Daemons: Printing

☞ Printing

➤ **lpd** – the "standard" BSD print spooling daemon.

➤ Accepts jobs, places them in a spool

Daemons: Printing

➠ If lpd is operating locally, then it does the interaction with printer (these days, almost always via a filter that does the actual communication)

➠ If not local, then (unsurprisingly) the daemon sends the job to another machine; the **lpd** protocol (RFC 1179, see http://www.ietf.org/rfc/rfc1179.txt) was not a great design success

Daemons: Printing

⇛ **lpsched** − the "standard" ATT version of **lpd** ; it is more complex to administer (see Chapter 23 of USAH) and, while it was less likely to wander off the reservation once it is in operation, configuration can be much more interesting and problem-laden than lpd. **lpsched** uses the same RFC 1179 protocol, which it inherited from BSD.

Daemons: Printing

⇛ **lprng** – an open source **lpd** replacement, includes a "Printing Cookbook" for people who like details. However, **lprng** doesn't seem to be gaining popularity; **cups** instead seems to have overtaken it

Daemons: Printing

⇶→ **cups** – a very popular open source replacement which disposes of the problematic RFC 1179 protocol, replacing it with IPP (RFC2567 (good explanation of the overall view of the protocol's design), RFC2568, RFC2569, RFC2639, RFC2910, RFC2911, RFC3196, RFC3239, RFC3380, RFC3381, RFC3382, RFC3391, RFC3510, RFC3712, RFC3995, RFC3996, RFC3997, RFC3998)

Daemons: Printing

⇛ Windows 2003/2008 – See Chapter 13 (page 1059) of W2K3 on how to configure and troubleshoot network print services. In particular, there is a nice 7 step summary on page 1064 of the printing process.

Daemons: MTAs/MSAs

☞ Mail Transfer Agents (MTAs, see for instance RFC2821) and Mail Submission Agents (MSAs, RFC2476)

☞ **sendmail**

☞ Routes local and network mail. Acts as MTA (and as an MSA listening on port 587), **sendmail** is one of the Internet email backbone workhorse programs.

Daemons: MTAs/MSAs

☞ One of the largest and historically "buggiest" daemons, although the latest versions have security patches aggressively developed as needed (it is always a good idea to check http://www.sendmail.org for the latest on **sendmail** security.)

☞ Configuration information is kept these days in the subdirectory `/etc/mail`.

Daemons: MTAs/MSAs

☞ The file `/etc/mail/sendmail.cf` is a set of rewriting rules for modifying addresses; luckily tools exist to automate creation of this file (basically, you use a "makefile" that rewrites a ".mc" file into a ".cf" file. Check http://www.sendmail.org/ for lots more information – the `op.ps` manual is the canonical reading material, although the O'Reilly book is easier.)

Daemons: MTAs/MSAs

☞ **sendmail** is covered some in Chapter 19 of USAH, plus there is an entire O'Reilly & Associates book is dedicated to **sendmail** .

Daemons: MTAs/MSAs

☞ Current, **sendmail** 8.14 is quite popular as an MTA. The ability to use a bolt-on "milter" (mail filter) was added (see http://www.milter.org), and now **sendmail** is probably the most flexible MTA when dealing with working at a message level; milters can detect and reject spam, they can check for legitimate users even for just forwarding MTAs, they can be implemented in C, C++, Perl, and Python.

Daemons: MTAs/MSAs

☞ **postfix**

☞ **postfix** comes from IBM, and has become probably the second most popular MTA. (http://www.postfix.org)

☞ It is very powerful: and **postfix** now does handle milters. It does have a large set of configuration files that work very well together.

☞ The configuration is typically in /etc/postfix.

Daemons: MTAs/MSAs

☞ **qmail** – Dan Bernstein's MTA (http://www.qmail.org).

