SYSTEMS ADMINISTRATION

for the

LOCAL Project LAN

at

ABC Corporation
Anywhere, USA

John Smith
Mary Jones

January 4, 1996

SYSTEMS ADMINISTRATION
for the
LOCAL Project LAN
at
ABC Corporation
Anywhere, USA

John Smith
Mary Jones

January 4, 1996

This manual is intended to be a partial prototype for documentation that must be developed for any UNIX system. It is based upon documentation developed originally in the summer of 1992, and once described an actual LAN. That document was subsequently used as the basis for several other documents describing other LANs. Some excerpts from those documents are presented, but have been "sanitized" somewhat for instructional purposes, and bear no relationship to any current enviroment. ***The technical information in this document is outdated, irrelevant to any existing environment, and in some cases was purely fabricated for illustrative purposes.***

Any document like this needs to start off with some ground rules defining its purpose and the style it uses. A typical excerpt follows:

> This manual is intended to document the LOCAL Project LAN (hereafter referred to as the LOCAL LAN) at ABC Corporation in Anywere, USA.
>
> This manual will attempt to document *local* configurations and practices; it is assumed that the reader is familiar with standard UNIX† LAN administration practices, and has access to vendor-supplied documentation for the hardware and software installed on the LAN.
>
> Throughout this document, the standard UNIX command line interpreter will be assumed to be **ksh** or **bash**, and names of commands will appear in bold type, as in
>
> **command**
>
> Bold type, followed by a parenthesised number, will be used to specify entries in the vendor-supplied UNIX documentation for commands and file formats; for example, **who**(1) refers to the entry for **who** in section 1 of the vendor's UNIX manual. Keep in mind that many of the commands cited in this document will work only for the system administrator ("root"). Italics will be used to indicate command arguments that should be replaced by an appropriate word, such as
>
> **command** *filename*
>
> and will be used to emphasize important text. The names of systems, printers, networks, etc. will appear in bold-italic type:
>
> ***mymachine***
>
> A fixed-width font, `like this`, will be used when displaying the contents of data files, and the output from system commands. Specific high priority tasks that should be completed will be described under "*WORK TO-BE-COMPLETED*" following each section; more general suggestions will be found under "*RECOMMENDATIONS*". Specific suggestions for identifying, diagnosing, and correcting problems are de-

---

scribed under "*TROUBLESHOOTING*".

This document was originally produced using Free Software Foundation GNU **groff** version 1.09 software on a SUN workstation running SunOS 4.1.4. It is currently being stored as the file /usr/local/doc/sysadmin.me, accessible to all workstations on the network. It may be viewed on any X-window display using the command

gpic sysadmin.me | **gtbl** | **gtroff** -me | **gxditview** -

or can be printed to a PostScript printer using the command

gpic sysadmin.me | **gtbl** | **groff** -me | lp

This document was most recently (re)produced using **groff** version 1.11 software on a notebook PC running RedHat Linux version 5.2.

1. LOCAL LAN Description

Describe the purpose of the network, including intended customers and business purpose. Present historical information that may explain why the network developed the way it did. A brief sample excerpt:

> The primary purpose of the LOCAL LAN is to provide a computing environment for the ongoing development, testing, porting, and documentation of LOCAL project software. The LAN had also been intended to provide an environment for testing system administration techniques and tools that might eventually be migrated to production environments, although recent developments have reduced the probably of deploying a large-scale production environment similar to the environment described here.

1.1. System and Network Descriptions

Describe the physical location and layout of the network in detail; don't make assumptions that everybody knows this information. Example:

> The LOCAL LAN currently consists of approximately one hundred eighty UNIX-based servers and workstations, most of which are located on the 10th through 21st floors of ABC Company's Western Center (WC) building on Main Street. There are four primary servers: two IBM AIX servers providing NFS, NIS, and software licensing; and two DG servers providing ORACLE database facilities. Additional systems include a backup AIX server, a multi-processor SUN server, and an HP database server. Desktop workstations are all HP model 735 workstations with 64mB of memory and 1gB local disks. The primary servers are connected to a single FDDI subnet, which they do not share with other machines. The desktop workstations in the WC building are now dispersed over approximately 8 ETHERNET subnets, with a theoretical maximum of 62 workstations per subnet. This network configuration has eliminated the bottlenecks of the previous bridged network, at some sacrifice in flexibility (primarily, the need for a relatively large number of NIS slave servers). Besides the HP 735s in the WC building, additional HP 735s are located two blocks away at ABC Company's State Street facility.

Develop at least one diagram showing the network configuration, as appears in Figure 1.

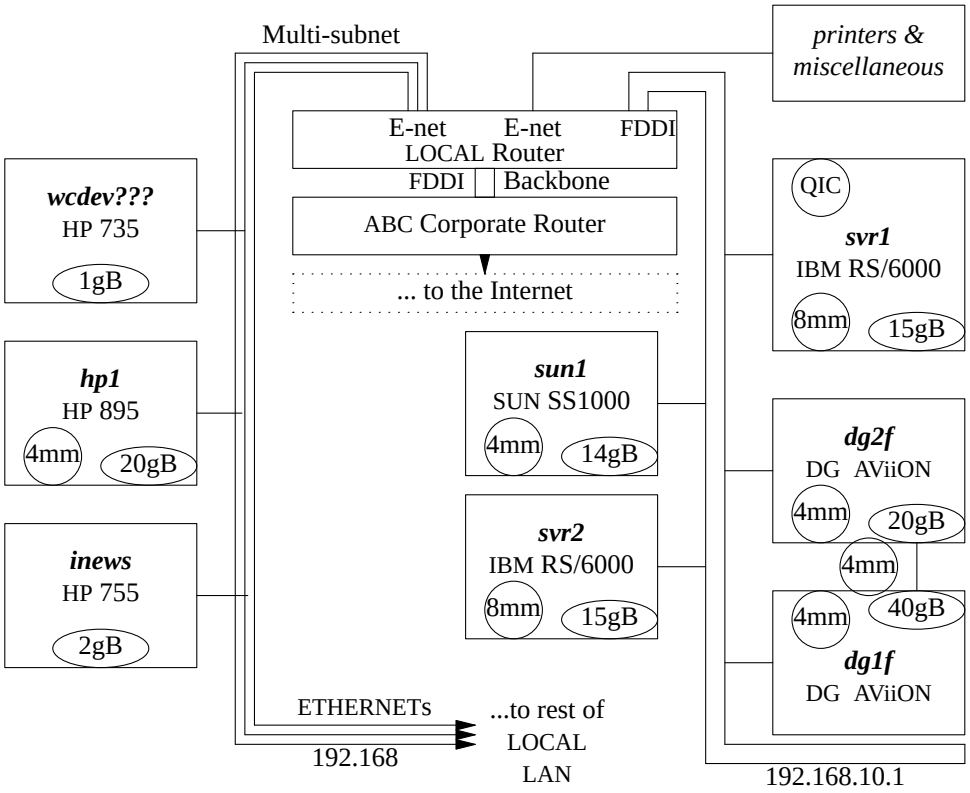

Figure 1: LOCAL LAN

Provide information on physical location access: keys, access cards, etc. Where applicable, describe facilities for storage of equipment and supplies, and describe any off-site storage arrangements. A real-life example:

> Due to politics surrounding the recent facility relocation, access to the computer center during off hours is officially impossible, even for authorized employees. However, access can be accomplished by prying open the unlocked window in room 112 on the ground floor of the Administration building and climbing through it. Keys located in the upper right hand drawer of the desk immediately inside that room will open the computer center door itself, located across the hall.

