Global Information Assurance Certification Paper

GIAC Certified UNIX Security Administrator (GCUX)
Practical Assignment Version 1.9 Option 1
AIX Version 5.l - Bastion Build Revisited

Abstract

This document will include a step-by-step process for a baseline build of AIX 5.1 with additional post-build steps to secure the system for a 3rd party FTP/web server. This document include a secure build based on the guidelines from IBM's "Strengthening AIX Security: A System-Hardening Approach". Security risks will be identified with the default OS configurations and what action was done to secure these risks. Lastly, there will be a discussion on a strategy for maintenance and monitoring that should be done to protect against possible attacks and compromises to the system. This document will not cover configuration details or vulnerabilities of the 3rd party application.

Description of the System

Installation was done on an IBM 7026 Model B80 symmetric multiprocessor (SMP) using a 1-way 375 MHz processor 64-bit, copper-based, POWER3-II microprocessors, 4MB of Level 2 (L2) and 512 MB of RAM memory. There are two 18.2 GB internal Enhanced 10 K rpm Ultra2 SCSI drives and a 4.7GB DVD-RAM in the available media bay.

The server will reside in a DMZ which clients will be able to able to push or pull data files via FTP or browser access. The FTP/web processes will run on Valicert's SecureTransport application that will be applied after the hardened AIX server is configured. SecureTransport bundles in a modified version of Apache 1.3.12 and a RFC compliant FTP protocol supporting SSL (see RFC 959, RFC 2228). It also includes its own installation of Perl 5.6.0. The system will run AIX version 5.1 with maintenance level 3 applied.

Risk Analysis of the System

The role of this host is to allow for only secure encrypted FTP and HTTP data traffic and SSH access within the DMZ. There will be no need for external mail services, network file sharing (NFS), NIS, r-services, and others that are turned on by default. In turn, the primary focus of this paper will be the steps taken to harden the system and allow the host to run only the needed services.

System based hardening is not a substitute for sound architectural planning. In this case, a DMZ network should always include correctly configured and placed firewalls and routers. Consult your networking team for assistance is this area. If you are the networking team and have little experience, its time to take some additional training classes in this area.

Obviously, a primary concern will be attacks against FTP, FTP/s, HTTP, and HTTP/s since all of the external connection will be made this way. There are the evident risks with using non-encrypted protocols such as FTP and HTTP. We will also look at the internal threats with authorized users that need to login to the system. SSH will help to solve the problem of passwords being sent in clear text on the wire. These issues will be discussed in more detail further into the paper.

The risk to our business if this server or application were to go down would be high since we run a 24x7 environment. Our contractual agreements with our customers require our ftp servers to be up available at all times to transmit data. There have been architecture improvements with redundant ftp servers which help to improve the chance that only one server would be affected at any one time and only hinder production. In a case where one of the ftp servers were to go down, this would buy us valuable time in finding the cause and fixing the problem before it could affect the other server if it hasn't already done so. The skinny is that down servers stop production to some degree, which ultimately can results in loss of revenue, affects relationships with our current customers and possibly new ones, and affect future growth. System hardening is one way to ensure known vulnerabilities can be secured. Lets start.

Step by Step Guide

Installing the Base Operating System
Loading AIX 5.1 from CD

You will need to acquire the base operating system disks which is *AIX 5L for POWER V5.1 5765-E61* as of 10/8/02. Later versions of the disks will already apply the patches to the OS as they are released. You can check with IBM at the time you request your CDs as to which patches are already applied. IBM will inform you of any additional patch CDs that will come separately. If you do not already have a copy, you can contact IBM at 1-800-879-2755, option 2, option 2.

Your server will have an installed version of the OS but it always a good idea to install a fresh version of the OS so you know what was installed on the server and can document the process for your organization.

