



CSI3660 – System Administration

Prof. Fredericks

Network Service Configuration

Infrastructure Services

- Infrastructure services
 - Provide network configuration and support for other computers on a network
- Include:
 - DHCP
 - DNS
 - NTP
 - NIS
 - Apache / MySQL
 - SVN

DHCP

- Dynamic Host Configuration Protocol (DHCP)
 - Used for automatically configuring a network interface
- Client sends a DHCP broadcast on the network
 - Requesting IP configuration information
- DHCP server leases IP addresses to client computers for a period of time
 - Ensures each client has a unique IP address
 - After expiration, client must send another DHCP request

The DHCP Lease Process

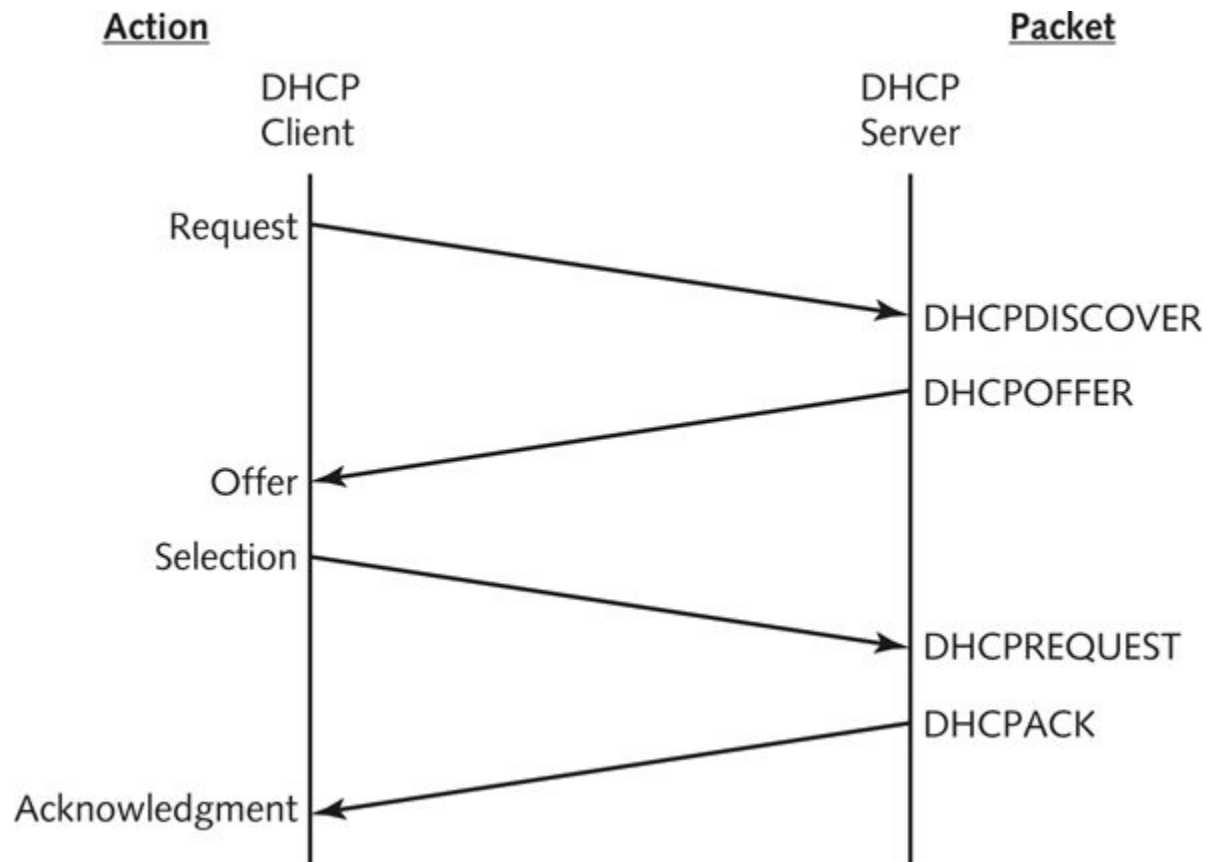


Figure 13-1: The DHCP lease process

The DHCP Lease Process

- Involves several stages:
 - Client sends a request to all hosts on network
 - In reply, a DHCP server sends an offer containing potential IP configuration
 - Client selects (accepts) offer
 - DHCP server sends acknowledgement indicating the amount of time client can use IP configuration
 - Client configures itself with IP configuration

The DHCP Lease Process

- To configure a Linux system as a DHCP server
 - Install the DHCP daemon (dhcp)
 - Available from online software repositories
- The two most common DHCP daemons:
 - DHCP
 - BusyBox DHCP (udhcp) → seems to have been deprecated and removed from yum repositories

Configuring a Linux DHCP Server using dhcpd

- Edit DHCP daemon configuration file to list appropriate IP address range for the network and lease information
 - /etc/dhcp/dhcpd.conf stores IPv4 configuration
 - /etc/dhcp/dhcpd6.conf stores IPv6 configuration
- Install and start dhcpd
 - Also configure it to start at boot time
 - **systemctl enable dhcpd**

DHCP Client

- dhclient command
 - -r : release
 - -v : verbose
- `$ sudo dhclient -r -v eth10`
- `$ sudo dhclient -v eth10`

DHCP

- Dynamically configure addresses for large groups of workstations
- Open protocol
 - Architecture of machine doesn't matter
- Why use DHCP?