1.2. Support Personnel and Procedures

List all the persons and organizations responsible for the environment, along with contact information. Do not assume that anything is common knowledge - imagine yourself as being new to the organization, and include everyone you might possibly need to know. An example:

> Some of the individuals involved with various aspects of LOCAL LAN are listed in Figure 2, along with a brief description of their responsibilities.

Person	Org	Responsibilities	Tel (000-555-)	Page (000-555-)
John Smith	ABC	UNIX Sysadmin	9715	1234
Mary Jones	ABC	Corporate Network, Internet Gateway, Routers & DNS	8939	5678

Figure 2: People

> Problem resolution for LOCAL users is handled through the 4HELP mechanism, which provides users with a single phone number, 004-HELP, to call for reporting problems with computers, printers, the network, etc. Solution delivery, which includes such activities as evaluating new requirements or installing new hardware or software, as well as file and database server configuration issues, is handled through John Smith.

Be sure to specify emergency contacts for situations like physical disasters or security breaches (this should be a component of the disaster recovery plan, which may be separate from this document).

2. Naming Conventions

Every site has developed naming conventions for users, systems, and networks that are unique. Provide a section in this document for each.

2.1. File Naming Conventions

2.1.1. Home Directory Naming Conventions

Describe whether user home directories are named /home/*username* (or something else), and are located on servers or individual workstations, and whether home directories are exported using NFS. Is the **automount**er used, and if so are any non-standard mapping conventions used?

2.1.2. System Directory Naming Conventions

Describe any file structure deviation from a freshly installed version of the operating system you are using. Are there any directories added for local use (such as /usr/local/*)? Any contributed software added? Is the **automount**er used to make software available? Where are applications (such as databases) stored, and where are their data files located? What will break if any of this information is changed?

Describe the overall naming philosophy for physical partitions, volume groups, and logical volumes as appropriate. Is striping or mirroring used? What were the tradeoffs between performance, redundancy, and simplicity considered in the original system design? Were any alternatives tried, but found to be unworkable?

2.2. User Naming Conventions

How are user (login) names constructed? Is there a central repository from which names must be obtained (to keep names unique across the organization)? An example excerpt:

> ABC uses the convention of three initials (first, middle, and last), followed by the last four digits of the social security number, for user names. These names are shared between PROFS (the IBM mainframe system) and UNIX. Contractors are an exception; they use the same three initials, the last three digits of the social security number, followed by the character 'c'.

2.3. Domain and Host Naming Conventions

How are host names and domain names constructed? This is especially important to document for systems that have mulitple network interfaces, such as in "failover" or "switchover" environments, where IP names and addresses my "flbat" between machines and interfaces. It's very important to have a database (such as a plain ascii file) which records all the important information about each computer: hostname, IP address, hardware address, hardware configuration, physical location, primary user, etc. Describe who is repsonsible for maintaining the database as well as how to access it.

2.4. Network Naming Conventions

How are network devices, such as routers, named? While the network support group may have primary (and often sole) responsibility for determining this, it should still be documented here. Example excerpt:

> The network is divided into 64-host subnets for the ETHERNET segments, and 16-host subnets for the FDDI rings. The convention of locating routers addresses one below the highest address of a subnet is used. Routers are named according to the facility identifier for the wiring closet they reside in.

You might want to include sample output from **netstat -r** to show the routing tables, and explain how they would vary from one system to another based on the subnet configuration. Referencing a detailed network diagram (probably already prepared by the network support personnel) might be a good idea.

3. Disks and File Systems

3.1. Disk Configuration

Describe the physical and logical disk layout of each system (or type of system, for identical servers, workstations, etc.). Describe any version-specific commands (such as **prtvtoc** or **lvdisplay**) that might be helpful in determining the disk layout. If there are only a few typical configurations, include some sample output from the appropriate commands, as in Figure 3. Explain the addressing scheme used, which is likely to be specific to the UNIX version. For example:

> The "08" in the drive specification shows that hdisk0 through hdisk3 are attached to the "fast" SCSI-2 controller card in slot 8 in each server, while hdisk4 (on ***svr2*** and ***svr1***) is attached to the slower integrated SCSI-1 controller. The first of the final two digits in the hyphenated grouping shows the SCSI address set in the switches on each disk.

Tables are useful for showing the configurations, as in Figure 4. Remember to update this

---

+ hdisk4	00-00-0S-20	Other SCSI Disk Drive
+ hdisk0	00-08-00-00	2.0 gB SCSI Disk Drive
+ hdisk1	00-08-00-10	2.0 gB SCSI Disk Drive
+ hdisk2	00-08-00-20	1.0 gB SCSI Disk Drive
+ hdisk3	00-08-00-30	1.0 gB SCSI Disk Drive

Figure 3: IBM Server Disk Configuration

---

information whenever the disk layout is changed. Servers with disk arrays connected to multiple SCSI controllers can be more complicated, as shown in Figure 5. It's important to augment the information in the tables by physically labelling the cables and connectors. Be sure to describe any special considerations in developing the existing scheme, or any problems that might be anticipated (performance bottlenecks, single points of failure, etc.).

3.2. NFS Configuration

Describe how files are shared using NFS, if appropriate, and also mention any tradeoffs (performance vs. disk capacity, for example) that were encountered in designing the environment. If there are any problems routinely encountered (such as "Cannot contact **statd**..."), document how to fix them. Are file systems exported to the world, or if exporting is restricted, what maintenance will be required when new hosts are added?

3.3. The Automounter

Describe the use and configuration of the **automount** or **amd** daemons. How are changes to the maps propagated? Include in the text an example of how the maps are used, for example:

> The /home/users files are managed through the **automount**er using the following entry in the auto.master map:
>
> ```
> /home/users /etc/auto.home
> ```

System	hdisk	Volume Group	Capacity/ Free*4mB	Logical Volume	Use	Size *4mB	Mount Point
svr1 and ***svr2***	hdisk0	rootvg	479/228	hd6 hd5 hd7 hd4 hd2 hd1 hd3 hd9var	paging boot sysdump jfs jfs jfs jfs jfs	68 2 2 4 142 49 3 11	none /blv /mnt / /usr /home /tmp /var
svr1	hdisk1	adminvg	479/0	admin1lv	jfs	478	/archive
svr1	hdisk4	appl0vg	2166/0	abctemplv common2lv appl1lv backup1lv backup2lv	jfs jfs jfs jfs jfs	500 500 500 500 165	/abc /export/common /export/apps /backup1 /backup2
svr1	hdisk2 hdisk3	abcvg1	499/0	abc1lv	jfs	499	/export/abc
svr2	hdisk1	capplvg0	479/228	appllv	jfs	250	/export/apps
svr2	hdisk4	cadminvg	2144/0	ccomlv csparelv cbackup0lv cbackup1lv cbackup2lv	jfs jfs jfs jfs jfs	500 500 500 500 165	none none /cbackup0 /cbackup1 /cbackup2