On a normal day, when the stars are in alignment, the base installation and applying updates can take about 2 hours. Here are the steps to installing the secure build base operation system:

Bold type indicates commands and keystrokes
Italicize indicate system/file reference
Narrow letters indicates (headings) and screen captures

Installing CDs and Reboot

When the server is powered up, you can insert the AIX 5L, disk 1 of 5, into the CD-ROM drive and initiate the reboot sequence.
For PCI architecture, **#shutdown -Fr** For MCA architecture (ex. R-series), you can quicken the reboot by setting FAST IPL. Set the key to Maintenance mode, then **#mpcfg -cf 11 1** **#shutdown -Fr** or 1. Press the power button so the LCD reads stand-by mode. 2. Press **enter** on keyboard. You will get a ">" prompt. 3. Type **sbb** 4. Press **1** and **return** to set flags. 5. Press **x** and **return**. 6. Press **x** and **return**. 7. Press the power button and turn key to normal position. 8. Go to the keyboard and press **return**. The system will reboot.

Note: You will have to do this on every reboot because the system resets the FAST IPL to disable.
After you hear the beeps and see the memory keyboard network scsi adapter screen Press **5** (for graphics terminals, press **F5** – but you should not need graphical terminal, right?)

Installing BOS

At the ******* Please define the System Console ******* Type 1 and press Enter to use this terminal as the system console Press **1** and **return**.
At the Type 1 and press Enter to have English during install Press **1** and **return**.
At the Welcome to Base Operating System Installation and Maintenance 2 Change/Show Installation Setting and Install Press **2** and **return**.
At the Installation and Settings Press **1** and **return** to select System Settings:
At the Method of Installation Press **1** and **return** to select New and Complete Overwrite.
At the Change Disk(s) Where You Want to Install Follow the screen instructions for selecting hdisk0 for rootvg By default, hdisk0 is already setup. Disk mirroring should be setup for the root volume group at some point. This is not covered in this paper. Press **0** and **return** to complete in installation settings.
This will bring you back to Installation and Setting press **3** and **return** to select Advanced Options press the numbers and **return** to toggle the settings as follows: Select 1 for Installation Package Set .....**minimal** Select 2 for Enabled Trusted Computing Base.....**yes** Select 3 for Enabled 64-bit Kernel and JFS2.....no (MCA architecture will be no) NOTE: AIX V5.1 has the option to be 64-bit and JFS2 enabled. These are new

enhancements to AIX. I generally allow for several releases to pass before I implement bleeding edge technology. This allows for bugs to be fixed and applications to catch up to platform changes. Currently, there are applications that will not run on a 64-bit or JFS2 enabled operating system so you will need to check with the vendor on compatibility if you choose to enable this feature.

Here is some information and a recommendation sent out from Bruce Spencer from IBM Server Sales about the subject:

> When you install AIX 5, you can choose either
> a 32 or 64 bit kernel. In most cases, the choice
> isn't critical. Here are the similarities and differences.
>
> Similarities
>
> Both 32 and 64 bit kernel support 64 bit applications
> Both support JFS2 (large filesystems)
>
> Differences
>
> The 64 bit kernel supports over 96 GB of memory.
>
> My recommendation is to install the 32 bit kernel, unless
> you're using JFS2 or need to support over 96 GB memory.
> The 32 bit kernel has been around longer, and internal
> benchmarks show comparable performance to the 64
> bit kernel. On the other hand, I understand JFS2 runs
> better on the 64 bit kernel.

NOTE: JFS2 is an enhanced version of JFS that allows for files size to reach 1 terabyte and an “architectural maximum file system sizes of 4 petabytes” (Redbooks, Dec 2001).

Once complete,
Press **0** and **return**.

Again, at the Installation and Setting
Press **0** and **return** to begin installation.

You should see the following at the bottom of the screen if you have done it correctly

Approximate % task complete | Elapsed time (in minutes)

This takes approximately 25 minutes based on this particular hardware spec.