NTP

- Network Time Protocol (NTP)
 - Used by the OS to obtain time information BIOS system clock or from network servers
 - `hwclock`
 - Modifies BIOS date and time
 - Uses UDP port 123
- Two NTP daemons commonly used on Linux systems:
 - NTP daemon (`ntp`)
 - Chrony NTP daemon (`chrony`)

Understanding NTP Strata

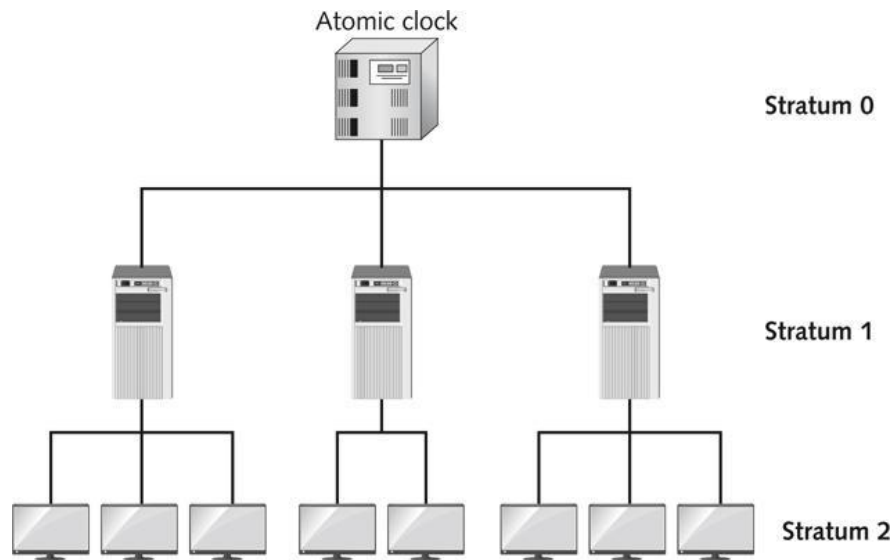


Figure 13-4 A sample strata structure

Understanding NTP Strata

- Strata:
 - Hierarchical series of time resources used by NTP
 - **Stratum 0**: Atomic clock or GPS clock
 - Reference clock
 - **Stratum 1**: Obtain time directly from stratum 0 device
 - Direct-connected to reference clock (not by network path)
 - **Stratum 2**: Obtain time directly from stratum 1 device
 - Connected to Stratum 1 via network
- Stratum is not an indication of quality or reliability
 - NTP servers obtain time information from multiple sources and use algorithm to determine most reliable time information

Working with ntpd

- NTP daemon (ntpd) usually installed by default
 - But not active
 - Can act as NTP client to obtain time from Internet time server or as NTP server
- To configure NTP client
 - Edit /etc/ntp.conf to add lines for different NTP servers that can be queried
- ntpdate
 - Manually synchronize the time
 - \$ sudo ntpdate 0.rhel.pool.ntp.org
 - Need to ensure that ntpd is not running
- Offset
 - Time difference between time on local computer and time on time server

Working with ntpd

- `ntpq`

- Used to see what actual time servers the system is synchronizing with

- Jitter buffer: stores the difference between the same time measurements from different NTP servers

- Jitter may appear to be random!
 - Used by NTP when determining the most reliable time
 - `ntpq -p` option: displays the offset and jitter

Working with ntpd

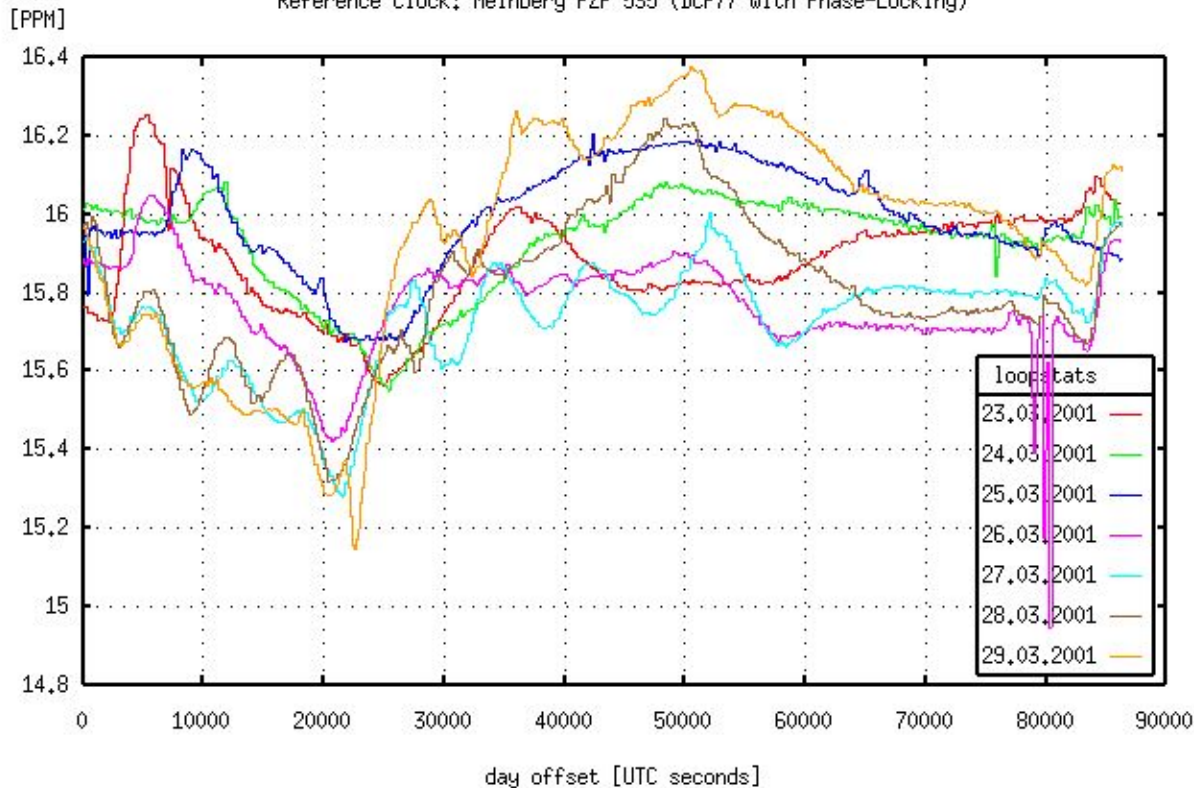
Example 3. Quartz Oscillators in IBM compatible PCs

In my experiments with PCs running [Linux](#) I found out that the frequency of the oscillator's correction value increases by about 11 PPM after powering up the system. This is quite likely due to the increase of temperature. A typical quartz is expected to drift about 1 PPM per °C.

Even for a system that has been running for several days in a non-air-conditioned office, the correction value changed by more than 1 PPM within a week (See [Figure 2](#) for a snapshot from that machine). It is possible that a change in supply voltage also changes the drift value of the quartz.

As a consequence, without continuous adjustments the clock must be expected to drift away at roughly one second per day in the worst case. Even worse, the values quoted above may increase significantly for other circuits, or even more for extreme environmental conditions.