☞ **smail** – an older, less popular MTA from GNU

☞ **exim** – an MTA from Cambridge, gaining in popularity, now found in many Linux distributions such as RedHat (CentOS) and Debian (where it is now the default MTA)

☞ **Exchange** – the enterprise Windows email server from Microsoft

Daemons: MTAs/MSAs

☞ SA relevance:

☞ Mail service is the most popular and, arguably, most important service on your system (along with web service)

☞ Users get very upset when mail does not work exactly right

☞ As with any other network service, you must keep up with the latest security patches

Daemons: MTAs/MSAs

☞ Configuring and tuning **sendmail** can take a lot of SA time, although generally the new m4 files (and the development of supporting GUI software) permit rapid deployment with just a little bit of effort (look at webmin at http://www.webmin.com, for instance)

Daemons: MTAs/MSAs

☞ Very important these days in both server and client support is anti-spam and anti-virus protection. The biggest server tools are

☞ MailScanner (http://www.mailscanner.info) (server only)

☞ clamav (http://www.clamav.net) (runs on both client and server)

☞ razor (http://razor.sourceforge.net) (generally invoked within SpamAssassin)

☞ dcc (http://www.dcc-servers.net/dcc) (generally invoked within SpamAssassin)

☞ SpamAssassin (http://spamassassin.apache.org/) (general run on servers)

☞ SpamHaus (or other RBL source) (http://www.spamhaus.o (best used at the MTA-level, although SpamAssassin can consult RBLs)

NFS – Network File Service

☞ NFS was developed by Sun and is now used by many UNIX/Linux systems

☞ It allows file access across the network as if the files were local

NFS - Network File Service

☞ NFS exists as a number of daemons - nfsd, biod, etc., as well as in kernel file system code

☞ NFS is covered in Ch. 17 of USAH and we will cover it in more detail in a later lecture

Yellow pages (NIS and NIS+)

☞ Allows key system files ("maps") to be shared over the net using a UNIX/Linux dbm database and a client/server model running on top of RPC.

1. ``ypcat passwd | more'' *vs*
2. ``more /etc/passwd''
3. /var/yp on the YP server and clients
4. YP == NIS (Network Information Service)

Yellow pages (NIS and NIS+)

☞ ypserv - server daemon

1. One master (see via "ypwhich")
2. Serves a YP domain — "csdept" via "domainname"

Yellow pages (NIS and NIS+)

3. NOTE: YP domain name != DNS domain name != Windows domain (The term "domain" is, unfortunately, overloaded and overused in the computing field.)

☞ ypbind – client daemon: Locates a yp server and serves up the maps

Yellow pages (NIS and NIS+)

☞ SA RELEVANCE: You may come across the use of NIS or NIS+ in future jobs. The idea is a good one, but the implementation is now somewhat dated and insecure. Systems are moving away from NIS/NIS+ and into more versatile directory services, such as the Lightweight Directory Access Protocol (LDAP). More on generalized directory services and LDAP later. Pp. 521-531 in USAH cover NIS/NIS+ in more detail.

ftpd

☞ **ftpd** is the File Transfer Protocol daemon, used by FTP client software to transfer files. As with sendmail, keeping up with security patches is critical in ftp implementations (the complex command set generally gives hackers plenty of places to look for flaws). On UNIX/Linux systems, a popular FTP implementation has been **wu-ftpd** (which has had many security flaws.)

ftpd

☞ **ftpd** is now being widely replaced by programs running over SSH/SSL such as **sftpd**, which gives a ftp-like capability over SSH (we will talk later more about **sshd**). While the user commands are similiar, the underlying protocol is quite different (see http://tools.ietf.org/html/draft-ietf-secsh-filexfer-10.txt for more details.) Security is much better since plaintext passwords are not sent over IP as they were

for the old protocol.

ftpd

☞ The old ftp protocol is specifed in RFC959.

☞ However, old ftp is not going away. There is still a useful place in Internet space for anonymous ftp, where the old version is better than more secure versions. If you don't care about security (and you don't with anonymous ftp), then why pay the overhead costs of using secure protocols?

