Strengthening AIX Security: A System-Hardening Approach

March 26, 2002

Authors: David Batten

Antonio Joglar

Linda St. Clair

Susan Schreitmueller

Rebecca Sanchez

Strengthening AIX Security: A System-Hardening Approach

Before reading the information in this paper, read the general information in “Notices” on page 41.

Contents

Introduction

AIX is an open UNIX operating environment that provides increased levels of integration, flexibility, and reliability that are essential for meeting the high demands of today's e-business applications. This focus on versatility allows AIX to be used under a wide variety of workloads, from running on a symmetric multiprocessor, capable of managing thousands of transactions per minute, to running on a single-node workstation used for application development.

Because one of the goals of AIX is to achieve this level of versatility and power, many services are immediately available when you finish installing the operating system. However, this can result in a configuration that is vulnerable to security exposures if the system is not configured appropriately. To minimize the number of possible security exposures, the system administrator must be able to identify the workload characteristics of the environment. *System hardening* is a global philosophy of system security that focuses strongly not only on detection, but also on prevention. It involves removing unnecessary services from the base operating system, restricting user access to the system, enforcing password restrictions, controlling user and group rights, and enabling system accounting.

Under the minimization procedures described in this paper, you identify and disable those operating system components and services that are not necessary for the task at hand. For example, if a system is being used as a file server, there is little benefit in enabling electronic mail (e-mail) services. E-mail services run as root, and there is a long history of e-mail-related security breaches. Proper system-hardening procedures call for these services to be shut down, resulting in a dedicated system with the fewest opportunities for exploitation.

This paper provides a baseline of AIX security for system administrators and offers guiding principles to help you begin securing your system. After reading it, you should be able to understand the importance of hardening an AIX system, as well as identify the location of the most important base operating system services and their functions. Hardening effectively empowers your system to provide that functionality which is specifically needed in your environment. You should also have a better idea of when it is appropriate to disable some of these services. This paper is not meant to be a thorough source of information on all AIX-related security issues, such as when to use the Lightweight Directory Access Protocol (LDAP) or Internet Protocol Security (IPsec). For information on those and other issues, refer to the appropriate documentation.

The scope of this paper includes enforcing adequate password rules, implementing proper user-security mechanisms, enabling system auditing, and monitoring file and directory access. Also covered are important X11 and CDE security issues, as well as how to identify open communication ports and list open files. The last section of the paper includes tables that summarize common AIX services and network tunable parameters. Use these tables as building blocks to start implementing an appropriate hardening strategy for your system.

Before you begin implementing any system-hardening measures, read this paper carefully. Note those items that you deem relevant to the security of your system. As you go through the sections of this paper that apply to your security needs, identify those files you will need to modify and back them up. It is always a good idea to back up modified files because this action enables you to revert to a previous configuration if you need to restore your previous security settings. After you complete your modifications and have thoroughly tested them to ensure that they work as you had planned, store the backup files in a secure place outside the newly secured system, such as a backup server. This precaution will prevent unauthorized reinstatement of your previous configuration that would disable all system-hardening modifications you have made.

In a typical environment, the installation of software patches, fixes, and updates can sometimes cause some of your modifications to revert to their original settings. You can easily avoid this problem by developing a security plan in which you log your actions and keep a listing of the location of your backup files.

Finally, perform all system-hardening procedures before the system goes into production. Bringing your system down when it is in production could prove costly to your operation, even if the objective is to make it more secure.

Topics Not Covered

This paper does not cover the following topics:

- Pluggable Authentication Modules (PAM)
- C2 Security Certification Level, as specified by Information Security Evaluation Criteria (ITSEC) and Information Technology Security Evaluation Manual (ITSEM)

For information about these topics, see the *AIX System Management Concepts: Operating Systems and Devices* or refer to the appropriate documentation.

Removing Information from Login Screens

Potential hackers can get valuable information from the default AIX login screen, such as the host name and the version of the operating system. This information would allow them to determine which exploitation methods to attempt. This section discusses how to avoid divulging unnecessary information about your system on your login screens. KDE and GNOME desktops share some of the same security issues. For more information about KDE and GNOME see the *AIX Installation Guide*.

Changing the Login Screen Welcome Message

To prevent displaying certain information on login screens, edit the *herald* parameter in the **/etc/security/login.cfg** file. The default *herald* contains the welcome message that displays with your login prompt. To change this parameter, you can either use the **chsec** command or edit the file directly.

The following example uses the **chsec** command to change the default *herald* parameter:

```
# chsec -f /etc/security/login.cfg -a default -herald
"Unauthorized use of this system is prohibited.\n\nlogin: "
```

For more information about the **chsec** command, see the *AIX Commands Reference, Volume 1*.

To edit the file directly, open the **/etc/security/login.cfg** file and update the *herald* parameter as follows:

```
default:
herald ="Unauthorized use of this system is prohibited\n\nlogin:"
 sak_enable = false
 logintimes =
 logindisable = 0
 logininterval = 0
 loginreenable = 0
 logindelay = 0
```

Note: Set the `logindisable` and `logindelay` variables to `a# >0` to make the system more secure.

Changing the CDE Login Screen

This security issue also affects the Common Desktop Environment (CDE) users. The CDE login screen also displays, by default, the host name and the operating system version. To prevent this information from being displayed, edit the **/usr/dt/config/$LANG/Xresources** file, where **$LANG** refers to the local language installed on your machine.

In our example, assuming that **$LANG** is set to **C**, copy this file into **/etc/dt/config/C/Xresources**. Next, open the **/usr/dt/config/C/Xresources** file and edit it to remove welcome messages that include the host name and operating system version.

For more information about CDE security issues, see “Managing X11 and CDE Concerns” on page 17.

Securing Unattended Terminals

Always lock your terminal when it is not being attended to prevent unauthorized access. Leaving system terminals unsecure poses a potential security hazard. To lock your terminal, use the **lock** command.

Note: If your interface is AIX windows, use the **xlock** command.