PPS Frequency Changes within a Week
 Linux 2.2.18 with PPSkit-1.0.3 and ntpd-4.0.99k17 on a Pentium 100MHz
 Reference Clock: Meinberg PZF 535 (DCF77 with Phase-Locking)



the frequency of the
 or powering up the
 e. A typical quartz is

a non-air-conditioned
 within a week
 e that a change in

Exam
 In my
 oscilla
 system
 expect

Even f
 office.
 (See E
 supply

As a consequence, without continuous adjustments the clock must be expected to drift away at roughly one second per day in the worst case. Even worse, the values quoted above may increase significantly for other circuits, or even more for extreme environmental conditions.

Working with ntpd

- By default, the NTP daemon is not configured as an NTP server
 - Runs as local service for machine
- To allow other computers to query NTP daemon:
 - Edit `/etc/ntp.conf`
 - Add line identifying specific computers or networks that are allowed to query the NTP daemon
 - Restart NTP daemon for changes to take effect

Working with chronyd

- Some modern Linux distributions use chronyd in place of ntpd
 - Offers a faster response time and is backwards-compatible with ntpd
- Uses the `/etc/chrony.conf` file
 - Nearly identical to `/etc/ntp.conf`
- Use the `chronyc` command to provide functionality that the `ntpdate` and `ntpq` commands do with ntpd

chronyd vs ntpd

- **chrony** tends to work better when network access is bad
 - **ntp** requires constant polling of servers
- **chrony** handles sudden clock changes
 - Synchronizes faster
- **ntp** supports broadcast, multicast, and different types of authentication methods
- **chrony**: intermittent network access
- **ntp**: systems left on or required to broadcast/multicast

Network Information Service

- Network Information Service (NIS)
 - Coordinate common configuration files across workstations
 - Participating computers belong to an NIS domain
 - Use NIS map to access configuration information
 - Eg., usernames, hostnames, etc
- NIS master server
 - Sends all NIS map configuration to NIS slave servers
- NIS slave servers
 - Distribute maps to NIS clients

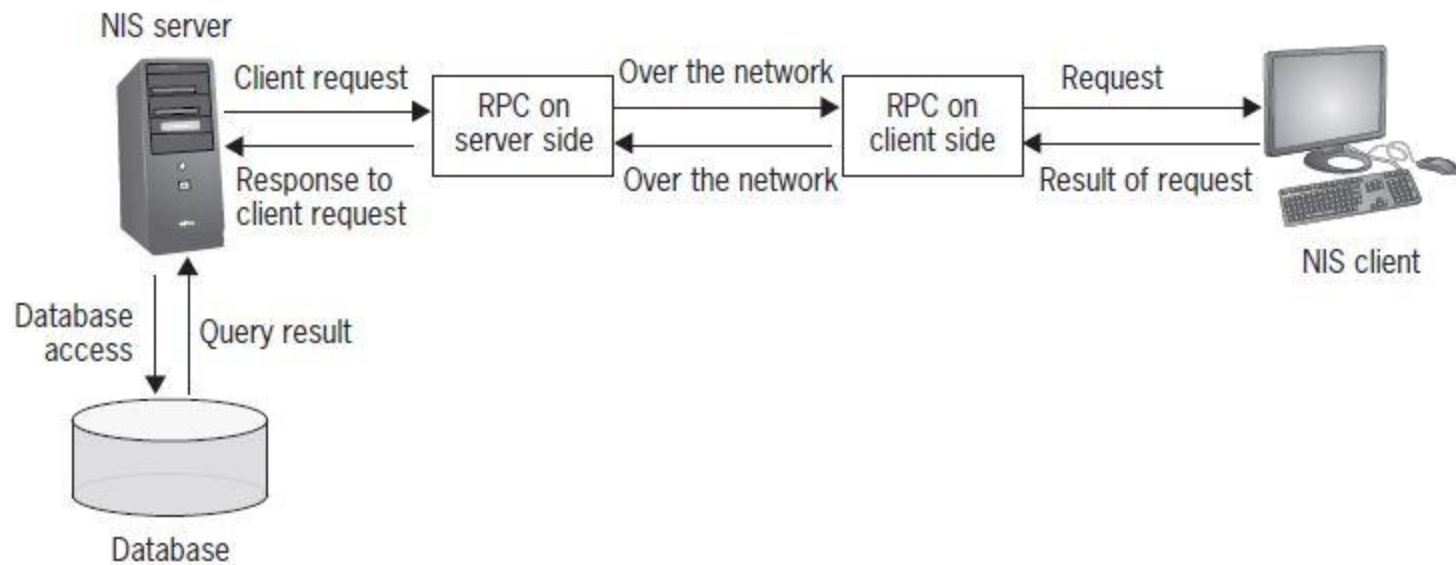


Figure 17-1 NIS communication with RPC
© Cengage Learning 2013

NIS Server Configuration

- Install ypserv and rpcbind packages from a software repository
 - rpcbind already installed on Scientific
 - Many commands start with 'YP'
 - Yellow pages → NIS was originally called YP but the term was trademarked
- Define the NIS domain name via `ypdomainname`
`NIS_domain_name` command
- In the `/var/yp/Makefile` file, edit the list of files to be made into maps
 - If no slave servers, ensure `NOPUSH=true`
- Add identification of allowed clients to `/var/yp/securenets` file

NIS Server Configuration

- Allow the allowed clients to access the appropriate maps in `/etc/ypserv.conf`
- Start the `ypserv` and `rpcbind` daemons and set them to start at boot time
- Generate the configuration file maps by typing `/usr/lib/yp/ypinit -m` at a command prompt

NIS Client Configuration

- Install the ypbind and rpcbind packages via a software repository
- Define the NIS domain name via `ypdomainname NIS_domain_name` command
- Edit the `/etc/sysconfig/authconfig` file to read `USENIS=yes`
- In the `/etc/yp.conf` file, add `domain NIS_domain server NIS_server`
 - Alternatively add `domain NIS_domain broadcast`

NIS Client Configuration

- Start the ypbind and rpcbind daemons
 - Set them to start at boot time (chkconfig)
- Locate NIS server by typing `ypwhich` command
- Edit the `/etc/nsswitch.conf` file and add the keyword `nis` before other methods for user, group, and host name lookup
- Ensure all users on NIS clients use the `yppasswd` command to change NIS password

