DNS – Recursive Queries

- If I don't have any info about foo.com
  - Talk to one of root nameservers
- Root servers punt me to .com nameservers
- .com servers punt me to foo.com nameservers
- foo.com answers authoritatively for its domain
  - Or it may continue to punt!
  - Delegate authority to research.foo.com
  - 'A' records mean something different in this case
- NS records are ALWAYS accompanied by A records

# DNS - /etc/resolv.conf, /etc/nsswitch.conf

- DNS queries on your machine start here
  - Specify up to three nameservers
  - Three search domains also configurable
- If a nameserver is down, try the next one
  - After a nice, annoying timeout
- Not all queries go through DNS first
  - /etc/hosts also investigated for information
  - Handled by the Name Service Switching utilities
  - /etc/nsswitch.conf

# DNS – Resource Record Format

- `[name] [ttl] [class] <type> <data>`
  - The name is a domain and is sometimes optional
  - The '@' sign is shorthand for the domain-name of the zone
    - Can be changed with the \$ORIGIN directive
  - When consecutive records refer to the same name
    - Only first record must specify the name
    - Subsequent records **MUST** start with a blank space
    - When names **ARE** specified they **MUST** start at column one
  - Unqualified names (i.e. names without a trailing dot) have the domain appended to the name

# DNS – Resource Record Format (cont)

- Fully qualified names **MUST** have a trailing dot
  - You know when you're missing a trailing dot with names like
  - `csel.cs.colorado.edu.cs.colorado.edu.`
- The TTL record can be set explicitly on a record
  - Although never really seen, always uses the default \$TTL at the top
- The 'class' field defaults to the IN class
  - The 'internet' class, the most common
  - Two other classes are CH from ChaosNet and HS from Hesiod
- Type field specifies the Resource Record type
  - e.g. A, NS, TXT, etc

# BIND Zone Files - \$TTL Directive

- First entry must be the \$TTL directive
  - Sets the default Time-To-Live on all data in the zone
    - The amount of time, in seconds, that a remote server will cache the resource record data
  - Syntax is simple
    - \$TTL 7200
  - Number of seconds is specified as an integer
  - \$TTL is not specified in the SOA record
    - Minimum TTL is specified in SOA, but is different!!!

# BIND Zone Files – SOA Record

- Start of Authority record comes after \$TTL
- Specifies a few things:
  - Root name for the zone
  - Fully qualified Master nameserver name
  - Email address for person responsible for the server
  - Serial no. to identify current zone data
  - refresh, retry, expiry, and minimum TTL fields
- All time values expressed in seconds or symbolic form
  - 7200 or 2H

# BIND Zone Files – SOA Record (cont)

- A typical SOA RR looks like this:
- ```
@ IN SOA cs.colorado.edu. admin.cs.colorado.edu. (  
  20010421 ; Serial Number  
  86400    ; Refresh - every 24 hours  
  1800     ; Retry   - 30 minutes  
  1209600  ; Expire  - 2 weeks  
  7200    ) ; Negative answers - 2 hours
```
- Name
  - Root name of the zone
  - '@' sign is shorthand reference to the zone configured in `/etc/named.conf` for this file

# BIND Zone Files – SOA Record (cont)

- `@ IN SOA cs.colorado.edu. admin.cs.colorado.edu. (`  
    `20010421 ; Serial Number`  
    `86400 ; Refresh - every 24 hours`  
    `1800 ; Retry - 30 minutes`  
    `1209600 ; Expire - 2 weeks`  
    `7200 ) ; Negative answers - 2 hours`
- Class is Internet 'IN'
- Type of record is SOA
- Name of Master nameserver is `cs.colorado.edu`
  - MUST have a trailing dot
  - This is NOT the root of the zone!

# BIND Zone Files – SOA Record (cont)

- @ IN SOA cs.colorado.edu. **admin.cs.colorado.edu.** (  
    **2001042100** ; Serial Number  
    86400 ; Refresh - every 24 hours  
    1800 ; Retry - 30 minutes  
    1209600 ; Expire - 2 weeks  
    7200 ) ; Negative answers - 2 hours
- Contact email address for responsible person
  - Also must have a trailing dot
  - The '@' is replaced with a period '.' because @ has special meaning in zone files
- Serial No.
  - Unique identifier for current zone information

# BIND Zone Files – SOA Record (cont)

- @ IN SOA cs.colorado.edu. admin.cs.colorado.edu. (  
2001042100 ; Serial Number  
**86400** ; Refresh - every 24 hours  
**1800** ; Retry - 30 minutes  
1209600 ; Expire - 2 weeks  
7200 ) ; Negative answers - 2 hours