Strengthening User Security

To achieve an appropriate level of security in your system, develop a consistent security policy to manage user accounts. The most commonly used security mechanism is the access control list (ACL). For information about ACLs and developing a security policy, see the *AIX System Management Concepts: Operating Systems and Devices*. This section discusses additional security measures you can take in order to enforce your security policy more completely.

Password Security

Guessing passwords is one of the most common hacker attacks that a system undergoes. Therefore, controlling and monitoring your password-restriction policy is essential. AIX provides mechanisms to help you enforce a stronger password policy, such as establishing values for the following:

- Minimum and maximum number of weeks that can elapse before and after a password can be changed
- Minimum length of a password
- Minimum number of alphabetic characters that can be used when selecting a password

In addition to these mechanisms, you can further enforce stricter rules by restricting passwords so that they cannot include standard UNIX words, which might be hacked. This feature uses the dictionlist, which requires that you first have the **bos.data** and **bos.txt** filesets installed.

To implement the previously defined dictionlist, edit the following line in the **/etc/security/users** file:

```
dictionlist = /usr/share/dict/words
```

The **/usr/share/dict/words** file will now use the dictionlist to prevent standard UNIX words from being used as passwords.

For more information on using the dictionlist and passwords, see *AIX System Management Concepts: Operating Systems and Devices*.

Disabling Direct root Login

A common attack method of potential hackers is to obtain the super user, or root, password.

To avoid this type of attack, you can disable direct access to your root ID and then require your system administrators to obtain super-user privileges by using the **su** command. In addition to allowing you to remove the root user as a point of attack, restricting direct root access allows you to monitor which users gained super-user access, as well as the time of their action. You can do this by viewing the **/var/adm/sulog** file. Another alternative is to enable system auditing, which will report this type of activity.

To disable remote login access for your root user, edit the **/etc/security/user** file. Specify **false** as the rlogin value on the entry for root.

Before you disable the remote root login, examine and plan for situations that would prevent a system administrator from logging in under a non-root user ID. For example, if a user's home file system is full, then the user would not be able to log in. If the remote root login were disabled and the user who could **su** to root had a full home file system, then root could never take control of the system. This issue can be bypassed by system administrators creating home file systems for themselves that are larger than the average user's file system.

Enforcing Automatic Logoff

Another valid security concern results from users leaving their accounts unattended for a lengthy period of time. This situation allows an intruder to take control of the user's terminal, potentially compromising the security of the system.

To prevent this type of potential security hazard, you can enable automatic logoff on the system. To do this, edit the **/etc/security/.profile** file to include an automatic logoff value for *all* users, as in the following example:

```
TMOUT=600 ; TIMEOUT=600 ; export readonly TMOUT TIMEOUT
```

The number 600, in this example, is in seconds, which is equal to 10 minutes.

While the previous action allows you to enforce an automatic logoff policy for all users, system users can bypass some restrictions by editing their individual **.profile** files. To completely implement an automatic logoff policy, take authoritative action by providing users with appropriate **.profile** files, preventing write-access rights to these files. This action ensures that only root can change the **INTERNAL FIELD SEPARATOR** (**IFS**) environment variable, which is used by some programs such as **sed**, **awk**, and **cut**, in the **.profile** files.

Disabling Group and Outside File Access Permissions

Another measure that provides very tight security is to deny, by default, group and outside permissions on your user's files. You can accomplish this by setting the umask value to 077 for user accounts. This action causes all files created by users to have appropriate reading, writing, and executing permissions on their files, while denying such access to members of their group, as well as to outsiders.

Note: On SP machines, set the umask value to 022 during installation. The default umask value of a new user is set to 022. Remember to change this value to 077 after installation is completed for a higher level of security. These can be set in the `default` section of the **etc/security/user** file.

Hiding User Names and Passwords

To achieve a very high level of security, ensure that user IDs and passwords are not visible within the system. The **.netrc** files contain user IDs and passwords. This file is not protected by encryption or encoding, thus its contents are clearly shown as plain text. To find these files, run the following command:

```
# find 'awk -F: '{print $6}' /etc/passwd&' -name .netrc -ls
```

After you locate these files, delete them. A more effective way to save passwords is by setting up Kerberos.

Setting User Password Options

The following table lists recommended values for some security attributes related to user passwords. Password options are located in the **/etc/usr/security** file. You can edit this file to include any defaults you want to use to administer user passwords. Alternatively, you can use the **chsec** command. For more information on the **chsec** command, see the *AIX Commands Reference, Volume 1*. The values presented in the following table are taken from *AIX Security Tools*, an IBM Redbook.

Attribute	Description	Recommended Value
dictionlist	Verifies passwords do not include standard UNIX words	/usr/share/dict/words
histexpire	Number of weeks before password can be reused	26

Attribute	Description	Recommended Value
histsize	Number of password iterations allowed	20
maxage	Maximum number of weeks before password must be changed	4
maxexpired	Maximum number of weeks beyond *maxage* that an expired password can be changed by the user	2
maxrepeats	Maximum number of characters that can be repeated in passwords	2
minage	Minimum number of weeks before a password can be changed	1
minalpha	Minimum number of alphabetic characters required on passwords	2
mindiff	Minimum number of unique characters that passwords must contain	4
minlen	Minimum length of password	6 (8 for root user)
minother	Minimum number of nonalphabetic characters required on passwords	2
pwdwarntime	Number of days before the system issues a warning that a password change is required	5

Tightening System Default Login Parameters

Edit the **etc/security/login.cfg** file to set up base defaults for many login parameters, such as those you might set up for a new user (number of login retries, login re-enable, and login internal).

Removing Unnecessary Default User Accounts

During installation of the operating system, a number of default user and group IDs are created. Depending on the applications you are running on your system and where your system is located in the network, some of these user and group IDs can become security weaknesses, vulnerable to exploitation. If these users and group IDs are not needed, you can remove them to minimize security risks associated with them.