Web Services

- Apache is the most common Web server
 - Started as the HTTP daemon
- Document root directory
 - /var/www/html
 - Local websites
 - Default document is index.html (or index.php if PHP is active...)
- /etc/httpd/conf/httpd.conf
 - Default configuration file
- Directive
 - Each line within a configuration file

Web Services

```
sudo yum install httpd
```

```
sudo firewall-cmd --permanent --add-port=80/tcp
```

```
sudo firewall-cmd --permanent --add-port=443/tcp
```

```
sudo firewall-cmd --reload
```

```
sudo systemctl start httpd
```

```
sudo systemctl enable httpd
```

Web Services

Directive	What It Specifies
<code>Listen 80</code>	Apache daemon will listen for HTTP requests on port 80.
<code>ServerName server1.class.com</code>	Name of the local server is server1.class.com.
<code>DocumentRoot "/var/www/html"</code>	Document root directory is /var/www/html on the local computer.
<code>DirectoryIndex index.html</code>	Index.html file in the document root directory will be sent to clients who request an HTML document.
<code>ServerRoot /etc/httpd</code>	All paths listed within the /etc/httpd/conf/httpd.conf file are relative to the /etc/httpd directory.
<code>ErrorLog logs/error_log</code>	All Apache daemon messages will be written to the /etc/httpd/logs/error_log file.
<code>CustomLog logs/access_log combined</code>	All Web requests will be written to the /etc/httpd/logs/access_log file using a combined log format.
<code>MaxClients 150</code>	Maximum number of simultaneous requests cannot exceed 150.
<code>User apache</code>	Apache daemon will run as the "apache" local user account.
<code>Group apache</code>	Apache daemon will run as the "apache" local group account.
<code><Directory /var/www/html> Order allow,deny Allow from all Deny from 192.168.1.51 </Directory></code>	All hosts are allowed to access HTML files and other Web content from the /var/www/html directory except for the computer with the IP address 192.168.1.51.

Table 13-2 Common httpd.conf directives

Web Services

- Default settings are sufficient for most Web servers
 - Copy appropriate HTML files into `/var/www/html`
 - Start Apache
 - `service httpd start`
- A separate `httpd` daemon is started each time a client request is received by Apache Web server
 - Called a Web page hit
 - First daemon started as root user
 - Others started as Apache user

Web Services

- curl

- Transfer data to/from a server using a protocol

- HTTP/S, FTP/S, etc.

- Defaults to HTTP if it can't guess protocol

- curl www.google.com

- curl www.google.com -o GOOG

- curl www.google.com > GOOG_REDIRECT

- Can get multiple files at once

- curl www.google.com -o GOOG www.yahoo.com -o YAHOO

curl

- Other options

- Get/put files to FTP servers

- (Upload) `curl -T myfile.zip ftp://example.com/mydirectory/ --user username:password`

- Show HTTP response headers

- `curl -I www.google.com`

- Send cookies to server

- `curl -b mycookies.txt http://example.com`

- Send HTTP POST requests

- `curl -X POST -u admin:admin http://example.com/myconfigs/status -Hcontent-type:application/xml -d @/home/user/file.xml`

- Send emails

- `curl --url "smtps://smtp.example.com:465" --ssl-reqd --mail-from "user@example.com" --mail-rcpt "friend@example.com" --upload-file mailcontent.txt --user "user@example.com:password" --insecure`

wget

- Non-interactive network download
 - wget <http://www.openss7.org/repos/tarballs/strx25-0.9.2.1.tar.bz2>
 - (part of a VOIP package, not important for this class)
- Rename file as you download
 - wget http://www.vim.org/scripts/download_script.php?src_id=7701
 - wget -O easiername.zip http://www.vim.org/scripts/download_script.php?src_id=7701
- Continue incomplete downloads with -c flag
- Mirror website with --mirror
- Download files of type
 - wget -r -A.pdf <website> (r and A are both flags)

Fun Script

- Generate a text message to get you “out of something” at a particular time
- (Ch 26)
- Using textbelt.com/text → RESTful API for texting
 - ~~■ 75 texts per day from IP address~~
 - Use curl to send message

```
curl -X POST https://textbelt.com/text --data-urlencode  
phone='xxxXXXxxxx' --data-urlencode message='Hello world'  
-d key=textbelt
```

Fun Script

■ Syntax:

```
$ curl -s http://textbelt.com/text -d  
number=YOUR_PHONE_NUMBER -d "message=YOUR MESSAGE"
```

Need to setup Github project now: (went paid)

<https://github.com/typpo/textbelt>

(Port blocked by school though)

Fun Script

■ Scriptified

```
$ cat saveme.sh
#!/bin/bash
phone="YOUR_NUMBER_WITH_AREA_CODE"
SMS_URL=http://textbelt.com/text"
message="You need to get home right now"

curl -s $SMS_URL -d number=$phone -d
"message=$message" > /dev/null
```

And of course...

- Set it to run in the future
 - Maybe 5 minutes into a meeting you don't want to attend?

```
$ at -f saveme.sh now + 15 minutes
```

- Could pass it an argument to send a particular message
 - 1) Emergency
 - 2) Need to call back a friend
 - 3) ...

/etc/hosts

- Small sites with no internet connectivity
 - Route to local sites
 - Can do “interesting” redirection

<REDIRECT - IP ADDRESS> <URL OF SERVER TO TRAP>

FTP

- Protocol most commonly used to transfer files on public networks
- In anonymous access, a special directory is available to any user who wants to connect to the FTP server
- User can log in, via an FTP client program, to a home directory on the FTP server

Connecting to a Linux FTP Server

- Most Web browsers have built-in FTP utility
 - Allows you to access files on remote computer
- To connect through Web browser, specify the location by typing *ftp://servername* in the browser
 - To log in as particular user, type *ftp://user:password@servername*
- Most OSs have command-line FTP utility
 - Use `ftp` command and specify host name as argument, log in as anonymous or as specific user
 - Receive prompt that accepts FTP commands

Configuring a Linux FTP Server

- Very Secure FTP daemon (vsftpd)

- Used by most Linux systems

```
$ sudo yum install vsftpd
```

- Configure appropriately, open up relevant ports, start daemon

- Port: `sudo firewall-cmd --permanent --add-port=21/tcp`
`sudo firewall-cmd --reload`