PCI based architecture will reboot automatically after completion of the installation. MCA based architecture will need to be rebooted manually after completion. If you have a command prompt, **#mpcfg -cf 11 1** **#shutdown -Fr**

After Rebooting and Using the Installation Assistant

At the Set Terminal Type Type ibm3151 (monochrome monitor) or which ever applies and press **return**. This terminal type is also used for the newer ibm3153 model.
At the Software License Agreements Select menu items Accept License Agreements → Accept License Agreement Press **Tab** to toggle the Entry Field to yes Press **return** After Command: 0K, Press **F10** key to exit.
At the Installation Assistant menu Select menu items Set Date and Time → Change / Show Date & Time Adjust the date or time. Absolute time may not necessary right at this moment. A time synchronization service will correct the time later. If you do not have a time synchronization service, looks like this is your next project. This not covered in this paper. Press **return** Press **F3** key twice to get back to Set Date and Time
At the Set Date and Time Select menu item Change Time Zone Using System Defined Values
At the Use DAYLIGHT SAVINGS TIME? Select 1 yes Press **return**.
At the CUT(Coordinated Universal Time) Time Zone Select your correct time zone. Press **return**.
At the Change Time Zone Press **return** to accept default settings.

F3 twice to get back to main menu.

At the Installation Assistant menu
Select Set root Password

Follow the instructions to set your root password.

At the Installation Assistant menu
Select menu items Configure Network Communications → TCP/IP startup → en0
You may choose another adapter at this time.

These are the configuration fields that need to be modified:

Hostname	[server_name]
Internet address	[###.###.###.###]
Network mask	[###.###.###.###]
Nameserver	
Internet Address	[###.###.###.###]
Domain name	[some.domain.com]
Default Gateway address	[###.###.###.###]
Start now	[yes]

After you have modified all the fields,

Press **return**
Press **F3** key three times twice to get back to Installation Assistant menu

At the Installation Assistant menu
Select menu item Manage System Storage & Paging Space (rootvg) → Add/Show Paging Space

NEW paging space (MB) [###]

Size paging space according to real memory based on IBM recommendations:

< or = 256 MB Total paging space = (memory size) x 2
> 256 MB Total paging space = 512 MB + (memory size – 256 MB) * 1.25

After completed, press **return**.
Press **F3** key three times twice to get back to Installation Assistant menu

If you need to have a non-root level account local to the system, you can do this by selecting menu item Create Users

If not, skip to the next step.

At the Installation Assistant
Select Task Completed – Exit to Login

Setting the Terminal Type

At the login prompt Login:
Login into the server as root or "su -" to root.
You should now have a root prompt.
#

Set the TERM settings to ibm3151
#export TERM=ibm3151
#smitty

Select menu items Devices → TTY → Change / Show Characteristics of a TTY

or

To skip the menu items, you can use the "fastpath". You can find the "fastpath" at any point in the menu by pressing the **F8** key.
#smitty chgtty

From the TTY pop-up screen, select
tty0 Available ##-##-##-## Asynchronous Terminal

Press **return**.

Arrow down to the configuration field and enter ibm3151 into the entry field:
TERMINAL type [ibm3151]

Press **return**.
Press **F10** to exit.

Applying Latest Patches to AIX 5L

Load the Update CD in the media tray.

From the SMIT menu
Select menu items Software Installation and Maintenance → Install and Update Software → Update Installed Software to Latest Level (Update All)

or

#smitty update_all

Pressing the **F4** key will show you the available input devices. This is the configuration field you will modify:
INPUT device / directory for software [/dev/cd0]

Press **return**.

Arrow down to Preview Only? and press **Tab** key to toggle no to yes

Preview Only? [yes]

Press **return**.

ARE YOU SURE?

Press **return**.

If executes cleanly,
Press **F3**.

If preview Failed, you can troubleshoot the failures, call IBM software support, or reinstall. After a successful or failed completion, you can view the complete screen output by pressing **Ctrl-V** keys to move down and **Ctrl-6** to move up.

Press **Tab** key again to Preview only? to no

Preview Only? [no]

Press **return**.

ARE YOU SURE?

For multiple volume patches, you will be asked to insert volumes from the update at specific times.

Press **return**.
After successful completion, press **F10** to get back to a command prompt.

Reboot the System

You will have to reboot the system to update the *bosboot*, rebuild the kernel, and lay down the updates.

MCA Architecture (R-Series)
#mpcfg -cf 11 1

#shutdown -Fr

PCI Architecture
#shutdown -Fr

Reboot time: 10 minutes

You can review all system modifications done thru *smitty* by looking in the *smit.log* file created in root's home directory /.