The following table lists the most common default user IDs that you might be able to remove:

User ID	Description
uucp, nuucp	Owner of hidden files used by uucp protocol
lpd	Owner of files used by printing subsystem
imnadm	IMN search engine (used by Documentation Library Search)
guest	Allows access to users who do not have access to accounts

The following table lists common group IDs that might not be needed:

Group ID	Description
uucp	Group to which uucpand nuucp users belong
printq	Group to which lpd user belongs
imnadm	Group to which imnadm user belongs

Analyze your system to determine which IDs are indeed not needed. There might also be additional user and group IDs that you might not need. Before your system goes into production, perform a thorough evaluation of available IDs.

Defining Access to the Trusted Communication Path

The Trusted Computing Base (TCB) is the part of the system that is responsible for enforcing systemwide information security policies. By installing and using the TCB, you can define user access to the trusted communication path, which allows for secure communication between users and the TCB. TCB features can only be enabled when the operating system is installed. To install TCB on an already installed machine, you will have to perform a preservation installation. Enabling TCB allows you to access the trusted shell, trusted processes, and the Secure Attention Key (SAK).

Because every device is part of the TCB, every file in the **/dev** directory is monitored by TCB. In addition, the TCB automatically monitors over 600 additional files, storing critical information about these files in **/etc/security/syschk.cfg**. If you are installing TCB, immediately after installing, back up this file to removable media, such as tape, CD, or disk, and store the media in a secure place.

For more information about TCB, see the *AIX System Management Guide: Operating Systems and Devices*.

Dealing with Special Situations

System administrators may be required to contend with many situations when strengthening their systems. This section discusses these special situations.

Special Permissions

If you set up special permissions for users and groups, document the special permissions being granted and the steps you have outlined to deal with security issues. Unless you document special situations, others will not be aware of those special situations and may bypass the steps you have put in place.

Special Privileges

When you install new software, such as databases or web servers, there can be issues with new accounts being created along with special privileges for those accounts. It is important to be aware of new IDs, their privileges and their ownership of files and directories, so that there is no inadvertent circumvention of your security policy.

Special Passwords

- Power-on password - A power-on password, when set, prevents someone from rebooting a machine by simply turning it off and then turning it back on again. If a bootable CD media is inserted into the CD-ROM drive, the system is rebooted, then the system will boot off the CD and therefore not adhere to its security configuration, causing a security exposure. If a system is rebooted, when the power-on password is set, then the system will require that power on password during the boot cycle.
- Supervisory password - Setting a supervisory password prevents an unauthorized user from booting into maintenance mode using installation media (installation CD, mksysb tape/CD). Booting off of such media allows full access to files and directories without security restrictions that you may have set in place. A supervisory password locked system, if the password is lost, will need to be serviced by IBM in order to unlock it.
- Root password - There are times when the power of the root password will need to be utilized, such as with ftp and telnet. Be aware of when the root user power will need to be invoked, and plan your security implementation around those instances.

Security Weaknesses

Be aware of systems in your network that might have security weaknesses. If an intruder breaks into a machine in your network, access may be granted to other machines through permissions set up between the point of entry machine and other systems in the network. Some intruders scan networks for certain machine types and certain versions of operating systems to find one to break into, and they can then use that point of entry to gain access to all other machines in the network.

Enabling System Auditing

Users regularly perform various system actions that you will want to monitor more closely. By setting up system auditing, you can record security-relevant information, which can be analyzed to detect potential and actual violations of the system security policy.

Predefined audit events can be found in the **/etc/security/audit/events** file. You can automate auditing by setting up the cron facility to generate regular reports.

For more information about system auditing, see the *AIX System Management Concepts: Operating Systems and Devices* and *AIX System Management Guide: Operating Systems and Devices*.

Monitoring Files, Directories, and Programs

This section discusses the mechanisms you can use to monitor access to files, directories, and executable programs.

Note: It is important to note the default umask values for the system, as well as any particular situations that are exceptions. It is typical to have specialty logins that are required to support database installations and ownership of items such as Tivoli Storage Manager (TSM) or ADSTAR Distributed Storage Manager (ADSM) users. It is typical to have a requirement in an environment to maintain speciality IDs for database administration and backup and recovery. It is advisable to limit the login access by way of duration or time of day for accounts that have particularly high security access.

Removing Obsolete Files

Occasionally, you need to remove unwanted and unneeded files from your system. AIX provides you with the **skulker** command, which allows you to automatically track and remove obsolete files. This facility works on candidate files located in the **/tmp** directory, executable **a.out** files, core files, and **ed.hup** files. To run the **skulker** command, type

```
# skulker -p
```

You can automate the **skulker** command by setting up the cron facility to perform this task regularly.

For more information about the **skulker** command, see the *AIX System Management Guide: Operating Systems and Concepts* and *AIX Commands Reference, Volume 5*.

Removing Unowned Files

When a user ID is removed, that user's files then have no owner assigned to them. To identify files that have no owner, you can use the **find** command as follows:

```
# find / -nouser -ls
```

After identifying files that have no owners, determine whether the files are needed. If they are needed, assign them to a different user. Otherwise, you can remove those files from the system.

Managing Unauthorized Remote Host Access

Some programs use the **.rhosts** file to gain access to a system. In some cases, access can be granted to unauthenticated users. To avoid this situation, remove the **.rhosts** file from your system.

For HACMP clusters, **.rhosts** files are required. Instead of removing them from these configurations, set the permissions to 600 and assign ownership of the files to **root.system**.

To find **.rhosts** files, run the following command:

```
# find / -name .rhosts -ls
```

Monitoring Executable Files

To monitor the activity of critical executable files, you need a good understanding of how these files are being used. The executable files that you need to monitor are those that are owned by root and have either their SUID or SGID bits set.