- Refresh field

- Tells a secondary nameserver how long to wait between checking for updates to the zone

- Retry field

- If the master is down, how long to wait before the next attempt to grab the zone

# BIND Zone Files – SOA Record (cont)

- @ IN SOA cs.colorado.edu. admin.cs.colorado.edu. (  
20010421 ; Serial Number  
86400 ; Refresh - every 24 hours  
1800 ; Retry - 30 minutes  
**1209600** ; Expire - 2 weeks  
7200 ) ; Negative answers - 2 hours
- Expire field
  - Tells a secondary how long the data they hold is *authoritative*
    - Usually is a very large value for potentially long outages
- Minimum TTL field – two possible meanings
  - How long negative answers (NXDOMAIN) are to be cached
  - Also could be the TTL for all resource records

# BIND Zone Files – SOA Record Values

- All values can be tweaked from short to long
  - Tradeoffs for efficiency and flexibility
  - Short timeouts mean accurate data (quick changing zones)
  - Long timeouts lend to longer caching (efficiency)
- Refresh value isn't as important anymore
  - Most servers notify secondaries for a zone change
- Expire value should be very long
  - Somewhere between 2-4 weeks

# BIND – Daemon Information

- BIND can run in a 'chrooted jail'
  - 'chroot' means 'change root' – `man chroot`
    - Changes the location of the '/' directory to a subdirectory for your shell
  - Means that the process only sees a subset of the filesystem
  - Used to prevent an exploited process from doing bad things
  - Many other daemons can also run in this way
    - SSHD, Apache, and others
- Jail location is usually `/var/named`
  - All config files, zone files, everything for BIND exist here

# Process Signaling Tangent

- Unix allows the sending of signals to a process
  - Used for inter-process communication
- Many different types of signals to send
  - Hangup, kill, interrupt, stop, and others
- Some signals can be caught by process to perform a task
  - Hangup is most common used for a 'reload'
  - Usually reloads configuration from files on disk
- The `kill` command is used to signal another process

# Process Signaling – kill

- Usage: `kill [-signal] <pid>`
- Using `kill` without a signal sends a TERM signal by default
- Send a HANGUP signal to a process
  - `kill -HUP 4758`
- Send a suspend signal to a process (similar to CTRL-Z)
  - `kill -STOP 7472`
- Send continue signal to process (similar to fg)
  - `kill -CONT 7472`

# Controlling BIND

- All versions of BIND respond to various signals
  - BIND reloads configuration with a HUP signal
- BIND 9 has a control command called `rndc`
  - Speaks to BIND over TCP
  - `rndc` requires its own configuration under `/etc/rndc.conf`
  - Use of `rndc` requires a secret key exchanged for authentication
    - Uses a Message Authentication Code (MAC) function to verify key
    - Key MUST be shared between both hosts
  - Send commands to see the status or reload configuration

# BIND – named Configuration

- Config file `/etc/named.conf`
  - Comments with either `#`, `//`, or `/* ... */`
    - Semicolons are NOT comments unlike zone files!
  - All config statements end with a semicolon ;
- Statements in the file include the following
  - `acl`, `include`, `options`, `key`, `trusted-keys`, `server`, `controls`, `zone`, `logging`, `view`
- Most statements are used more than once
  - Except for 'options' and 'logging'

# BIND – named Configuration (cont)

- Address Match Lists specify a set of IP addresses
  - ACLs are labeled versions of these lists
  - List may consist of one or more of the following types
    - IP address
    - CIDR mask
    - Previously defined ACL
    - A negated version of any of the above with a preceding ! sign
  - Address may also utilize a keyword
    - any, none, localhost (127.0.0.1), localnets (networks attached to network interfaces)

# BIND – acl Statement

- ```
acl <acl-id> {  
    ! 1.2.3.4; 1.2.3/24; localhost;  
};
```
- First-match wins for ACLs
- ACLs must be top-level in the config file
  - Can't be defined within other blocks
- Must also be defined before it is used
  - Otherwise named will barf on startup

# BIND – include Statement

- `include "/path/to/file";`
- Absolute paths start at the chrooted jail
  - Relative paths start at the working directory of named
- Does a simple text-insert of the include file
  - The include must create a syntactically correct config file
- Usually rndc keys are included in this manner
  - So the config file is world-readable but the key file is not

# BIND – options Statement

- There can be only one!
  - Error is returned otherwise
- Defines global options
  - Can be individually overridden in sub-statements
- Over 50 options available, all viewable in the manpage
  - We'll go over important ones
  - Many setups only require a few options