- Start the vsftpd daemon and ensure that it is started at boot time

vsftpd

- Login as root (su)

- Install vsftpd

```
$ yum install vsftpd ftp -y
```

- Configure vsftpd (/etc/vsftpd/vsftpd.conf)

- Set `anonymous_enable=NO`

- Set `chroot_local_user=YES`

- Add `use_localtime=YES` to end of file

- Start service

```
$ sudo systemctl start vsftpd
```

```
$ sudo systemctl enable vsftpd
```

vsftpd

■ Configure firewall

```
sudo firewall-cmd --add-port=21/tcp --permanent  
sudo firewall-cmd --add-service=ftp --permanent  
sudo firewall-cmd --reload
```

- Update SELinux

```
sudo setsebool -P ftp_home_dir on
```

vsftpd

- Create an FTP user

```
$ useradd ftpuser
```

```
$ passwd ftpuser
```

- Connect using ftp user:

```
$ ftp <IP address>
```

- Uh oh!

- SELinux (Security Enhanced Linux) enabled by default

- Additional security for systems --- messes with services

```
$ setsebool -P ftp_home_dir on
```

Connecting to a Linux FTP Server

Command	Description
Help	Displays a list of commands
Pwd	Displays the current directory on the remote computer
dir ls	Displays a directory listing from the remote computer
cd <i>directory</i>	Changes the current directory to <i>directory</i> on the remote computer
lcd <i>directory</i>	Changes the current directory to <i>directory</i> on the local computer
get <i>filename</i>	Downloads the <i>filename</i> to the current directory on the local computer
ascii	Specifies text file downloads (default)
binary	Specifies binary file downloads
mget <i>filename</i>	Downloads the file named <i>filename</i> to the current directory on the local computer; it also allows the use of wildcard metacharacters to specify the <i>filename</i>

Table 13-3: Common FTP commands

Connecting to a Linux FTP Server

Command	Description
<code>put filename</code>	Uploads the <i>filename</i> from the current directory on the local computer to the current directory on the remote computer
<code>mput filename</code>	Uploads the <i>filename</i> from the current directory on the local computer to the current directory on the remote computer; it also allows the use of wildcard metacharacters to specify the <i>filename</i>
<code>!</code>	Runs a shell on the local computer
<code>close</code>	Closes the FTP connection to the remote computer
<code>open hostname or IP</code>	Opens an FTP connection to the <i>hostname or IP</i> address specified
<code>bye</code> <code>quit</code>	Quits the FTP utility

Table 13-3 (cont'd): Common FTP commands

File Sharing Services

- There are many file sharing services available on Linux systems
- The most common include:
 - Samba
 - NFS
 - FTP

Samba

- Samba daemon
 - Emulates SMB protocol
 - Formats TCP/IP data like Windows computers
- NetBIOS name daemon
 - Create and advertise NetBIOS name for Windows computers to connect to Linux server

Configuring a Samba Server

- `$ yum install samba`
- Create a local Linux user for each Windows user
 - `smbpasswd` command: Generate Samba passwords
- `/etc/samba/smb.conf`
 - Default Samba configuration file
 - Edit to include NetBIOS name
 - `testparm` command
 - Checks syntax of `/etc/samba/smb.conf`
- If you change the `smb.conf` file, you must restart the Samba and NetBIOS name daemons

Create Directory

```
$ sudo mkdir /home/share
```

```
# Update for SELinux
```

```
$ sudo chcon -t samba_share_t /home/share
```

```
# Create Samba password
```

```
$ sudo smbpasswd -a <username>
```

```
# Start smb and nmb services
```

```
$ sudo service start smb
```

```
$ sudo service start nmb
```

```
# Firewall
```

```
$ sudo firewall-cmd --permanent --add-port=445/tcp  
&& sudo firewall-cmd --reload
```

Connecting to a Samba Server

- Test Samba functionality after configuration
 - From Windows client enter `\\Samba_server` in the Run dialog box
 - *Samba_server* is the NetBIOS name or IP address of the Samba server
- `smbclient` command
 - Used to connect a Linux computer to a Samba server
 - Can also be used to display an FTP-like interface on Samba or Windows servers
- `$ smbclient \\\localhost\\share`

Windows Side

- Map Network Drive (right click on My Computer / This PC)
 - `\\<your ip>\share`
 - Enter Samba credentials

Connecting to a Samba Server

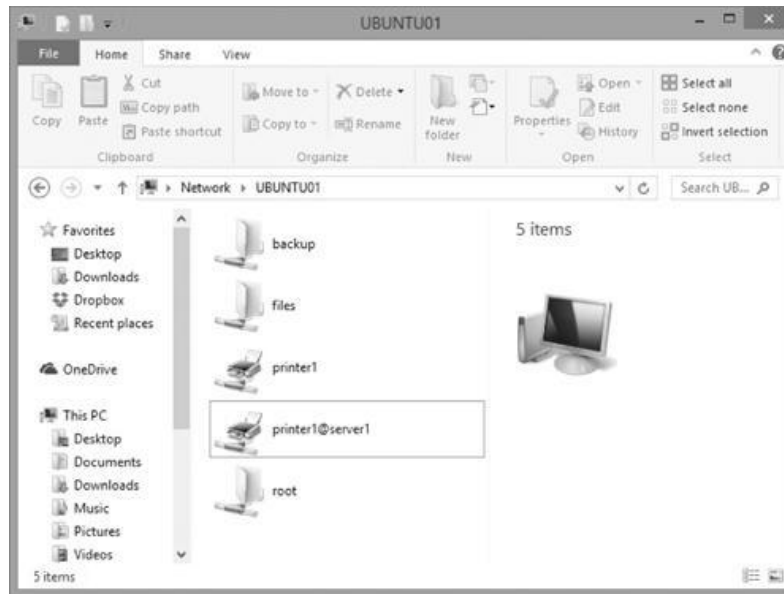


Figure 13-6 Accessing a Samba server from a Windows client

NFS

- Network File System (NFS)
 - Allows Unix, Linux, and Macintosh OS X computers to share files transparently
- One computer shares (or exports) a directory by placing its name in the `/etc/exports` file
- `mount` command:
 - Used by another computer to access an exported directory across the network by mounting the remote directory on the local computer