After carefully monitoring these files during normal system activity, you can generate a report that includes a list of files that are normally executed. You can then contrast that report with subsequent reports that show new files with these attributes that were set without your knowledge. To create the baseline report, run the following commands:

```
# find / -perm -4000 -user 0 -ls
# find / -perm -2000 -user 0 -ls
```

Managing cron and at Jobs

To manage **cron** and **at** jobs, you must do the following:

- Make sure the only user listed in **cron.allow** and **at.allow** files is root.
- Remove the **cron.deny** and **at.deny** from the **var/adm/cron** directory.
- Ensure that **cron** and **at** jobs are owned and writeable only by root.

Managing X11 and CDE Concerns

This section discusses potential security vulnerabilities involved with the XII X server and the Common Desktop Environment (CDE).

Removing the /etc/rc.dt File

Although running the CDE graphical user interface (GUI) is convenient for users, security issues are associated with it. For this reason, do not run CDE on servers that require a high level of security. The best solution is to avoid installing CDE (dt) file sets. If you have installed these file sets on your system, consider uninstalling them, especially **/etc/rc.dt** script, which starts CDE.

For more information about CDE, see the *AIX System Management Guide: Operating Systems and Devices*.

Preventing Unauthorized Monitoring of Remote X Server

An important security issue associated with the X11 server is unauthorized silent monitoring of a remote server. The **xwd** and **xwud** commands can be used to monitor X server activity because they have the ability to capture keystrokes, which can expose passwords and other sensitive data. To solve this problem, remove these executable files unless they are necessary under your configuration, or, as an alternative, change access to these commands to be root only.

The **xwd** and **xwud** commands can be found in the **X11.apps.clients** file set.

If you do need to retain the **xwd** and **xwud** commands, consider using OpenSSH or MIT Magic Cookies. These third-party applications help prevent the risks that are created by running the **xwd** and **xwud** commands.

For more information on OpenSSH and MIT Magic Cookies, refer to each application's respective documentation.

Enabling and Disabling Access Control

The X server allows remote hosts to use the **xhost +** command to connect to your system. Ensure that you specify a host name with the **xhost +** command, because it disables access control for the X server. This allows you to grant access to specific hosts, which eases monitoring for potential attacks to the X server. To grant access to a specific host, run the **xhost** command as follows:

```
# xhost + hostname
```

For more information about the **xhost** command, see the *AIX Commands Reference, Volume 6.*

Disabling User Permissions to Run the xhost Command

Another way to ensure that the **xhost** command is being used appropriately is to restrict execution of this command to super user authority only. To do this, use the **chmod** command to change the permissions of **/usr/bin/X11/xhost** to 744.

```
chmod 744/usr/bin/X11/xhost
```

Ensure that you specify a host name with the **xhost** command because it disables access control for the X server. This allows you to grant access to specific hosts, which eases monitoring for potential attacks to the X server.

If you do not specify a host name, access will be granted to all hosts.

Disabling Unnecessary Services

This section discusses open communication ports and how you can identify and close those ports.

Identifying Network Services with Open Communication Ports

Client-server applications open communication ports on the server, allowing the applications to listen to incoming client requests. Because open ports are vulnerable to potential security attacks, identify which applications have open ports and close those ports are open unnecessarily. This exercise is useful because it allows you to understand what systems are being made available to anyone who has access to the Internet.

To determine which ports are open, you must:

1. Identify the services with the **netstat** command as follows:

   ```
   # netstat -af inet
   ```

   The following is an example of this command output. The last column of the **netstat** command output indicates the state of every service. Services that are waiting for incoming connections are in the LISTEN state.

Active Internet connection (including servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	(state)
tcp4	0	0	*.echo	*.*	LISTEN
tcp4	0	0	*.discard	*.*	LISTEN
tcp4	0	0	*.daytime	*.*	LISTEN
tcp	0	0	*.chargen	*.*	LISTEN
tcp	0	0	*.ftp	*.*	LISTEN
tcp4	0	0	*.telnet	*.*	LISTEN
tcp4	0	0	*.smtp	*.*	LISTEN
tcp4	0	0	*.time	*.*	LISTEN
tcp4	0	0	*.www	*.*	LISTEN
tcp4	0	0	*.sunrpc	*.*	LISTEN
tcp	0	0	*.smux	*.*	LISTEN
tcp	0	0	*.exec	*.*	LISTEN
tcp	0	0	*.login	*.*	LISTEN
tcp4	0	0	*.shell	*.*	LISTEN
tcp4	0	0	*.klogin	*.*	LISTEN
udp4	0	0	*.kshell	*.*	LISTEN
udp4	0	0	*.echo	*.*	
udp4	0	0	*.discard	*.*	
udp4	0	0	*.daytime	*.*	
udp4	0	0	*.chargen	*.*	

Active Internet connection (including servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	(state)
udp4	0	0	*.time	*.*	
udp4	0	0	*.bootpc	*.*	
udp4	0	0	*.sunrpc	*.*	
udp4	0	0	255.255.255.255.ntp	*.*	
udp4	0	0	1.23.123.234.ntp	*.*	
udp4	0	0	localhost.domain.ntp	*.*	
udp4	0	0	name.domain..ntp	*.*	

.....

2. After you have identified which services are listening, open the **/etc/services** file and check the Internet Assigned Numbers Authority (IANA) services to map the service to port numbers within the operating system.