Figure 4: IBM Server Logical Volume Configuration

System	Volume Path	Capacity/ Free*1mB	Virtual Disk /dev/dsk/	Use	Size *1mB	Mount Point
dg1f	sd(ncsc(0),0) sd(ncsc(0),1)	1000/0	x01	ufs	1000	/x01
dg1f	sd(ncsc(0),2)	500/0	*unused*	-	-	-
dg1f	sd(ncsc(1,7),0,0)	4000/500	root usr usr_opt_rpm usr_opt_xdt usr_opt_networker usr_opt_X11 var_opt_networker u01 swap	ufs ufs ufs ufs ufs ufs ufs ufs swap	50 300 5 35 30 80 20 1000 2000	/ /usr usr/opt/rpm usr/opt/xdt usr/opt/networker usr/opt/X11 var/opt/networker u01 -
dg1f	sd(ncsc(1,7),0,1)	4000/2000	u02 u03 u06	ufs ufs ufs	500 700 800	/u02 /u03 /u06
dg1f	sd(ncsc(6,7),1,2)	4000/2200	u05 u07	ufs ufs	1500 300	/u05 /u07
dg1f	sd(ncsc(6,7),1,3)	4000/2300	u04	ufs	1700	/u04
dg1f	sd(ncsc(6,7),1,4)	4000/1600	u08	ufs	2400	/u08
dg2f	sd(ncsc(0),0) sd(ncsc(0),1)	1000/0	u01	ufs	1000	/u01
dg2f	sd(ncsc(2,6),0,0)	2311/331	root usr usr_opt_rpm usr_opt_xdt usr_opt_networker usr_opt_X11 var_opt_networker x01	ufs ufs ufs ufs ufs ufs ufs ufs	50 300 5 23 20 80 10 1000	/ /usr usr/opt/rpm usr/opt/xdt usr/opt/networker usr/opt/X11 var/opt/networker x01
dg2f	sd(ncsc(2,6),0,1)	2311/1311	swap	swap	2000	-
dg2f	sd(ncsc(7,6),1,2)	2311/911	x02 x03	ufs ufs	650 750	/x02 /x03
dg2f	sd(ncsc(7,6),1,3)	2311/611	x04 x05	ufs ufs	900 800	/x04 /x05
dg2f	sd(ncsc(7,6),1,4)	2311/11	x08	ufs	2300	/x08

Figure 6: *dg1f* and ***dg2f*** Virtual Disk Configuration

The auto.home map looks like this:

```
dsm1035       -rw,bg,hard,intr wcdev853:/home/dsm1035
dof1930       -rw,bg,hard,intr svr2:/home/dof1930
lac1035       -rw,bg,hard,intr wcdev202:/home/lac1035
. . .
```

4. Printers

Describe all printers available on the network, including those accessible through other operating systems. Be sure to include detailed configuration information for each printer, such as the exact input used to create a printer using **sam** on a HP-UX. Also describe any commands used to configure the printer using a command like HP's **jetadmin**.

Give detailed trouble shooting procedures for each type of printer where applicable, and add to the *TROUBLESHOOTING* section as experience with the printer grows. For example:

> □ The QMS printers are unlikely to report operational errors, such as paper jams, until they have been power cycled. Therefore, if a QMS seems to be "idle" when it has jobs queued for it, but no error conditions are found on the control panel display, power cycle the printer, and (after a long wait for the reboot) check the display again. A paper jam or other error may have appeared after cycling the power. Correct the jam or other indicated problem if possible (instructions are located on the inside of the front compartment door of each printer), power cycle again if necessary. Check the queue for the printer by logging onto ***wcprtsrv*** and using the command
>
> **qchk** -P *queuename*
>
> to check the printer queue status. If the queue is down, it can be brought back up with the command
>
> **qadm** -U *queuename*
>
> □ The most frequent problem encountered with the SUN SPARCPrinter is failure of the paper size adjusting slides on the paper tray to remain in place, especially when the tray is removed for refilling paper. When the slides move, printing appears on the wrong portion of the paper. This situation can be corrected easily by removing the tray, checking that the slides are set according the instructions on the side of the paper tray, and re-inserting the tray. Power cycling the printer is not required.

5. System Installation

Describe how each type of system was originally installed from a distribution tape or CDROM. List in this section exactly which packages are to be included or excluded, and show the detailed answers to every question asked by the installation program. Any special considerations, such as "Don't load the international font data sets", must be provided. Also include the physical location of the installation media and vendor-supplied installation manuals. If your backup procedures include creating and using your own customized installation media or files (as with HP's Ignite mechanism), include the detailed procedures here (this is very UNIX-version specific, and may be site-specific as well). Don't forget to be very detailed, for example:

> When the code "c07" appears on the system LED, insert the second installation diskette. When "c07" appears again, insert the third installation diskette.
>
> Make sure the display is turned on and connected. A message will appear on the display asking you to press the F1 key and then hit enter - do so. A message will then appear asking you to "Insert BOS Install/Maint diskette". This is the forth installation diskette. Place this diskette in the disk drive and hit the enter key.

This description may require many pages, but has to be exact. Verify the instructions by having a person totally unfamiliar with the process actually perform it using only the instructions from this document. Remember to add any local configuration required to complete an installation.

6. System Software

6.1. Required Operating System Patches

Many systems require a bizzare array of patches to function at all. List all the patches currently installed, and what command can be used to produce the list, for example:

> The HP 735s run HP-UX 10.20, and installed patches can be viewed with the command
>
> **swlist -l product 'PH*'**

Make sure you specify which version of the operating system was initially installed (in case a later version has become available). Make sure to specify how much space the patches require, and

document how long they take to install, as well as where the patches are stored locally. Also document any patches that were installed but then backed out due to problems, to avoid making the same mistake twice. Document the process of how patches are identified for possible installation: who is responsible for finding the patches and who authorizes their installation? Remember that the patch strategy should be somewhat pro-active, so document any help your vendor provides in identifying patches that may be important for your installation.

6.2. Kernel Configuration

Describe every kernel configuration change made to your system, and include a table of the old vs. new values for every parameter changed. Remember to include any comments on future changes that might be required. List the sources for recommendations for any changes "not invented here". For example:

> It is absolutely essential to modify the kernel parameter `maxdsize` on the HP server ***hp1***, because without increasing the value the month-end reports will fail with an "out of memory" error. The `maxdsiz` value should be set to at least 0x800000. We learned this from Lori Ann at the HP Response Center.

Also describe the process for installing the new kernel, and describe how the old kernel can be restored in the event the new one turns out to be a (possibly unbootable) disappointment.

6.3. Electronic Mail

Describe the **sendmail** (or other mail transfer agent) configuration in detail, and include a listing of the configuration file (or at least a location where a listing can be found). Describe the capabilities and limitations of the mail system, and describe any problems encountered. Also describe a test process (for example, using **Mail -v**, although non-BSD-based systems may require invoking **sendmail** directly), and give sample results. This portion of the document may include references to the the classic O'Reilly Sendmail[1] text.