Update firewall:

```
$ sudo firewall-cmd --permanent --add-service=nfs
```

```
$ sudo firewall-cmd --reload
```

Configuring a Linux NFS Server

- Install and start the NFS server package (nfs-utils, nfs-utils-lib)
- Create a directory containing information to share
- Edit /etc/exports file:
 - Add a line listing the directory to be shared/options
 - Save your changes to the /etc/exports file
- Run `exportfs -a` command
 - Update list of exported filesystems
- Restart the **nfs** daemon and **nfslock** daemons
- Make local directory to mount share on
- Mount with:
 - `$ mount <IP>:<server location> <local location>`

Connecting to a Linux NFS Server

- To access files using NFS, mount a directory from a remote NFS server to a directory on a local computer
 - Use `mount` command specifying `nfs` filesystem type, server name or IP address, remote directory, and local directory as arguments
- Use `umount` command to dismount remote directory

Short Interlude (Case Studies)

- In small groups, solve the following challenges (turn in after class)

Case Study 1

In 2015 a client was parking an Email server in a shared collocation space at a nearby data center. The data center did not provide any type of network security, including any firewalls. Our client had installed an out-of-the-box copy of Linux that wasn't properly secured. A month after installation, the Linux server became vulnerable to attacks whereas the server would continue to crash.

How could this be resolved?

Case Study 2

In 2015 a well known business had built an in-house email system. The system ran a variant of Microsoft Windows and used the ISS software suite for Exchange Server. The system had been performing well until the system suffered a hard power shutdown as a result of a power outage.

The system was not equipped with a UPS supply backup. When the server was turned [back] on, although the operating system was functional, the email software wasn't. As it happens, the NTFS based Windows filesystem doesn't handle hard stops very well and become corrupt.

Case Study 3

In 2014 an organization came calling that had an impending security breach on their hands, namely; The organization's Web Server had been hacked and most, if not all the files had become corrupt or were missing. The Web Server was running a variant of Linux with an unpatched, insecure Apache Server.

The organization was a law firm and it was imperative that they have the problem solved as quickly as possible. The server had been installed by somebody working from a manual, instead of somebody who had worked in the industry. Most components of the server, as well as the network, were found to be extremely insecure, including the WiFi router that connects to the members wireless devices. All devices were on the same network.

E-mail Services

- Various e-mail protocols exist, including SMTP, ESMTP, POP, and IMAP
- An e-mail server looks up the name of target e-mail server in domain's MX records, stored on public DNS server
 - Resolves target e-mail server name to IP address using public DNS server
- Daemons and system components rely on e-mail to send important information to the root user

E-mail Services

- Most common e-mail daemon used on modern Linux systems is Postfix
 - Configured by default to accept e-mail on TCP port 25 and route to appropriate user
 - To check e-mail, use the `mail` command
- The `/etc/aliases` file contains other e-mail names used to identify different users on the system
 - If that file is edited, run the `newaliases` command in order to rebuild the aliases database

E-mail Services

Line	Change
<code>mydomain = sample.com</code>	Sets the e-mail domain name; changes to desired name
<code>myorigin = \$mydomain</code>	Sets local access to the domain name
<code>inet_interfaces = all</code>	Configures Postfix to listen for e-mail on all interfaces
<code>mydestination = \$myhostname, localhost.\$mydomain, localhost, \$mydomain</code>	Configures destination domain for e-mail
<code>mynetworks_style=class</code>	Trusts e-mail from computers on the local network

Table 13-4: Sample lines in `/etc/postfix/main.cf` to modify or add when configuring Postfix

DNS

- DNS
 - Hierarchical namespace used to identify computers on large TCP/IP networks
- Zone
 - Portion of DNS administered by one or more DNS servers
- Forward lookup
 - FQDN resolved to IP address
- Reverse lookup
 - IP address resolved to FQDN