   The following is a sample fragment of this file:

```
tcpmux                          1/tcp                # TCP Port Service Multiplexer
tcpmux                          1/tcp                # TCP Port Service Multiplexer
Compressnet                     2/tcp                # Management Utility
Compressnet                     2/udp                # Management Utility
Compressnet                     3/tcp                # Compression Process
Compressnet                     3/udp                Compression Process
Echo                            7/tcp
Echo                            7/udp
discard                         9/tcp                sink null
discard                         9/udp                sink null
..............
rfe                             5002/tcp             # Radio Free Ethernet
rfe                             5002/udp             # Radio Free Ethernet
rmonitor_secure                 5145/tcp
rmonitor_secure                 5145/udp
pad12sim                        5236/tcp
pad12sim                        5236/udp
sub-process                     6111/tcp             # HP SoftBench Sub-Process Cntl.
sub-process                     6111/udp             # HP SoftBench Sub-Process Cntl.
xdsxdm                          6558/ucp
xdsxdm                          6558/tcp
afs3-fileserver                 7000/tcp             # File Server Itself
```

```
afs3-fileserver                 7000/udp            # File Server Itself
af3-callback                    7001/tcp            # Callbacks to Cache Managers
af3-callback                    7001/udp            # Callbacks to Cache Managers
```

3. Close down the unnecessary ports by removing the running services.

Listing Open Files

It is useful to identify TCP sockets that are in LISTEN state and idle UDP sockets that are waiting for data to arrive. Use the **lsof** command, a variant of the **netstat -af** command. The **lsof** command is included with AIX 5.1 and is located in the *AIX Toolbox for Linux Applications* CD.

For example, to display the TCP sockets in LISTEN state and the UDP sockets in IDLE state, run the **lsof** command as follows:

```
# lsof -i | egrep "COMMAND|LISTEN|UDP"
```

The output produced is similar to the following:

Command	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
dtlogin	2122	root	5u	IPv4	0x70053c00	0t0	UDP	*:xdmcp
dtlogin	2122	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
syslogd	2730	root	4u	IPv4	0x70053600	0t0	UDP	*:syslog
X	2880	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
X	2880	root	8u	IPv4	0x700546dc	0t0	TCP	*:6000(LISTEN)
dtlogin	3882	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
glbd	4154	root	4u	IPv4	0x7003f300	0t0	UDP	*:32803
glbd	4154	root	9u	IPv4	0x7003f700	0t0	UDP	*:32805
dtgreet	4656	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)

.....

After identifying the process ID, you can obtain more information about the program by executing the following command:

```
" # ps -fp PID#"
```

The output contains the path to the command name, which you can use to access the program's man page.

Summary of Common AIX Services

The following table lists the more common services within AIX. Use this table to recognize a starting point for hardening your system.

Before you proceed with your system-hardening tasks, back up all your original configuration files, especially:

- **/etc/inetd.conf**
- **/etc/inittab**
- **/etc/rc.nfs**
- **/etc/rc.tcpip**

Service	Daemon	Started by	Function	Comments
inetd/bootps	inetd	/etc/inetd.conf	bootp services to diskless clients	• Necessary for NIM and remote booting of systems • Works concurrently with tftp • Disable in most cases
inetd/chargen	inetd	/etc/inetd.conf	character generator (testing only)	• Available as a TCP and UDP service • Provides opportunity for Denial of Service attacks • Disable unless you are testing your network
inetd/cmsd	inetd	/etc/inetd.conf	calendar service (as used by CDE)	• Runs as root, therefore a security concern • Disable unless you require this service with CDE • Disable on back room database servers
inetd/comsat	inetd	/etc/inetd.conf	Notifies incoming electronic mail	• Runs as root, therefore a security concern • Seldom required • Disable
inetd/daytime	inetd	/etc/inetd.conf	obsolete time service (testing only)	• Runs as root • Available as a TCP and UDP service • Provides opportunity for a Denial of Service PING attacks • Service is obsolete and used for testing only • Disable

Service	Daemon	Started by	Function	Comments
inetd/discard	inetd	/etc/inetd.conf	/dev/null service (testing only)	• Available as TCP and UDP service • Used in Denial of Service Attacks • Service is obsolete and used for testing only • Disable
inetd/dtspc	inetd	/etc/inetd.conf	CDE Subprocess Control	• This service is started automatically by the inetd daemon in response to a CDE client requesting a process to be started on the daemon's host. This makes it vulnerable to attacks • Disable on back room servers with no CDE • CDE might be able to function without this service • Disable unless absolutely needed
inetd/echo	inetd	etc/inetd.conf	echo service (testing only)	• Available as UDP and TCP service • Could be used in Denial of Service or Smurf attacks • Used to echo at someone else to get through a firewall or start a datastorm • Disable
inetd/exec	inetd	/etc/inetd.conf	remote execution service	• Runs as root and therefore is dangerous • Requires that you enter a user id and password, which are passed unprotected • This service is highly susceptible to being snooped • Disable
inetd/finger	inetd	/etc/inetd.conf	finger peeking at users	• Runs as root and therefore is dangerous • Gives out information about your systems and users • Disable

Service	Daemon	Started by	Function	Comments
inetd/ftp	inetd	/etc/inetd.conf	file transfer protocol	• Runs as user root • User id and password are transferred unprotected, thus allowing them to be snooped • Disable this service and use a public domain secure shell suite
inetd/imap2	inetd	/etc/inetd.conf	Internet Mail Access Protocol	• Ensure that you are using the latest version of this server • Only necessary if you are running a mail server. Otherwise, disable • User ID and password are passed unprotected with service
inetd/klogin	inetd	/etc/inetd.conf	Kerberos login	• Enabled if your site uses Kerberos authentication
inetd/kshell	inetd	/etc/inetd.conf	Kerberos shell	• Enabled if your site uses Kerberos authentication
inetd/login	inetd	/etc/inetd.conf	rlogin service	• Susceptible to IP spoofing, DNS spoofing • Data, including User IDs and passwords, is passed unprotected • Runs as root and is therefore dangerous • Use a secure shell instead of this service
inetd/netstat	inetd	/etc/inetd.conf	reporting of current network status	• Could potentially give network information to hackers if run on your system • Disable
inetd/ntalk	inetd	/etc/inetd.conf	Allows users to talk with each other	• Runs as root and is therefore dangerous • Not required on production or back room servers • Disable unless absolutely needed