It is common to use a mail hub mechanism, and perhaps have clients NFS-mount a common mail spool (although precautions must be taken, and any require NFS mount options must be documented). Outgoing mail headers must often be re-written to allow return email, and cooperation with network administrators (who may control port access on corporate routers, and the corporate DNS servers) may be required. Regardless of who maintains the DNS nameserver configuration, record how MX records for typical systems should appear using **nslookup**. For example:

```
# nslookup
> set type=MX
> server relay.hp.com
Default Server:  relay.hp.com
Address:  15.255.152.2

> external.hp.com
Server:  relay.hp.com
Address:  15.255.152.2

Non-authoritative answer:
external.hp.com     preference = 100, mail exchanger = palsmtp.hp.com
external.hp.com     preference = 100, mail exchanger = atlsmtp.hp.com
external.hp.com     preference = 10, mail exchanger = charon.cns.hp.com
external.hp.com     preference = 50, mail exchanger = shadow.corp.hp.com

Authoritative answers can be found from:
```

[1] Bryan Costales and Eric Allman, Sendmail, 2nd Edition, O'Reilly and Associates, 1997.

```
external.hp.com     nameserver = atlrel1.hp.com
external.hp.com     nameserver = palrel1.hp.com
external.hp.com     nameserver = bbnrel4.hp.com
palsmtp.hp.com      internet address = 156.153.255.226
palsmtp.hp.com      internet address = 156.153.255.242
atlsmtp.hp.com      internet address = 156.153.255.210
atlsmtp.hp.com      internet address = 156.153.255.202
atlrel1.hp.com      internet address = 156.153.255.210
atlrel1.hp.com      internet address = 15.10.176.10
palrel1.hp.com      internet address = 15.81.168.10
palrel1.hp.com      internet address = 156.153.255.242
bbnrel4.hp.com      internet address = 15.199.64.4
bbnrel4.hp.com      internet address = 155.208.254.68

Non-authoritative answer:
external.hp.com preference = 100, mail exchanger = palsmtp.hp.com
external.hp.com preference = 100, mail exchanger = atlsmtp.hp.com
external.hp.com preference = 10, mail exchanger = charon.cns.hp.com
external.hp.com preference = 50, mail exchanger = shadow.corp.hp.com
```

6.3.1. **sendmail** Aliases

Describe how to create aliases using the traditional **sendmail** aliasing mechanism, for example

> Aliases can be created by adding entries to the /etc/aliases file on the the HP server ***hp1***, and then executing the **newaliases** program. An example alias entry is
>
> ```
> root: user1,user2,user3
> ```
>
> This line will cause mail sent to root to be sent to the three users listed; no mail will actually be sent to the user root. These **sendmail** aliases are resolved upon mail *delivery*, as opposed to aliases entered into a user's ~/.mailrc (for **mail** and **mailx**), which are resolved at the time the mail is *sent*. No changes should be required for the /etc/aliases file except on ***hp1***, because ***hp1*** should be performing local mail delivery for all of the desktop workstations.

If you use any other mail user agents, such as **elm**, be sure to desribe aliasing (or mail list creation) mechanisms for them as well.

Here is an excerpt from a typical *WORK TO-BE-COMPLETED* for this section, which might be used to describe either problems that need to be fixed or just shortcomings that might cause problems in the future:

> □ Aliases do not currently use the NIS alias map; the "Op" option to specify an NIS map name has not been incorporated into the sendmail.cf. So far this has not been an issue, because local aliases are resolved automatically on ***svr2***, as long as they appear in ***svr2's*** /etc/aliases file. One potential problem with using NIS for aliases is that the NIS master is not the mail hub; thus the usual attempt to use **sendmail** to rebuild the necessary map files on the NIS master will not work. Some customization would necessary to get this combination of servers to work properly.
> □ The /usr/local/etc/mailclients file has not been kept up to date, and in fact appears to be missing. Although this isn't necessarily a problem (because there are virtually no MX-records in the corporate gateway that point the desktop machines' mail to ***svr2***), it should be fixed in the event the gateway MX configuration changes in the future.

This is an example *TROUBLESHOOTING*, showing that you shouldn't wait until every problem has been resolved to start working on the documentation:

> □ Occasionally **sendmail** on ***svr2*** becomes hung in loop, repeatedly sending the same mail messages over and over. The cause is unknown, but the problem can be resolved by locating the offending message in the /var/spool/mqueue directory on ***svr2*** and removing it.

6.3.2. Mail Forwarding

Describe how *.forward* file and/or the **vacation** program may be used in your environment, if appropriate. In many environments, all UNIX email is redirected to an appropriate PC-based email system. Describe diagnotic procedures, such as using **telnet** to connect to port 25 on a system to carry on an interactive SMTP conversation. And as always document any anomalies, such as this *TROUBLESHOOTING* excerpt:

> □ AIX imposes what may be (?) a non-standard restriction that the .forward cannot be owned by root, no matter how readable by "other" it may be, so system administrators must refrain from creating the files themselves, or at least remember to **chown** the files as required, for forwarding to work properly.

6.3.3. External Mailing from PROFS

Even though we're discussing UNIX systems, if users commonly interact with other environments administrators may need to document how to that interaction works. This is especially important because even an experienced UNIX administrator may not have any idea how something like PROFS works (and of course might not want to know, either!).

6.4. NTP Time Service

Describe how time synchronization is performed using NTP. Many sites use this mechanism to coordinate time between systems, even if they are not connect to the Internet directly. An example excerpt:

> Version 3 of the NTP time server software is running on all AIX desktop workstations in the local area network. However, since incoming traffic from the Internet to the port reserved for NTP (123) is blocked by the ABC Internet gateway machine ***gw.abc.com***, and no time server within ABC is available, ***svr1*** has been set up as the single time server for the LOCAL LAN. All other AIX machines set their time from ***svr1*** continuously, by running the **xntpd** daemon. The configuration file for **xntpd**, /usr/local/etc/xntpd.client.conf, contains the single entry
>
> ```
> server  192.168.10.222
> ```
>
> The IP address is used here because there are unresolved issues involving the use of hostnames in the configuration file. The workstations execute **xntpd** from a script invoked by the inittab entry
>
> ```
> rcxntp:2:wait:/usr/local/bin/xntpd -c  \
>    /usr/local/etc/xntpd.client.conf -f /var/tmp/xntpd.drift
> ```
>
> that is executed whenever the workstation enters run level 2.
>
> The only exception to this configuration among the AIX machines is ***svr1*** itself, which must obtain a fairly accurate idea of the current time from an "authoritative" source. Since it cannot connect to an outside server via **xntpd**, due to the privileged port limitation, a special version of **ntpdate** has been compiled, which allows synchronizing time over the Internet without binding to a privileged port. The modified source is located in /usr/local/src/xntp/src/ntpdate directory; searching ntpdate.c for the string "ABC_Corp" will show the code that has been deleted using the C preprocessor. Currently, **cron** runs this modified **ntpdate** four times per hour, using the following crontab entry:
>
> ```
> 12,27,42,57 * * * * /usr/local/bin/ntpdate.unpriv \
>    louie.udel.edu eagle.tamu.edu  gus.ecn.purdue.edu \
>    > /usr/adm/ntpdate.log 2>&1
> ```
>
> These servers, two stratum 2 and one stratum 3, were selected primarily because Internet documentation indicated they were available for unrestricted NTP access, not because they have particularly accurate time. Other available servers could be substituted.