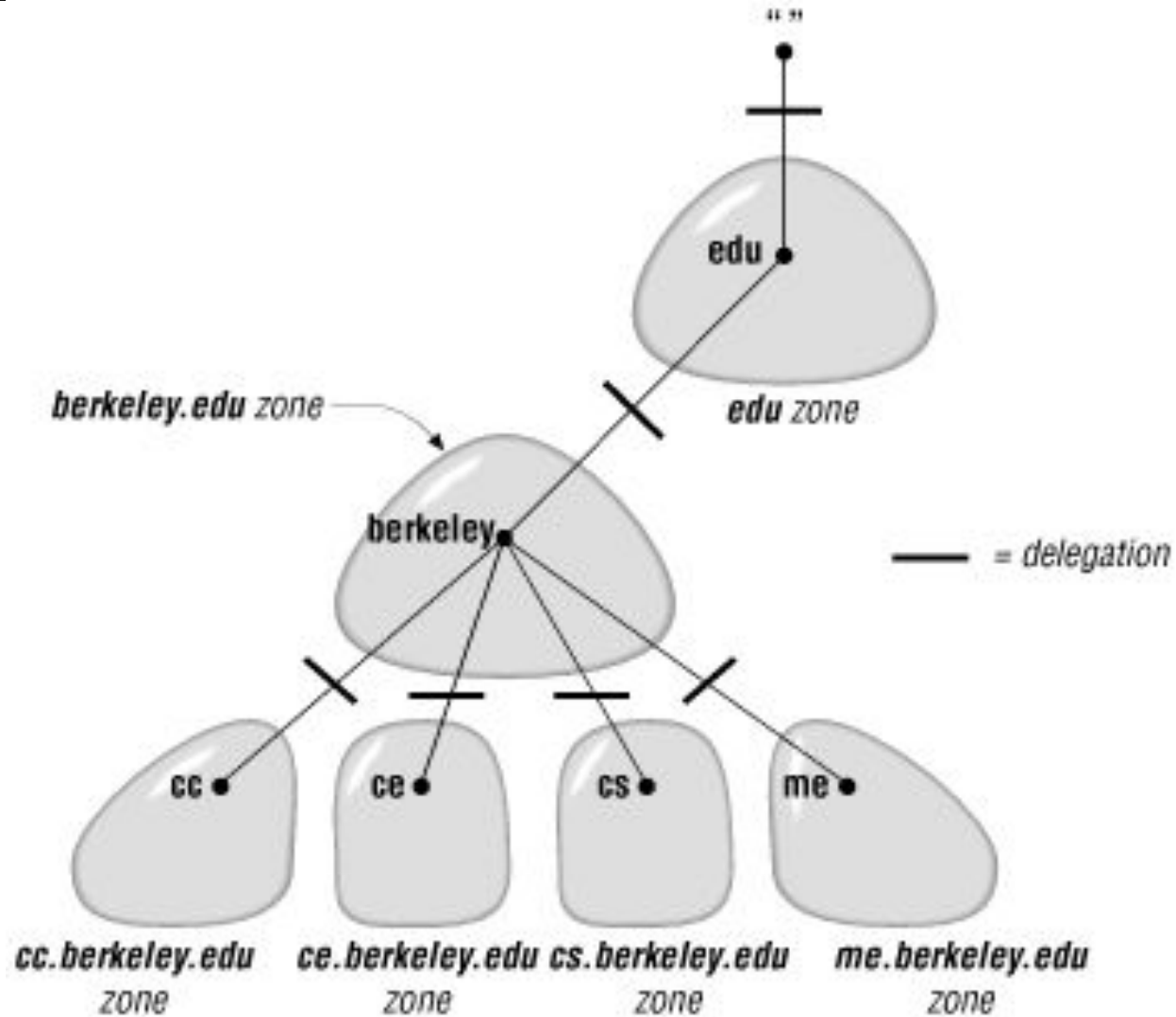
The DNS Lookup Process

- When contacting a Web server using a web browser:
 - Web browser performs a forward lookup of FQDN to contact the IP address of the Web server
 - Forward lookup is performed by DNS servers
- Iterative query
 - Resolved using DNS cache
 - Does not use top-level DNS servers
- Recursive query
 - Resolved with the use of top-level DNS servers

Configuring a Linux DNS Server

- Configure DNS name daemon for a specific zone
 - Add resource records that list FDQNs and associated IP addresses for computers in that zone
- Configuration files have BIND format
 - Difficult to create manually
 - Use graphical utility such as BIND configuration utility
- Start the DNS name daemon
- `dig` command
 - Used to query records that exist on a specific DNS server
 - `dig www.google.com`

DNS Zones



dig

dig hostname

dig domain-name

dig @DNS-server

Type	Purpose
A	IPv4 IP address
AAAA	IPv6 IP address
CNAME	Canonical name record (Alias)
MX	Email server host names
NS	Name (DNS) server names
PTR	Pointer to a canonical name. Mostly used for implementing reverse DNS lookups
SOA	Authoritative information about a DNS zone
TXT	Text record

Configuring a Linux DNS Server

File	Description
/etc/named.conf	Contains the list of DNS zones and their type (master/slave) that the name daemon will manage.
/var/named/zone_name.db or /var/named/zone_name.zone	Contains resource records used to perform forward lookups for a particular <i>zone_name</i> . Lines in this file have a type that determines the kind of resource record: <ul style="list-style-type: none">• A (add host) records map FQDNs to IPv4 addresses.• AAAA (add host) records map FQDNs to IPv6 addresses.• CNAME (canonical name) records provide additional aliases for A records.• NS (name server) records provide the names of DNS servers for the zone.• MX (mail exchange) records provide the IP address for the e-mail server for a zone.• SOA (start of authority) determines the parameters used for zone transfers as well as how long information can be cached by the computer performing the forward or reverse lookup (called the Time-To-Live, or TTL).

Table 13-1: Common zone configuration files

Configuring a Linux DNS Server

File	Description
<code>/var/named/reverse_network_ID.in-addr.arpa</code> or <code>/var/named/network_ID.db</code> or <code>/var/named/network_ID.zone</code>	Contains resource records of type PTR (pointer), which list names used for reverse lookups for a particular network. The network is incorporated into the filename itself; for example, the filename that contains PTR records for the 192.168.1.0 IPv4 network would be called 1.168.192.in-addr.arpa or 192.168.1.db or 192.168.1.zone.
<code>/var/named/named.local</code> & <code>/var/named/named.ip6.local</code> or <code>/var/named/named.localhost</code> or <code>/var/named/named.loopback</code>	Contains PTR records used to identify the loopback adapter (127.0.0.1 for IPv4, ::1 for IPv6).
<code>/var/named/named.ca</code> or <code>/var/named/named.root</code>	Contains the IP addresses of top-level DNS servers; it is commonly called the DNS cache file.

Table 13-1 (cont'd): Common zone configuration files

Database Services

- Databases
 - Large files that store information in the form of tables
- Table
 - Organizes information into a list
- Record
 - Set of information about a particular item within a list
- Fields
 - Categories of information within a record
- Relational databases
 - Databases in which information within one table is related to information within other tables
 - Tables are usually linked by a common field

Database Services

EmployeeID	FirstName	LastName	Address	ZIP	Home Phone	Mobile Phone
A518	Bob	Smith	14 Wallington St.	49288	555-123-1399	555-144-2039
A827	Jill	Sagan	51 York Ave. N.	49282	555-132-1039	
A988	Frank	Kertz	623 Queen St.	44922	555-209-1039	555-199-2938
A472	Bethany	Weber	82 Shepherd Ave.	49100	555-299-0199	555-203-1000
A381	John	Lauer	55 Rooshill Ave.	49288	555-123-2883	555-203-2811

Common field

EmployeeID	Week	Hours	MgrOK
A518	08/20/10	37	Yes
A827	08/20/10	40	Yes
A988	08/20/10	40	Yes
A472	08/20/10	40	Yes
A381	08/20/10	22	Yes
A518	08/27/10	40	Yes
A827	08/27/10	35	Yes
A988	08/27/10	0	Yes
A472	08/27/10	40	Yes
A381	08/27/10	22	Yes

Field names

Record

Figure 13-8: A simple relational database structure

Database Services

- Structured Query Language (SQL)
 - Programming language used to store and access data in databases
 - SQL servers
 - Offer advanced backup, repair, replication, and recovery utilities for data
 - Allow programs to access databases from across the network
- `SELECT * FROM TABLE WHERE id = <ID>`

PostgreSQL and MySQL

- Two common SQL database approaches
 - Again, near-religious argument for using either
 - Both open source systems with thriving communities
 - Comparisons on a year-by-year basis regarding performance of each
- (I've always used MySQL for web projects)