Service	Daemon	Started by	Function	Comments
inetd/pcnfsd	inetd	/etc/inetd.conf	PC NFS file services	• Disable service if not currently in use • If you need a service similar to this, consider Samba, as the pcnfsd daemon predates Microsoft's release of SMB specifications
inetd/pop3	inetd	/etc/linetd.conf	Post Office Protocol	• User IDs and passwords are sent unprotected • Only needed if your system is a mail server and you have clients who are using applications that only support POP3 • If your clients use IMAP, use that instead, or use the POP3s service. This service has a Secure Socket Layer (SSL) tunnel • Disable if you are not running a mail server or have clients who need POP services
inetd/rexd	inetd	/etc/inetd.conf	remote execution	• Runs as root and therefore is dangerous • Peers with the **on** command • Disable service • Use **rsh** and **rshd** instead
inetd/quotad	inetd	/etc/inetd.conf	reports of file quotas (for NFS clients)	• Only needed if you are running NFS file services • Disable this service unless required to provide an answer for the **quota** command • If you need to use this service, keep all patches and fixes for this service up to date
inetd/rstatd	inetd	/etc/inetd.conf	Kernel Statistics Server	• If you need to monitor systems, use SNMP and disable this service • Required for use of the **rup** command

Service	Daemon	Started by	Function	Comments
inetd/rusersd	inetd	/etc/inetd.conf	info about user logged in	• This is not an essential service. Disable • Runs as root and therefore is dangerous • Gives out a list of current users on your system and peers with **rusers**;
inetd/rwalld	inetd	/etc/inetd.conf	write to all users	• Runs as root user and therefore is dangerous • If your systems have interactive users, you might need to keep this service • If your systems are production or database servers, this is not needed • Disable
inetd/shell	inetd	/etc/inetd.conf	rsh service	• Disable this service if possible. Use Secure Shell instead • If you must use this service, use the TCP Wrapper to stop spoofing and limit exposures • Required for **xhier**
inetd/sprayd	inetd	/etc/inetd.conf	RPC spray tests	• Runs as root user and therefore is dangerous • Might be required for diagnosis of NFS network problems • Disable if you are not running NFS
inetd/systat	inetd	/etc/inted.conf	″ps -ef″ status report	• Allows for remote sites to see the process status on your system • This service is disabled by default. You must check periodically to ensure that the service has not been enabled

Service	Daemon	Started by	Function	Comments
inetd/talk	inetd	/etc/inetd.conf	establish split screen between 2 users on the net	• Not a required service • Used with the **talk** command • Provides UDP service at Port 517 • Disable unless you need multiple interactive chat sessions for UNIX user
inetd/ntalk	inetd	/etc/inetd.conf	″new talk″ establish split screen between 2 users on the net	• Not a required service • Used with the **talk** command • Provides UDP service at Port 517 • Disable unless you need multiple interactive chat sessions for UNIX user
inetd/telnet	inetd	/etc/inetd.conf	telnet service	• Supports remote login sessions, but the password and ID are passed unprotected • If possible, disable this service and use Secure Shell for remote access instead
inetd/tftp	inetd	/etc/inetd.conf	trivial file transfer	• Provides UDP service at port 69 • Runs as root user and might be compromised • Used by Network Installation Management (NIM) • Disable unless you are using NIM or have to boot a disk-less workstation

Service	Daemon	Started by	Function	Comments
inetd/time	inetd	/etc/inetd.conf	obsolete time service	• Internal function of **inetd** that is used by **rdate** command. • Available as TCP and UDP service • Sometimes used to synchronize clocks at boot time • Service is outdated. Use **ntpdate** instead • Disable this only after you have tested your systems (boot/reboot) with this service disabled and have observed no problems
inetd/ttdbserver	inetd	/etc/inetd.conf	tool-talk database server (for CDE)	• The **rpc.ttdbserverd** runs as root user and might be compromised • Stated as a required service for CDE, but CDE is able to work without it • Should not be run on back room servers or any systems where security is a concern
inetd/uucp	inetd	/etc/inetd.conf	old fashioned UUCP network	• Disable unless you have an application that uses UUCP
inittab/dt	init	/etc/rc.dt script in the /etc/inittab	desktop login to CDE environment	• Starts the X11 server on the console • Supports the X11 Display Manager Control Protocol (xdcmp) so that other X11 stations can log into the same machine • Service should be used on personal workstations only. Avoid using it for back room systems
inittab/dt_nogb	init	/etc/inittab	desktop login to CDE environment (NO graphic boot)	• No graphical display until the system is up fully • Same concerns as **inittab/dt**

Service	Daemon	Started by	Function	Comments
inittab/httpdlite	init	/etc/inittab	web server for ″docsearch″	• Default web server for the docsearch engine • Disable unless your machine is a documentation server
inittab/i4ls	init	/etc/inittab	license manager servers	• Enable for development machines • Disable for production machines • Enable for back room database machines that have license requirements • Provides support for compilers, database software, or any other licensed products
inittab/imnss	init	/etc/inittab	search engine for ″docsearch″	• Part of the default web server for the docsearch engine • Disable unless your machine is a documentation server
inittab/imqss	init	/etc/inittab	search engine for ″docsearch″	• Part of the default web server for the docsearch engine • Disable unless your machine is a documentation server
inittab/lpd	init	/etc/inittab	BSD line printer interface	• Accepts print jobs from other systems • You can disable this service and still send jobs to the print server • Disable this after you confirm that printing is not affected
inittab/nfs	init	/etc/inittab	Network File System/Net Information Services	• NFS and NIS services based which were built on UDP/RPC • Authentication is minimal • Back room database servers should have no need for this • Disable this for back room machines