Now that a default AIX OS has been installed on the system, lets verify the OS and patch level.

```
# instfix -ivq | grep AIX_ML
5.0.0.0_AIX_ML Abstract: AIX 5.0.0.0 Release
5.1.0.0_AIX_ML Abstract: AIX 5.1.0.0 Release
5.1.0.0_AIX_ML Abstract: AIX 5.1.0.0 Release
5100-01_AIX_ML Abstract: AIX 5100-01 Update
5100-02_AIX_ML Abstract: AIX 5100-02 Update
5100-03_AIX_ML Abstract: AIX 5100-03 Update
```

or

```
# instfix -i | grep AIX_ML
    All filesets for 5.0.0.0_AIX_ML were found.
    All filesets for 5.1.0.0_AIX_ML were found.
    All filesets for 5.1.0.0_AIX_ML were found.
    All filesets for 5100-01_AIX_ML were found.
    All filesets for 5100-02_AIX_ML were found.
    All filesets for 5100-03_AIX_ML were found.
```

Verify 32 bit kernel

```
#bootinfo -K
32
```

The Before

Here are the initial outputs from a *netstat* and a *ps* before we start hardening the system. This is a snapshot of the TCP/UDP ports opened by default.

```
# netstat -an | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address        (state)
tcp4       0      0  *.13                *.*                    LISTEN
tcp        0      0  *.21                *.*                    LISTEN
tcp4       0      0  *.22                *.*                    LISTEN
tcp        0      0  *.23                *.*                    LISTEN
tcp4       0      0  *.25                *.*                    LISTEN
tcp4       0      0  *.37                *.*                    LISTEN
tcp4       0      0  *.111               *.*                    LISTEN
tcp4       0      0  *.199               *.*                    LISTEN
tcp        0      0  *.512               *.*                    LISTEN
tcp        0      0  *.513               *.*                    LISTEN
```

```
tcp        0      0  *.514              *.*                    LISTEN
tcp4       0      0  *.657              *.*                    LISTEN
tcp4       0      0  *.1334             *.*                    LISTEN
tcp4       0      0  127.0.0.1.49213    *.*                    LISTEN
tcp4       0      0  *.32769            *.*                    LISTEN
tcp4       0      0  *.32771            *.*                    LISTEN
tcp4       0      0  *.32772            *.*                    LISTEN
tcp4       0      0  x.x.x.x.32769      x.x.x.x.32770        ESTABLISHED
tcp4       0      0  x.x.x.x.32770      x.x.x.x.32769        ESTABLISHED
tcp4       0      0  127.0.0.1.199      127.0.0.1.32768      ESTABLISHED
tcp4       0      0  127.0.0.1.32768    127.0.0.1.199        ESTABLISHED
udp4       0      0  *.13                    *.*
udp4       0      0  *.37                    *.*
udp4       0      0  *.111                   *.*
udp4       0      0  *.161                   *.*
udp4       0      0  *.514                   *.*
udp4       0      0  *.518                   *.*
udp4       0      0  *.32776                 *.*
udp4       0      0  *.32794                 *.*
udp4       0      0  *.32845                 *.*
```