Configuring PostgreSQL

- PostgreSQL is a powerful SQL server that provides a large number of features
 - Can be installed via software repositories
- Prepare for use by:
 - Use `passwd postgres` command to assign user a password
 - Modify the PostgreSQL configuration file, `postgresql.conf` (location differs depending on Linux distribution)
 - Start the PostgreSQL engine

Configuring PostgreSQL Databases

- Log in as postgres user
- Execute PostgreSQL command-line utilities to create and manage databases
 - Can create tables and add records within the PostgreSQL utility using appropriate SQL statements
 - PostgreSQL utility has many built-in commands
 - Prefixed with a \ character
 - Can be used to obtain database information or perform functions within the utility

Configuring PostgreSQL Databases

Command	Description
clusterdb	Associates a PostgreSQL database to another database on a different server
createdb	Creates a PostgreSQL database
createlang	Allows a new programming language to be used with PostgreSQL
createuser	Creates a PostgreSQL user
dropdb	Deletes a PostgreSQL database
droplang	Removes support for a programming language within PostgreSQL
dropuser	Deletes a PostgreSQL user
pg_dump	Backs up PostgreSQL database settings
pg_dumpall	Backs up a PostgreSQL database cluster settings
pg_restore	Restores PostgreSQL database settings
psql	The PostgreSQL utility
reindexdb	Reindexes a PostgreSQL database
vacuumdb	Analyzes and regenerates internal PostgreSQL database statistics

Table 13-6: PostgreSQL command-line utilities

Configuring PostgreSQL Databases

Command	Description
<code>\l</code>	Lists available databases
<code>\c <i>database_name</i></code>	Connects to a different database
<code>\d</code>	Lists the tables within the current database
<code>\d <i>table_name</i></code>	Lists the fields within a table
<code>\q</code>	Exits the PostgreSQL utility

Table 13-7: Common built-in PostgreSQL utility commands

MySQL

■ Install MySQL service

```
$ yum install mysql-server # already installed  
$ service mysqld start
```

■ Autoconfigure MySQL

```
$ mysql_secure_installation  
Setup a root password, answer yes to all questions
```

■ Open up ports for remote access (add to iptables)

- `-I INPUT -p tcp --dport 3306 -m state --state NEW,ESTABLISHED -j ACCEPT`
- `-I OUTPUT -p tcp --sport 3306 -m state --state ESTABLISHED -j ACCEPT`
- OR
- `firewall-cmd --permanent --add-service=mysql`

MySQL

- Command line

```
$ mysql -u root -p
```

- Show MySQL users

```
mysql> select user,host,password from mysql.user;
```

- Show databases

```
mysql> show databases;
```


MySQL

■ Add database

```
mysql> create database <dbname>;
```

```
mysql> use <dbname>; # select database
```

■ Add table

```
mysql> CREATE TABLE cit348_tbl(  
    id INT NOT NULL AUTO_INCREMENT,  
    name VARCHAR(50),  
    PRIMARY KEY (id)  
    );
```

MySQL

■ Add data

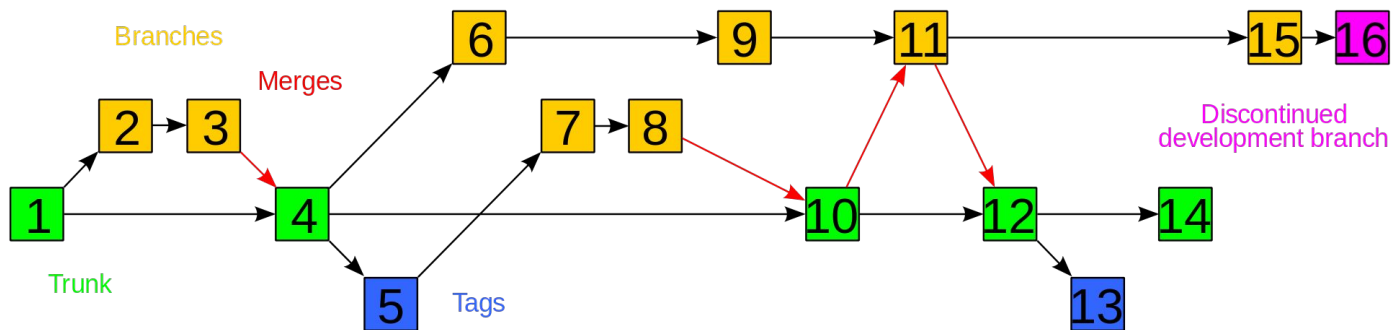
```
mysql> INSERT INTO cit348_tbl (name) VALUES ("student1");
```

■ Retrieve data

```
mysql> SELECT * FROM cit348_tbl
```

SVN

- Subversion
 - Version and revision control software



```
$ yum install mod_dav_svn subversion
```

```
$ mkdir /var/www/svn
```

Configure Web Access

```
$ vim /etc/httpd/conf.d/subversion.conf  
  
## Uncomment <Location /repos> block  
  
## Change AuthName to "Subversion Repos"  
  
## Change AuthUserFile to "/etc/svn-auth-conf"
```

Configure SVN Password File

```
## Grant user access
```

```
$ htpasswd -cm /etc/svn-auth-conf <username1>
```

```
$ htpasswd -cm /etc/svn-auth-conf <username2>
```

```
...
```

Configure Repository

```
$ cd /var/www/svn
```

```
## Create the repository
```

```
$ sudo svnadmin create temp_proj
```

Import Project

```
## Create a dummy project
```

```
$ mkdir /tmp/temp_proj
```

```
$ cd /tmp/temp_proj
```

```
$ touch file1 file2 file3
```

```
## Import files into SVN
```

```
$ sudo svn import /tmp/temp_proj
```

```
file:///var/www/svn/temp_proj -m "Initial commit"
```

Check Out

- Pull last working copies

```
$ cd ~
```

```
$ svn co file:///var/www/svn/temp_proj/
```


Check In (Commit)

- Creates new version of files

- (Change file1)

```
$ svn commit -m "Updated file1"
```

Group Project

- Given that there is 1 month left, SVN / Git is **not** a required service
 - However, it is an opportunity for extra credit
 - To qualify:
 - All project documents (project summary, status reports, final writeup, presentation) must all be under version control
 - Configuration files that you have changed must also be managed
- Also, email service (as mentioned in the syllabus) also not required since we won't be setting up DNS servers to manage it