Service	Daemon	Started by	Function	Comments
inittab/piobe	init	/etc/inittab	printer IO Back End (for printing)	• Handles the scheduling, spooling and printing of jobs submitted by the **qdaemon** • Disable if you are not printing from your system because you are sending print job to a server
inittab/qdaemon	init	/etc/inittab	queue daemon (for printing	• Submits print jobs to the **piobe** daemon • If you are not printing from your system, then disable
inittab/uprintfd	init	/etc/inittab	kernel messages	• Generally not required • Disable
inittab/writesrv	init	/etc/inittab	writing notes to ttys	• Only used by interactive UNIX workstation users • Disable this service for servers, back room databases, and development machines • Enable this service for workstations
inittab/xdm	init	/etc/inittab	traditional X11 Display Management	• Do not run on back room production or database servers • Do not run on development systems unless X11 display management is needed • Acceptable to run on workstations if graphics are needed
rc.nfs/automountd		/etc/rc.nfs	automatic file systems	• If you use NFS, enable this for workstations • Do not use the automounter for development or back room servers
rc.nfs/biod		/etc/rc.nfs	Block IO Daemon (required for NFS server)	• Enabled for NFS server only • If not an NFS server, then disable this along with **nfsd** and **rpc.mountd**

Service	Daemon	Started by	Function	Comments
rc.nfs/keyserv		/etc/rc.nfs	Secure RPC Key server	• Manages the keys required for secure RPC • Important for NIS+ • Disable this if you are *not* using NFS and NIS and NIS+
rc.nfs/nfsd		/etc/rc.nfs	NFS Services (required for NFS Server)	• Authentication is weak • Can lend itself to stack frame crashing • Enable if on NFS file servers • If you disable this, then disable **biod**, **nfsd**, and **rpc.mountd** as well
rc.nfs/rpc.lockd		/etc/rc.nfs	NFS file locks	• Disable if you are not using NFS • Disable this if you are not using file locks across the network • **lockd** daemon is mentioned in the SANS Top Ten Security Threats
rc.nfs/rpc.mountd		/etc/rc.nfs	NFS file mounts (required for NFS Server)	• Authentication is weak • Can lend itself to stack frame crashing • Only should be enabled on NFS file servers • If you disable this, then disable **biod** and **nfsd** as well
rc.nfs/rpc.statd		/etc/rc.nfs	NFS file locks (to recover them)	• Implements file locks across NFS • Disable unless you are using NFS
rc.nfs/rpc.yppasswdd		/etc/rc.nfs	NIS password daemon (for NIS master)	• Used to manipulate the local password file • Only required when the machine in question is the NIS master; disable in all other cases
rc.nfs/ypupdated		/etc/rc.nfs	NIS Update daemon (for NIS slave)	• Receives NIS database maps pushed from the NIS Master • Only required when the machine in question is a NIS slave to a Master NIS Server

Service	Daemon	Started by	Function	Comments
rc.tcpip/autoconf6		/etc/rc.tcpip	IPv6 interfaces	• Disable unless you are running IPV6
rc.tcpip/dhcpcd		/etc/rc.tcpip	Dynamic Host Configure Protocol (client)	• Back room servers should not rely on DHCP. Disable this service • If your host is not using DHCP, disable
rc.tcpip/dhcprd		/etc/rc.tcpip	Dynamic Host Configure Protocol (relay	• Grabs DHCP broadcasts and sends them to a server on another network • Duplicate of a service found on routers • Disable this if you are not using DHCP or rely on passing information between networks
rc.tcpip/dhcpsd		/etc/rc.tcpip	Dynamic Host Configure Protocol (server	• Answers DHCP requests from clients at boot time; gives client information, such as IP name, number, netmask, router, and broadcast address • Disable this if you are not using DHCP • Disabled on production and back room servers along with hosts not using DHCP
rc.tcpip/dpid2		/etc/rc.tcpip	outdated SNMP service	• Disable unless you need SNMP
rc.tcpip/gated		/etc.rc.tcpip	gated routing between interfaces	• Emulates router function • Disable this service and use RIP or a router instead
rc.tcpip/inetd		/etc/rc.tcpip	inetd services	• A thoroughly hardened system should have this disabled, but is often not practical • Disabling this will disable remote shell services which are required for some mail and web servers

Service	Daemon	Started by	Function	Comments
rc.tcpip/mrouted		/etc/rc.tcpip	multi-cast routing	• Emulates router function of sending multi-cast packets between network segments • Disable this service. Use a router instead
rc.tcpip/names		/etc/rc.tcpip	DNS name server	• Only use this if your machine is a DNS name server • Disable for workstation, development and production machines
rc.tcpip/ndp-host		/etc/rc.tcpip	IPv6 host	• Disable unless you use IPV6
rc.tcpip/ndp-router		/etc/rc.tcpip	IPv6 routing	• Disable this unless you use IPV6. Consider using a router instead of IPv6
rc.tcpip/portmap		/etc/rc.tcpip	RPC services	• Required service • RPC servers register with **portmap** daemon. Clients who need to locate RPC services ask the **portmap** daemon to tell them where a particular service is located • Disable only if you have managed to reduce RPC service so that the only one remaining is **portmap**
rc.tcpip/routed		/etc/rc.tcpip	RIP routing between interfaces	• Emulates router function • Disable if you have a router for packets between networks
rc.tcpip/rwhod		/etc/rc.tcpip	Remote ″who″ daemon	• Collects and broadcasts data to peer servers on the same network • Disable this service

Service	Daemon	Started by	Function	Comments
rc.tcpip/sendmail		/etc/rc.tcpip	mail services	• Runs as root user and therefore is dangerous • Has a long history of security breaches • Disable this service unless the machine is used as a mail server • If disabled, then do one of the following: – Place an entry in crontab to clear the queue. Use the **/usr/lib/sendmail -q** command – Configure DNS services so that the mail for your server is delivered to some other system
rc.tcpip/snmpd		/etc/rc.tcpip	Simple Network Management Protocol	• Disable if you are not monitoring the system via SNMP tools • SNMP may be required on important servers, but not likely on workstations
rc.tcpip/syslogd		/etc/rc.tcpip	system log of events	• Never disable this service • Prone to denial of service attacks • Required in any system
rc.tcpip/timed		/etc/rc.tcpip	Old Time Daemon	• Disable this service and use xntp instead
rc.tcpip/xntpd		/etc/rc.tcpip	New Time Daemon	• Keeps clocks on systems in sync • Disable this service. • Configure other systems as time servers and let other systems synchronize to them with a cron job that calls ntpdate
dt login		/usr/dt/config/Xaccess	unrestricted CDE	• If you are not providing CDE login to a group of X11 stations, you can restrict dtlogin to the console.