This is a snapshot of the processes running by default.

```
# ps -ef
    UID   PID  PPID   C    STIME    TTY  TIME CMD
    root     1     0   0   Dec 20      -  0:11 /etc/init
    root  2802     1   0   Dec 20      -  0:00 /usr/ccs/bin/shlap
    root  3990     1   0   Dec 20      -  0:00 /usr/sbin/srcmstr
    root  4190     1   0   Dec 20      -  1:25 /usr/sbin/syncd 60
    root  4610     1   0   Dec 20      -  0:00 /usr/lib/errdemon
    root  4936  3990   0   Dec 20      -  0:00 /usr/sbin/syslogd
    root  5302     1   0   Dec 20      -  0:02 /usr/sbin/cron
    root  5424  3990   0   Dec 20      -  0:10 /usr/sbin/portmap
    root  5686  3990   0   Dec 20      -  0:06 sendmail: accepting
connections
    root  5934  3990   0   Dec 20      -  0:00 /usr/sbin/inetd
    root  6192  3990   0   Dec 20      -  0:05 /usr/sbin/snmpd
    root  6450  3990   0   Dec 20      -  0:00 /usr/sbin/dpid2
    root  6708  3990   0   Dec 20      -  0:01 /usr/sbin/hostmibd
  daemon  7746  3990   0   Dec 20      -  0:00 /usr/sbin/rpc.statd
    root  8002  3990   0   Dec 20      -  0:00 /usr/sbin/biod 6
    root  8264  3990   0   Dec 20      -  0:00 /usr/sbin/rpc.lockd
    root  8522     1   0   Dec 26      -  0:00 /usr/sbin/getty
/dev/console
    root  9038     1   0   Dec 20      -  0:00 /usr/sbin/uprintfd
    root  9340  3990   0   Dec 20      -  0:00 /usr/sbin/qdaemon
    root  9556  3990   0   Dec 20      -  0:00 /usr/sbin/writesrv
    root  9810     1   0   Dec 20      -  0:00
/usr/lpp/diagnostics/bin/diagd
    root 10324     1   0   Dec 20      -  0:00
/usr/bin/AIXPowerMgtDaemon
    root 11094  3990   0   Dec 20      -  0:03 /usr/sbin/rsct/bin/rmcd
-r
  imnadm 11352     1   0   Dec 20      -  0:00
/usr/IMNSearch/httpdlite/httpdlite -r
```

```
/etc/IMNSearch/httpdlite/httpdlite.conf
    root 11870  3990   0   Dec 20      -  0:00
/usr/sbin/rsct/bin/ctcasd
    root 13160  3990   0   Dec 20      -  0:01
/usr/sbin/rsct/bin/IBM.ERrmd
    root 13420  3990   0   Dec 20      -  0:04
/usr/sbin/rsct/bin/IBM.CSMAgentRMd
    root 14192  3990   0   Dec 20      -  0:01
/usr/sbin/rsct/bin/IBM.ServiceRMd
```

Caveat

The ultimate goal is to enable only the needed services to run a secure base operating system for our FTP/HTTP services. For non-privileged ports >1024, I'll identify the port function and then decide if it could be safely shutdown without affecting critical components needed to run the system. Processes, based on a **ps –ef** output, will be identified and disabled if they are not critical to the application or the system.

Begin System Hardening

SSH

Installing SSH is an important fundamental step in system and network security. Telnet, ftp, or r-services are easily vulnerable to obtaining username and password information plus other sensitive data. The data is sent in clear text over the wire. Anyone having access to the network can capture these packets and read targeted character strings.

There are many articles on these types of vulnerabilities and exploits. You can use any search engine on the Internet and do a search for these key words and the type of service. This is why using encryption is a must. If you are in the business of giving out this information and allowing someone outside or within your organization compromising your servers, you can skip this section.

Some data centers have a policy that DMZ or other sensitive servers do not allow for remote administration. If the only way to login onto the machine is to physically go to the console, then you will want to disable all remote login capability.

With AIX5.1, all that is needed to get SSH working on AIX are 2 packages:

- LLP package *openssh34p1_51.tar.Z*
- RPM package *openssl-0.9.6e-1.aix4.3.ppc.rpm*

Why both these packages could not both be in either LLP or RPM format, IBM technical support could not state. Here are some helpful links to download the packages.

OpenSSH download website:
http://www-124.ibm.com/developerworks/downloads/index.php?group_id=108

OpenSSl download website (you will have to register to obtain access, it's free):
https://www6.software.ibm.com/dl/aixtbx/aixtbx-i?S_PKG=dlaixww&S_TACT=&S_CMP

Contact IBM at 1-800-879-2755, option 2, option 2 to obtain media.
OpenSSH/OpenSSL media:

- AIX Toolbox for Linux Applications for POWER Systems CD
- AIX 5.1 Bonus Pack CD starting in April 2002
- Linux Toolbox CD

Since I'm protecting the system from the network, I'll have to make sure SSH is running so I can access the system once on the network since other remote login protocols will be disabled (telnet, r-services, ftp, etc.) Therefore, I need the SSH and SSL packages on CDs so I can install these services. If you are going to download and burn the files to a CD (or other form of media), the OpenSSH package from IBM, make sure to run the *inutoc* command to create the *.toc* file prior to burning it to CD so that *smitty* can recognize the contents of the LLP package. Make sure to remove any previous *.toc* files in the directory before you run the *inutoc* command.