And again, it is important to remember to document any exceptions or problems that may occur, as in this *WORK TO-BE-COMPLETED* excerpt:

> □ The **xntpd** daemon seems to die occasionally, but only on the workstations. This may be because it is located on an NFS-mounted filesystem on all the workstations, and the network does occasionally become unavailable (it is the only NFS-mounted daemon). Installing the daemon, and configura-

tion file, locally on all the workstations might be a reasonable workaround to this problem. For the moment, a script runs nightly via **cron** on ***svr1*** to query all the clients and restart any daemons that may have died. The script entry is described in the "Cron Jobs" section.
□ **xntpd** does not appear to be capable of reliably making *large* adjustments to a system clock; thus, the clock should be close to the correct time before starting the daemon. This can be easily accomplished by using **ntpdate** first, to synchronize the local time with the time server.

6.5. DNS Configuration

Describe the DNS configuration, assuming DNS is used (otherwise, describe the use of the hosts file or NIS maps), as in this excerpt:

The network consists of a single DNS domain known as ***abc.com***. There is currently one primary nameserver, ***ns.abc.com***, and one secondary nameserver, ***ns2.abc.com***, both located at State Street. LOCAL has in addition a secondary nameserver, ***hp1***, located at the WC complex on Main Street. Its purpose is to provide a nameserver that is not dependent on the somewhat unreliable network link to State Street. The secondary nameservers gets all their zone files from the single primary nameserver. The zone files are updated twice a day by the Network Operations department. Once an hour the ***hp1*** secondary nameserver checks with the primary nameserver to verify that the secondary has the latest version of the zone files. If the version has changed, a new copy is downloaded to the secondary nameserver. Should the link to the primary nameserver be unavailable, the secondary nameserver will use the zone files that it had previously downloaded for up to 5 days. After 5 days in which the link to the primary nameserver is not reestablished, the secondary nameserver will cease to function using the zone files previously downloaded. It will be necessary to update the "expire" parameter in the various zone files to utilize the files beyond the 5 day limit.

Be aware that any modifications made to the zone files on the secondary nameserver will be overwritten on the next version update of the primary nameserver zone files (provided that the connection to the primary nameserver is up so that the updates can be retrieved).

You might want to add a *WORK TO-BE-COMPLETED* note to indicate that something like this might be a good added feature:

□ An automated procedure should be set up to prevent the secondary nameserver from turning itself off after not connecting to the primary for 5 days. This problem has already occurred when a configuration change on the primary nameserver resulted in files failing to download for a week, at which time at least some nameserver functionality was completely lost, even though valid (if dated) information was available on the local secondary server.

6.6. NIS Configuration

6.6.1. NIS Overview

Describe the NIS or NIS+ configuration for the network, and show (with narrative or a diagram) the relationship between master servers, slave servers, and clients. An example excerpt:

The local NIS environment is composed of the master NIS server ***svr1***, plus at least two slave servers for each subnet. The slave servers are required if broadcasts are to used to bind clients to servers, as is done on the LOCAL network, since the broadcasts do not pass through routers onto other subnets. The local domain is known as ***.abc.com***, which is easily confused with the DNS domain of the same name.

The master data files for most of the NIS maps are stored in /var/yp/src on ***svr1***, rather than the default (generally /etc) directory. This was done to allow configurability for certain local files, such as the /etc/passwd file, on ***svr1***. As an example, using /etc/passwd as the master for the NIS maps would preclude using it to prevent users from logging onto the server directly, possibly impacting performance, should loads increase beyond the present levels.

If there are considerations in the design of the local NIS that may not be obvious to someone new to the enviroment, meke sure to include a description. For example,

The database servers do not currently utilize NIS, and therefore do not share user accounts with

the workstations. This should probably be left as-is, since users are not supposed to utilize the database servers directly.

6.7. Other Shared and/or Distributed Files

Describe any important files shared through the use of symbolic links and NFS partitions, as well as any files copied around to each system using **rdist** or simple shell scripts.

6.7.1. Files Shared Using NFS

This section would include information on configuration files located on a server that are shared with clients, such as perhaps a **sendmail** configuration file.

6.7.2. Automated File Distribution

Describe how programs like **rdist** are used to update files, as well as any unique use of symbolic links or NFS mounting that might not be obvious. Describe the exact location and use of wrapper scripts that use **rdist** or **rcp**, including any that may be invoked automatically using the **cron** daemon. If **rcp** is used, remember to document the trust relationship between all the systems on the network, but be careful to describe any security aspects that were considered in developing the strategy.

7. Application Software

7.1. Vendor-Supplied Applications

Detail each application purchased from the system vendor (or that are products of the vendor, but supplied by a reseller), added to the basic UNIX installation. This section will probably be broken into subsections for each major server. One of the most important aspects to document is the installation and use of license manager programs for each application. Be sure to track licenses and document the source for each (with address and phone numbers for both sales and support contacts), along with the procedure to renew the license, and the date renewal is required. An example of a vendor-supplied application would be **glance** and related tools from HP.

7.2. Third-Party Commercial Software

Describe every application purchased, including purpose of the package, vendor name, contacts, installation process, and licensing information. This will typically be one of the largest sections, and should be broken into subsections for every package. A small excerpt from a subsection on the ***FrameMaker*** publishing software:

Three shared licenses for FrameMaker4 are installed in the /app/frame heierarchy on ***svr1***. These licenses, plus twenty-five FrameViewer licenses, are available to any workstation on the network. Two additional FrameMaker licenses are reserved for specific user-IDs. The license data, set up by the **fmaddlicense** program, is shown in Figure 7.

Product	License	Password	Users	Reserved For
maker	00-1-01-01-4-6536A	B2390B	1	ehp648c
maker	00-1-01-01-4-7157E	08734A	1	dxb944c
maker	00-1-01-01-4-71345	3AEE27	1	*shared*
maker	00-1-01-01-4-716E1	D56340	1	*shared*
maker	00-1-01-01-4-712B8	06DE23	1	*shared*
viewer	06-1-01-19-4-420E7	30F3CC	25	*shared*

Figure 7: FrameMaker License Data

Document any difficulties encountered with software or license manager installation, as shown in this excerpt:

> The Frame documentation on installing the software is extremely confusing with regard to license manager startup. The license manager is said to start "automatically"; this appears to mean that any Frame application that starts and is unable to contact a license manager at the expected IP address will try to **remsh** to that machine and start one up. There is no mention of the user account or trusted host permissions required to do this, although there is some reference to the fact that it may not work correctly (probably an understatement). We have attempted to circumvent this issue by starting the license manager, **fm_fls**, on ***svr1*** manually, by creating a /etc/rc.local file. The log file created by the license manager, by default fm_fls.log in Frame's fminit/tmp directory, is useful in determining the usage of Frame licenses, as well as diagnosing licensing problems.
>
> We have created an account, using the login name *frame*, for the purpose of installing and maintaining Frame; the password for this account is currently the same as for the user *root* on the servers.