Service	Daemon	Started by	Function	Comments
anonymous FTP service		user rmuser -p <username>	anonymous ftp	• Anonymous FTP ability prevents you from tracing FTP usage to a specific user • Remove user ftp if that user account exists, as follows: **rmuser -p ftp** • Further security can be obtained by populating the **/etc/ftpusers** file with a list of those who should not be able to ftp to your system
anonymous FTP writes			anonymous ftp uploads	• No file should belong to ftp. • FTP anonymous uploads allow the potential for misbehaving code to be placed on your system. • Put the names of those users you want to disallow into the **/etc/ftpusers** file • Some examples of system-created users you might want to disallow from anonymously uploading via FTP to your system are: root, daemon, bin.sys, admin.uucp, guest, nobody, lpd, nuucp, ladp, imnadm • Change the owner and group rights to the **ftpusers** files as follows: **chown root:system /etc/ftpusers** • Change the permissions to the **ftpusers** files to a stricter setting as follows: **chmod 644 /etc/ftpusers**
ftp.restrict			ftp to system accounts	• No user from the outside should be allowed to replace root files via **ftpusers** file

Service	Daemon	Started by	Function	Comments
root.access		/etc/security/user	rlogin/telnet to root account	• Set the rlogin option in the **etc/security/user** file to false • Anyone logging in as root should first login under their own name and then **su** to root; this provides an audit trail
snmpd.readWrite		/etc/snmpd.conf	SNMP readWrite communities	• If you are *not* using SNMP, disable the SNMP daemon. • Disable community private and community system in the **/etc/snmpd.conf** file • Restrict 'public' community to those IP addresses that are monitoring your system
syslog.conf			configure syslogd	• If you have not configured **/etc/syslog.conf**, then disable this daemon • If you are using **syslog.conf** to log system messages, then keep enabled

Summary of Network Options

To achieve a higher level of system security, there are several network options that you can change using 0 to disable and 1 to enable.The following list identifies these parameters you can use with the **no** command.

Parameter	Command	Purpose
bcastping	/usr/sbin/no -o bcastping=0	Allows response to ICMP echo packets to the broadcast address. Disabling this prevents Smurf attacks.
clean_partial_conns	/usr/sbin/no -o clean_partial_conns=1	Specifies whether or not SYN (synchronizes the sequence number) attacks are being avoided.
directed_broadcast	/usr/sbin/no -o directed_broadcast=0	Specifies whether to allow a directed broadcast to a gateway. Setting to 0 helps prevent directed packets from reaching a remote network.
icmpaddressmask	/usr/sbin/no -o icmpaddressmask=0	Specifies whether the system responds to an ICMP address mask request. Disabling this prevents access through source routing attacks.
ipforwarding	/usr/sbin/no -o ipforwarding=0	Specifies whether the kernel should forward packets. Disabling this prevents redirected packets from reaching remote network.
ipignoreredirects	/usr/sbin/no -o ipignoreredirects=1	Specifies whether to process redirects that are received.
ipsendredirects	/usr/sbin/no -o ipsendredirects=0	Specifies whether the kernel should send redirect signals. Disabling this prevents redirected packets from reaching remote network.
ip6srcrouteforward	/usr/sbin/no -o ip6srcrouteforward=0	Specifies whether the system forwards source-routed IPv6 packets. Disabling this prevents access through source routing attacks.
ipsrcrouteforward	/usr/sbin/no -o ipsrcrouteforward=0	Specifies whether the system forwards source-routed packets. Disabling this prevents access through source routing attacks.
ipsrcrouterecv	/usr/sbin/no -o ipsrcrouterecv=0	Specifies whether the system accepts source-routed packets. Disabling this prevents access through source routing attacks
ipsrcroutesend	/usr/sbin/no -o ipsrcroutesend=0	Specifies whether applications can send source-routed packets. Disabling this prevents access through source routing attacks.

Parameter	Command	Purpose
nonlocsroute	/usr/sbin/no -o nonlocsrcroute=0	Tells the Internet Protocol that strictly source-routed packets may be addressed to hosts outside the local network. Disabling this prevents access through source routing attacks.
tcp_pmtu_discover	/usr/sbin/no -o tcp_pmtu_discover=0	Disabling this prevents access through source routing attacks.
udp_pmtu_discover	/usr/sbin/no -o udp_pmtu_discover=0	Enables or disables path MTU discovery for TCP applications. Disabling this prevents access through source routing attacks.

For more information about network tunable options, see *Performance Management Guide*.

Appendix. Notices

This document was produced in the United States. IBM may not offer the products, programs, services or features discussed herein in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the products, programs, services, and features available in your area. Any reference to an IBM product, program, service or feature is not intended to state or imply that only IBM's product, program, service or feature may be used. Any functionally equivalent product, program, service or feature that does not infringe on any of IBM's intellectual property rights may be used instead of the IBM product, program, service or feature.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed ″AS IS″. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. The use of this information or the implementation of any techniques described herein is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. Customers attempting to adapt these techniques to their own environments do so at their own risk.

IBM is not responsible for printing errors in this publication that result in pricing or information inaccuracies.

The information contained in this document represents the current views of IBM on the issues discussed as of the date of publication. IBM cannot guarantee the accuracy of any information presented after the date of publication.

Any performance data contained in this document was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally available systems. Some measurements quoted in this document may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

The following terms are trademarks of International Business Machines Corporation in the United States and/or other countries: AIX. A full list of U.S. trademarks owned by IBM can be found at http://iplswww.nas.ibm.com/wpts/trademarks/trademar.htm. UNIX is a registered trademark of The Open Group in the United States and other countries. Microsoft is a registered trademark of Microsoft Corporation in the United States, other countries, or both. Other company, product and service names may be trademarks or service marks of others.