Remote execution daemons (the "r" commands)

☞ A number of commands exist that permit a closer coupling between servers that support them. First there was **telnet** and **ftp**, then came the "r" command. Examples include: **rsh** (remote shell) and **rlogin** (remote login). A number of inetd-managed daemons exist to handle these services; strongly advised to disable all the "r" daemons in `/etc/inetd.conf`

Remote execution daemons (the "r" commands)

☞ In these security-conscious days you must move away from the "r" commands and into more secure equivalents, such as **ssh** (secure shell) and **scp**/**sftp** (secure copy)

☞ Ironically, original **telnet** and **ftp** are still quite useful!

named (and djbdns)

☞ **named** is the name of the popular Domain Name Server daemon and it comes as part of the BIND package, originally from UC Berkeley. We will discuss DNS later, but in short **named** provides:

⇛ Mapping of host names to IP addresses

⇛ Mapping of IP addresses to host names

named (and djbdns)

⫸ Other mappings

⫸ A distributed, reasonably robust protocol (RFC1034)

⫸ The standard BIND distribution has had some severe criticism; Dan Bernstein has an alternative package called

named (and djbdns)

djbdns which is more secure (and you still can receive $500 if you find a security lapse; see the offer at http://cr.yp.to/djbdns/guarantee.html) We will talk more about **djbdns**, but it lacks some features that many system administrators regard as critical.

named (and djbdns)

☞ SA RELEVANCE: DNS is a major SA task, if you control your own domain Both UNIX/Linux and 2003/2008 can act as a DNS server (as well as other operating systems).

fingerd

The finger protocol is an older method for getting information about users. As with the "r" commands, most consider **fingerd** (and the **finger** command) to be too problematic and remote finger should be disabled.

httpd

☞ Many web servers exist, both in the public domain and commercially. One of the most popular, Apache, uses the daemon name of **httpd**. It offers a great variety of services and enhancements; the relatively recent rewrite from 1.x to 2.x his finally gaining widespread acceptance (a few operating system distributions had been lingering on 1.3, which is still available as 1.3.41). See http://httpd.apache.org

httpd

☞ The management of web service is usually a fundamental service provided by the system administrator.

☞ The popular Windows 2003/2008 web server equivalent from Microsoft is IIS. Historically, it was ridden with serious security lapses, but more recent versions are vastly improved.

httpd

☞ There are lightweight servers also, such as **thttpd** (http://www.acme.com/software/thttpd) and specialized ones such that allow development in SOAP-like manner (see Perl http://www.cpan.org (JOAP, HTTP–Server–Simple, etc.))

Databases: LDAP servers

☞ The main open source choice for UNIX/Linux-based LDAP service is OpenLDAP (http://www.openldap.org). The daemon process is called **slapd**, and it supports replication (via **slurpd** or more recently, **syncrepl**), a wide variety of backends (including relational databases such as **MySQL** and **PostgreSQL**.)

Databases: LDAP servers

☞ However, Netscape has partnered with Red Hat (December 2005) to provide its Directory Server (http://www.redhat.com/directory_server/) as an open source LDAP server. It's technically somewhat interesting; its ability to handle very arbitrary backends (such as MySQL and Postgres) is more flexible than the **openldap** approach of "metadata".

Databases: relational

☞ **MySQL** – fast, becoming more featureful in version 5; the company itself was recently bought out by Sun. Expect to find the daemon **mysqld** in the process table. The client is **mysql**. Only a small amount of text file configuration in the poorly named file `/etc/my.cnf`; the rest is resident in the database.

Databases: relational

☞ **PostgreSQL** – very featureful, (ironically, it was supported by Sun well before Sun's acquisition of **MySQL**). Grep for "post" when you are looking for its daemons, which typically have "postmaster" and other keywords with "post" in them. The client is **psql**. Surprising amount of text configuration files, such as `hba.conf`. Generally not as fast as MySQL.

Miscellaneous UNIX/Linux daemons

☞ A number of other UNIX/Linux daemons have been around for years to provide more specialized services. Examples include such daemons as **dhcpd**, **bootpd**, **bootparamd**, **tftpd**, **rarpd** and others listed in /etc/inetd.conf.

Miscellaneous UNIX/Linux daemons

☞ SA RELEVANCE: If you are running a UNIX/Linux server on a network you should learn exactly what each and every network daemon does so you can decide if you want to run the security risk of offering that service. Come to think of it, this is true for Windows 2003/2008 (and any other computer system with external connections!)