7.3. Public Domain Software

Software variously referred to as "public domain", "free", "freely redistributable", "GNU", or more recently "open source", is a nearly indispensible component of many UNIX installation. The packages may come from publicly available Internet archive servers in either source code or binary form. Many of the packages are GNU software, from the Free Software Foundation. The name "GNU" stands for "GNU is Not UNIX" - but many GNU programs mimic the behavior of similar UNIX programs. Anyone working with this software should read the detailed "copyleft" license agreement, which must be made available to users (along with the source code, should they request it). It is important to document the layout of free software on each system, for example:

> The software, although actually installed into the /export/common or other hierarchies on ***svr1*** or ***svr2***, appears (through symbolic links) in the following hierarchies on each workstation:
>
> - /usr/local -for software supported to some extent by 4HELP.
> - /usr/contrib -for software that is completely unsupported.
>
> The /usr/contrib hierarchy is available to anyone who wants to make the effort to install software; the subdirectories bin, src, doc, info, man, lib, and include are all set up for writing by "other". The /usr/local hierarchy is primarily intended for software that is supported to one extent or another, although this is not strictly true, since some otherwise unsupported packages may have been obtained in pre-compiled binary form, and must be located in /usr/local to function properly.

A complete list of every free software package in use should be included in this document, along with descriptions of any users or projects that rely upon it, and the degree of support provided if any.

7.3.1. Internet News

Describe the mechanism by which users can access internet news on your system, and how the news itself is supplied. For example:

Several public domain software packages have been installed to provide internet news ("netnews") service to the LOCAL project. The server ***news*** is the local news server; it receives a feed from ***news.isp.net*** continuously, using Network News Transport Protocol (NNTP) in the form of the /usr/local/etc/**innd** daemon.

7.3.1.1. Newsgroup Administration

Describe which newsgroups your system subscribes to, and why. The policy for adding and deleting groups should be described. An example excerpt:

> The list of newsgroups ***news*** receives could be controlled at the ***news.isp.net*** end; however, that machine provides a full feed. Although not the most efficient method, groups are being selected locally by modifying the approriate entries in the /netnews/active file.
>
> Before modifying the file directly, it is extremely important to stop **innd** by becoming the user "news" and issuing the command:
>
> **ctlinnd** shutdown *'reason for shutdown'*
>
> The *reason* exists only to provide documentation for the log files. Groups can then be eliminated by deleting them from the active file, or added by providing a line of the form
>
> *groupname* 0000000000 000000001 y
>
> Once the active file has been modified, **innd** should be restarted using the command
>
> /usr/local/etc/**rc.news**
>
> which is the same command used to start **innd** during the system boot process.
>
> A description of the active file format can be found in the manual pages, located with the **innd** source code, in /usr/local/src/inn1.4sec. Very useful additional information can be found in the Frequently Asked Questions, located in the same source hierarchy.
>
> Rather than edit the active file directly, which requires stopping **innd**, groups can also be deleted from the file using the
>
> **ctlinnd** rmgroup *groupname*
>
> command. The file /netnews/DELETED_GROUPS is used to record the names of groups which have been deleted; this file needs to be updated manually. A script has been provided in /usr/local/etc, **rm_news_group.sh**, which reads a list of groups to remove from standard input, performs the necessary **ctlinnd** command, removes any existing articles posted to the group, and then appends the group name to the DELETED_GROUPS file. The script is commonly used by placing the names of groups to be deleted into a text file (*filename*), one per line, and invoking
>
> /usr/local/etc/**rm_news_group.sh** < *filename*
>
> The program will prompt for interactive input, regardless of whether it sees input coming from a file, as in the above example. The similar **add_news_group.sh** script allows groups to easily be added in the same manner. An additional script, **new_newsgroups.sh**, is run daily from cron to notify the news administrator (via email) of the addition of new news groups. The administrator can then chose to leave new groups in the active file, or delete them using the **rmgroup** process described previously.

Troubleshooting news installations is very important and should be explained in detail. Here is an example *TROUBLESHOOTING* discussion of how to correct a corrupted /netnews/active file:

> Editing the active file directly, as opposed to using **ctlinnd**, would likely result in the /netnews/active file becoming corrupted. This may be indicated by "can't symlink" accompanied by "throttled" messages in the /var/log/new/news.debug file. This situation can be rectified by rebuilding the active file, as follows (commands below can be found in the /netnews/bin directory):
>
> **ctlinnd** renumber ''
>
> The third argument above is a pair of single quotes with no space in between. This operation has to be performed *without* stopping **innd** (as described above), and will take several minutes. The FAQs, described above, contain many useful suggestions for daily operation of **innd**.

7.3.1.2. Expiring Articles

News accumulates at a very rapid rate and will soon fill up all but the largest disk areas. Describe how this problem is being resolved on your system, as in this excerpt:

> Articles are expired through the /netnews/bin/**news.daily** script, which is run nightly by **cron**, under the "news" user-id. The configuration file, /netnews/expire.ctl, contains comments explaining how to modify the retention times for articles. The **news.daily** script also emails reports to the news administrator nightly, providing statistics on news utilization. Available disk space is one of the most critical considerations; **innd** does not accommodate runnning out of space, so it's wise to err on the side of too much free space, and too many file system i-nodes, rather than too little or too few.

8. Routine Operations

8.1. Setting the Message of the Day

While setting the message-of-the-day may seem a standard UNIX task, X-windows does not provide a particularly standard way of displaying a message similar to the display of /etc/motd by the **login** program. This is usually rectified by one or more scripts run as part of the default user startup scripts, although the specifics are unique to each site, and thus must be documented here. An example excerpt:

> The locally developed **Xmotd** script, located in /usr/local/sbin, is called from the default .profile (in /common/admin). It sets variables specifying the color and size of the X-display, then displays the contents of a text file specified on its command line. The actual text files for each architecture are stored in /usr/local/etc/motd.$OS, where $OS is the output displayed by **uname** ("HP-UX" in the case of the HP machines). The message of the day can be modified by simply editing this text file.

8.2. Adding/Modifying/Deleting Users

8.2.1. Adding a New User

Sites that set up new user accounts only occasionally often use whatever mechanism is provided by their UNIX version for creating user accounts, such as **sam** on HP-UX. Other sites that must create numerous accounts frequently, such as in a university environment, often write extensive scripts to automate the task. Whatever the case, the process must be documented, with information included on how names, user-IDs, group-IDs, full names, home directories, and shells are to be selected and entered. Frequently there are ranges of user-IDs and group-IDs allotted to different departments or classifications of users, and often the information must be coordinated through a central authority of some kind (to eliminate possible duplication of names or ID numbers).

8.2.2. Updating User Accounts

Document the procedure for changing an account, including who can authorize the change and how the change can be performed and propagated to all the applicable systems. Include information on when and how an account can be disabled and re-enabled.

8.2.3. Removing User Accounts

Many sites leave old accounts languishing on the system, which is both a security risk and bad housekeeping. Accounts should be purged when users no longer have need for them, and the process should be documented, along with information on any archiving of the files that should take place. Remember to remove or reassign any files belonging to a user before the user is removed.

8.2.4. Customizing User and Desktop Profiles

When a new user account is created, a custom desktop profile, including X Desktop and shell configuration files, should usually be automatically installed in the user's home directory. The process for accomplishing this should be carefully documented, and the location of the master files specified by full path name.

8.2.5. Adding Workstations

Describe in detail the process for the local enviroment, including any authorizations and contact with other support groups (network, telecom, etc.) required.