# BIND – options Configuration

- ```
options {  
    version      "we're not telling!";  
    directory    "/";  
};
```
- **version** - specifies the text to return on a version query
  - If omitted, shows real version by default
- **directory** - sets named's runtime directory
  - "/" for chrooted servers
  - /var/named for non-chrooted servers

# BIND – options Configuration (cont)

- ```
options {  
    . . .  
    notify yes;  
    also-notify { IP1 port 1053; IP2; };  
    listen-on [port NN] {acl_id3}  
};
```
- **notify** – turns on secondary server notification
- **also-notify** – notifies secondaries of new zone data
- **listen-on** – optionally specify a port and a list of allowed addresses to query the server

# BIND – key Statement

- ```
key key_id {  
    algorithm hmac-md5; // the only alg  
    secret "XYZZY";  
};
```
- Defines a secret key and MACing algorithm to use
- 'secret' is an unencrypted base64 encoded string
- 'key' statement is top-level
  - Must be unique between different servers if you are managing from a single machine
  - Must match in the server and client rndc config

# BIND – controls Statement

- ```
controls {  
    inet IP_ADDR [port N] allow { acl_id8 }  
    keys { key_id1; key_id2; };  
};
```
- Configures access for rndc
- Multiple 'inet' clauses may be given
- IP\_ADDR is the inet interface to listen on
  - use '\*' for all interfaces
- Port number defaults to 953
- Keys must be used in BIND 9

# BIND - /etc/rndc.conf

- Client rndc configuration, similar to named.conf
- Only four valid statements - key, options, server, include

- key statement looks exactly like it does in named.conf

- ```
key key_id {  
    algorithm hmac-md5; // the only alg  
    secret "XYZZY";  
};
```

- Typical options:

- ```
options {  
    default-server localhost;  
    default-key key_id;  
};
```

# BIND – zone Statement

- ```
zone "domain_name" {  
    type  master;  // or slave  
    file  "path/to/zone/file";  
};
```
- Other possible types are `stub`, `forward`, and `hint`
  - Indicate whether your server is primary or secondary
  - Other options allowed here (e.g. zone transfers)
- Special zone type `hint`
  - All nameservers need to include a `hint` zone
  - Includes a seed file for the root nameservers

# BIND – zone Statement (cont)

- Most nameservers have a reverse zone for localhost
  - The 'localhost' name is dealt with in forward zones
  - The in-addr.arpa. address needs to be in a separate zone
  - ```
zone "0.0.127.in-addr.arpa" {  
    type master;  
    file "localhost";  
    notify no;  
};
```
  - Every nameserver is always the 'master' for the zone
  - Zone doesn't need to receive notifications

# BIND – logging Statement

- Here is the general syntax for a logging statement
- ```
logging {  
    <channel-definition-1>;  
    <channel-definition-2>;  
    ...  
    category <category-a> {  
        <channel-name-1>;  
        ...  
    };  
    category <category-b> {  
        <channel-name-2>;  
        ...  
    };  
};
```

# BIND – logging Statement (cont)

- Here is the syntax for a channel definition
- ```
channel <channel-name> {  
    file /file/path [versions N | unlimited] [size M];  
    syslog <facility>;  
    severity <severity>;  
    print-category yes; // or no  
    print-severity yes; // or no  
    print-time      yes; // or no  
};
```
- Use either the `file` clause OR the `syslog` + `severity` clauses

# BIND – logging Statement (cont)

- There are some default channels
  - default\_syslog - logs with facility 'daemon' and severity 'info' and worse
  - default\_debug - logs to a file called 'named.run'
  - There are some others
- The default logging statement for Bind 9 is
- ```
logging {  
    category default { default_syslog; default_debug; };  
};
```

# BIND – view Statement

- Views facilitate a concept known as 'Split DNS'
  - Provides different data for the same zone
    - Information returned depends on the query source
    - Traditionally used to hide some information about hosts from outside
    - Also useful for NATed networks
- If a `view` is used at all, views must be used completely
  - All `zone` statements must occur within a `view`
- Relies on ACLs to match IP addresses to serve data
  - uses the `match-clients` option

# BIND – view Statement Example

- ```
view "internal" {
    match-clients { "my_net"; };
    recursion yes;
    zone "xbalanque.net" {
        type master;
        file "xbalanque.forward";
    };
    zone "0.0.10.in-addr.arpa" {
        type master;
        file "xbalanque.reverse";
    };
    zone "." {
        type hint;
        file "cache.db";
    };
    zone "0.0.127.in-addr.arpa" {
        type master;
        file "localhost";
        notify no;
    };
};
```