8.2.6. Moving Workstations

The same considerations apply as for adding workstations. Remember to include any common problems, such as forgetting to change the IP address before moving a workstation to a new subnet.

8.2.7. Removing Workstations

The same considerations apply as for adding and moving workstations.

8.3. Booting and Shutdown

Describe in detail, with separate sections for each architecture type, both attended and unattended boot processes. For example, for HP machines, separate descriptions would be required for a 715 workstation and a K series server. If any special passwords are required (such as for an attended mode boot), document where they can be obtained (and how to work around not having them, if appropriate).

Some systems may have an LED display that can display hardware test error codes or other important information; document where this information can be found at the local site for each model of computer. Don't rewrite the vendor manual, but make sure to include where it can be found and note any problems that commonly occur at the local site, for example, in a *TROUBLESHOOTING* section:

> □ The 9gB disks on the servers frequently cause the boot process to hang while for an unknown reason. Attempts to resolve this problem by modifying the disk jumpers to supply terminator power to the bus have not been successful, however we have recently discovered that the disks will work properly if connected to the add-on SCSI controller card, as opposed to the built-in SCSI interface. Until all the systems are reconfigured to meet this requirement, they can usually be booted by temporarily powering off the disks, then powering them on once the system is running. The problems appear most often when the system has been powered off, as opposed to merely rebooted; however, apparent disk corruption on the 9gB may occur even when the systems are rebooted without powering off. Running **fsck** on any one of the 9gB partitions will likely cause the remaining partitions to become "clean", and enable them to be mounted normally.

Specify the time required for each model to complete each phase of the boot process, and list in detail all messages displayed, paying particular attention to any that seem unusual or that might indicate an error condition. Document the location of backup kernels and the exact commands required to boot them.

The /etc/inittab file controls the latter phases of the boot process on most system; include listings of the files and related scripts where appropriate.

8.4. Managing Failover

This section should describe the local process for failover/switchover of disk or other resouces between systems. Most systems have this capability, but perform it with varying degrees of automation - and varying degrees of success. Include a listing and description of all customized

scripts, along with references to vendor documentation where appropriate.

8.5. Performing Backups

8.5.1. Tape Drive Configuration

Describe the method used to name each tape drives, which varies from one UNIX vendor to another. Make sure the drives are physically labeled with both their hardware and logical drive names, and drive type (DDS2, etc).

8.5.2. Server Backups

Describe whatever the local procedure is, and describe any special procedures used to create bootable backups or backups of applications (such as databases). Servers are more likely to have a backup bootable disk, which is highly recommend to speed recovery. Remember that for systems like HP-UX, the data on the disks may have to be backed up separately from the logical volume manager disk layout. Document the exact layout of data on the backup media, for example:

> The backups are written to the tape array on the HP systems using the following layout:
>
> File 1: **fbackup** image of root
> File 2: **fbackup** image of /usr
> File 3: **fbackup** image of /export/home

For every backup process, create a section that details how the backup can be restored, and include sample sessions (such as by using the **script** command).

8.6. Accessing CDROMs

Describe local conventions for CDROM access, such as where CDROMs are usually mounted in the file system. Describe any vendor-specific information, such as the format of bootable installation CDROMs. Describe any special considerations, such as if the vendor documentation CDROM should always remain mounted in a particular drive.

8.7. Cron Jobs

Describe each **cron** job that already exists or (using *WORK TO-BE-COMPLETED*) should exist. Note any problems with jobs that routinely fail or produce any suspect output. An example excerpt:

> **sar** scripts that accumulate system usage statistics during working hours, and print summaries to the default printer. This is accomplished using the entries
>
> ```
> 20  7-21 * * 1-5 /usr/lib/sa/sa1 1800 2  >/dev/null 2>&1
> 0   22   *  *  1-5  /usr/local/etc/print.sar
> ```
>
> The information from the printouts is used to calculate system load and uptime for the weekly status reports. Since printouts can easily be lost, copies are automatically saved in the /usr/adm/sar_summary directory on each server; these copies are purged after thirty days.

8.8. Hardware Maintenance

Document all vendor contact information for each contract, including authorized contact personnel and any "handles" or "codewords" required. Also describe any local procedures that have to be performed, for example:

> There is a schedule for cleaning posted on the side of every server. At the appropriate time, the tape drives should be cleaned by inserting a cleaning cassette, the interior of the servers examined for dirt accumulation and cleaned, and the fans checked for proper operation.

Increasingly, disks can be hot-swapped without vendor attention. Be sure to document any systems that "phone home" when they detect problems, and test this functionality (such as by dialing the

modems manually) periodically. At some point, self-maintenance on some equipment may be a reasonable approach, particularly for workstations. You may be able to purchase spares for less than the cost of a maintenance contract.

8.8.1. Third-Party Hardware Maintenance

Describe any "finger pointing" problems in this section, along with any incompatibilities or features that might not be obvious, as in this excerpt:

> The vendor will also provide free technical support (configuration assistance, etc.) for the life of the drives, including assistance in moving them to other UNIX variants if required. The drives do require some software support in the form of a special formatting program provided. This program is supplied on 3.5" fbppy disks, and permits reformatting the drives in the event the format becomes corrupted (note that this software is not necessary to *use* the drives - only to reformat them in the event that becomes necessary). The drives are too large to be formatted with the standard HP formatting tools; attempting to format them with standard HP tools will fail and leave the drive with corrupted formatting. Detailed instructions for installing the drives on HP-UX and other UNIX systems are provided in the vendor installation manuals, located in the filing cabinet in Room 1804.

8.9. Software Maintenance

8.9.1. System Software Maintenance

Document the entire contact procedure for each system software vendor (SUN, HP, etc), including any contract numbers, code words, serial numbers, or purchase order numbers required. Identify who is authorized to contact each vendor and under what circumstances.

8.9.2. Application (Third-Party) Software Maintenance

Provide the same information for each application vendor as for the system software vendors. Create a subsection for each package.

8.9.2.1. *BrandX* Software

As an example for the entry required for each vendor, consider this entry for one software package:

> The only telephone contact currently listed for BrandX is in Sweden: +46-8-555-55-55. The company also has a US office which has never been known to be staffed. The software serial number is 07845, and the support contract, number 12345, allows any person (smart enough to figure out what time it is in Sweden) to call for assistance. It is also possible to send email to the US support office, using the address *support@brandx-usa.com*, which of course will be forwarded to Sweden after some prolonged delay. Note that the software (original distribution and patches) was delivered on QIC tape that appeared to be in QIC-24(!) or QIC-11(!!) format; it was unreadable on most of the "modern"(?) QIC drives on the servers, but could be read on an Archive Viper QIC-150 using SunOS 4.1.3, which has since been donated to the Computer Museum.

8.10. Supplies and Purchasing

8.10.1. On-Site Supplies

Identify where routine supplies can be obtained, for example this excerpt from one particularly handy site:

> Almost any supply you could possibly want can be found at the stockroom in the Tech Building basement, room 001. You can select whatever you want from bins of paper, pens, paperclips, fbppy disks, 10bT cables, DAT tape, etc. Non-commodity items like volt-ohm meters (really nice Flukes!), soldering irons, and tool sets, can be obtained from the clerks. No authorization

is required except your signature.

8.10.2. Ordering Supplies

Most sites will require some formal purchase process for certain items that fit between the traditional supply (paperclips) and a capital budget item (new server). Consider this possibility:

> Electronic supplies, including RS232 connectors, soldering tools, cable, car stereos(?) etc., can be purchased at XYZ Electonics, located directly across the main employee parking lot. Just say you're from ABC, sign the bill, and you can walk out with anything you want.

More complex environment may require a table to identify where or how to buy each commonly needed item, as in Figure 8.

Component	Vendor	Part Number	Contact Information
HP Laserjet4 Toner	HP	2222	Discount Warehouse, 800-555-5555
DC-6525 Tape	3M	DC6525	Global Supplies, 800-555-3333

Figure 8: Supplies

Clarify any supply selection that might not be obvious. For example, some newer tape drives will not perform at their highest density or speed capability with shorter length tapes, but the newer tapes might not be suitable for older tape drives. Specify brand preferences where required:

> 3M Black Watch tape has seemed to be unreliable in the TE-16 tape drive, although it has been more reliable than BASF tape. 3M 777 seems to be the most reliable tape we have used so far.

And don't forget the storage required for supplies, which can be important in helping to maintain an accurate inventory of items, as indicated in this *RECOMMENDATIONS* section:

> □ A cabinet for storing tapes, diskettes, and CDROMs would be very beneficial in keeping track of the many sets of distribution and backup media floating around.

8.10.3. Shipping Information

Provide both USPS and non-USPS addresses (remember that UPS and other carries cannot use Post Office Box addresses) if required for shipping of items to any sites where it may be required. Note if you have experienced problems actually getting items delivered.

8.11. Weekly Status Memo

Management often requests, or at least would like to have, a regular report on system activity. This can be a useful tool in demonstrating the need for upgrades, as well as to take credit for any improvements or upgrades. Document how the status reports should be prepared, and where they are archived:

> An ascii text version of each weekly memo is stored in the /usr/local/doc/newsletteredirectory; the **groff** original is kept in the /usr/local/doc/newsletter/src subdirectory.
>
> Preparing a new newsletter requires examining the /usr/adm/sar_summary directory on each server, and manually calculating uptime averages for each day (although frequently uptime will be 100%, greatly simplifying the calculations). To view the averages use the following command:
>
> **grep** Average *
>
> After creating a new newsletter in the src directory, which must be named news.*N*, where *N* is the sequence number of the new issue, verify that the uptime figures for the current week are correct. Then, run the script ./**avail.sh**, which will calculate the average historical uptime; these numbers must then be manually copied into the new issue. The newsletter can be printed using
>
> **gtbl** news.*N* | **groff** -me -fH |**lp**

or previewed with

gtbl news.*N* | **gtroff** -me -fH | **gxditview** -

An ascii version should also be created and installed:

gtbl news.*N* | **gnroff** -me | **col** -b > ../news.*DDmonthYY*

You may want to edit the ascii file which was created above, to remove the excessive number of blank lines. The existing file names will provide an example. Printed copies should be distributed as specified in the newsletter header, and at least one copy should be posted on the bulletin board in the duplication room (2017).

9. Security

9.1. Hardware Security

Specify any security that has been provided for the environment, remembering that hardware security refers to both protection from accidental loss (fire or natural disaster), as well as intentional theft or damage. An example:

> There is very little physical security to protect any of the desktop machines from physical loss or damage. The servers are located in the computer room, which is normally locked; only personnel responsible for maintaining the communication or computer equipment in the room have access. The remaining servers are located in the open lab area along with thee desktop workstations. All these machines would be subject to water damage in the event of a fire or sprinkler system failure. No attempt has been made to restrict access (via keyswitch, bootprom, or other means) to bootable devices attached to the desktop workstations or servers; thus the workstations are not secure from anyone equipped with an appropriate bootable device or media. No UPS or power conditioning is currently used for any machines other than the servers. The servers are connected to individual, idential APC UPS systems.

This might be an appropriate place to mention any problems with the equipment described, such as this excerpt:

> The UPS on ***hp1*** sometimes goes into bypass, with a corresponding alarm, when the system is power cycled. This appears not to be a function of the UPS being too small, as the rating is more than ample for the maximum possible power consumption, but happens for unknown reasons.

9.2. Software Security

Describe any provisions made in the standard UNIX configuration files to provide security, as well as any additional products installed:

> The /etc/hosts.equiv, /etc/hosts.lpd, and /.rhosts files are set up to permit relatively free activities between ABC LAN machines and personnel, while limiting access for others. The /etc/passwd files contain no password-less entries that would permit shell access.
>
> /etc/exports on each of the machines exporting files has been configured to allow only LOCAL LAN systems to access exported file systems. This has been accomplished using the netgroup host name aliasing mechanism in export access lists.

Mention whether features like mandatory password restrictions or password aging have been configured. If any programs like **cops** or **crack** are run to verify security, describe their operation in detail.

10. Performance Monitoring

10.1. Standard UNIX Performance Monitoring Tools

Several (fairly) standard UNIX programs are available to monitor performance of systems and network. Sample output from all of these programs (or at least those that are supported on each configuration), taken during typical periods of system activity, should be stored on the network available for future comparisons. The location of those output data files needs to be included in this

document,xbut has been left out of this sample. If you have proprietary programs to monitor performance, such as **glance** for HP, you should briefly describe them here, and provide information as to where detailed documentation can be found.

netstat ***netstat*** with the -i will produce a listing showning input errors, output errors, and collisions. Input and output errors should both be less than about .0002 times the total number of input or output packets. Collisions should generally be less than about .05 times the total number of output packets. Note that **netstat** data is historical; adding an *interval* option to the command line, as in

netstat -i 5

will restrict output to only current statistics.

nfsstat on an NFS client, with the -c option, will provide client NFS statistics. RPC retransmissions should be less than about .025 times total RPC calls. If this value is high, and similar to the value in the badxid field, the NFS server is probably overtaxed.

swapinfo is useful for monitoring swap space utilization. This is an HP-UX-specific command.

vmstat also helps identify problems related to the paging system. Normally **vmstat** should be run with an interval option:

vmstat 3

and the first (summary) line of output should generally be ignored. Changes in memory management algorithms tend to change the way **vmstat** statistics should be interpreted, but basically extensive page-out ("po") activity indicates that more memory would probably be beneficial. Page-in ("pi") activity is generally harmless, and may just indicate new processes being started. Adding the -S option to **vmstat** produces *swapping*, in addition to *paging*, activities. Frequent swap-outs may indicate that the system is short of memory.

iostat produces data that can help identify a disk or controller as a possible bottleneck. As with **vmstat**, **iostat** should be run with an interval option:

iostat 3

If the cpu is never idle, and there aren't any runaway processes on the system (check this first), the machine may be short of cpu power. If the cpu spends more than half its time running in "system" mode, suspect a slow i/o subsystem.

sar can be used in either of two distinct modes: interactive and analysis. If your system makes use of **sar**, be sure to carefully document any implementation dependencies, and note if/how **cron** entries may be used to accumulate or summarize **sar** information.

11. Future of the LOCAL LAN

This sections should contain future directions for the network, including planned upgrades.

Table of Contents

List of Figures