Per IBM technical support, the *prngd* (pseudo random number generator daemon) binaries are incorporated into the LPP package of OpenSSH. IBM recommends not installing *prngd* on AIX 5.1. This is an excerpt from an IBM article that was sent me by technical support:

> OpenSSH on AIX 5.1 is compiled using the entropy gathering mechanism (random numbers) provided with the OpenSSH source code (`ssh-rand-helper`), as opposed to AIX 4.3.3 (AIX Linux Toolbox) which uses the `PRNGD` open source daemon (`prngd-0.9.23-3.aix4.3.ppc.rpm package`) (prngd, 2002?).

The complete document can be found in the References page and the end of this paper.

Installing SSH

As of 12/13/2002, I have not found, nor could IBM technical support direct me to, an authorized published document from IBM on how to install OpenSSH via *installp* or OpenSSL via RPM. However, IBM technical support did send me an internal document which details their recommended install instructions for

OpenSSH. I've added this document to the end of the paper as Appendix A. This is the paraphrased version that IBM recommends:

Install the OpenSSL RPM package first. The default install of the AIX V5.1 include the RedHat Package Manager (RPM) LPPs. You will need this to open RPM commands.

#rpm -i openssl-0.9.6e-1.aix4.3.ppc.rpm

Verify installation
#rpm -aq | grep openssl

Then install the OpenSSH packages via *smitty*.

#smitty install_latest

Make sure and toggle the "yes" field to accept the license before you install to avoid failure.

```
                              Install Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* INPUT device / directory for software               /dev/cd0
* SOFTWARE to install                                [_all_latest]
  PREVIEW only? (install operation will NOT occur)    no
  COMMIT software updates?                            yes
+
  SAVE replaced files?                                no
  AUTOMATICALLY install requisite software?           yes
  EXTEND file systems if space needed?                yes
  OVERWRITE same or newer versions?                   no
  VERIFY install and check file sizes?                no
  Include corresponding LANGUAGE filesets?            yes
  DETAILED output?                                    no
  Process multiple volumes?                           yes
  ACCEPT new license agreements?                      yes
  Preview new LICENSE agreements?                     no
```

Verify Installation
#lslpp -l | grep ssh

NOTE: SSH installs the start/stop scripts in */etc/rc.d/rc2.d* directory therefore is controlled by the entry *l2:2:wait:/etc/rc.d/rc 2* in */etc/inittab.*

Start SSH
#startsrc -g ssh

Reboot the system
#shutdown -Fr

System Hardening - /etc/inetd.conf
After looking at the contents of the *inetd.conf* file, IBM had done some of the hardening work for you. They have commented out a number of the services. Lets take this one step further. I move and secure the file with its original contents with an ".orig" extension. At any point *inetd* services maybe called for, a simple copy of lines can be done back from the original file.

My feeling is there should not be anything running out of *inetd* on this base system so I disable *inetd* completely. I'm not going into detail about the 37 services that are going to be disabled thru *inetd* but you can reference the risk assessment of these services in the AIX "Strengthening AIX Security" document starting on page 23. Also, this goes for the additional services in the *inittab, rc.tcpip,* and *rc.nfs* configuration files that are going to be disabled.

I will have a copy of the original file if I ever need to enable any services from *inetd.* As a general rule, all disabled lines are removed due to possible root toolkits looking for lines commented out in configuration files and un-commenting them for exploits.

Here is the assessment of the processes started by the *inetd.conf* file that do not need to run for our system and is not a dependency of our application.

inetd entry	associated process	brief description
daytime	/usr/sbin/inetd	obsolete time service
ftp	/usr/sbin/inetd	file tranfer protocol; using Valicert
telnet	/usr/sbin/inetd	telnet service; use SSH
time	/usr/sbin/inetd	obsolete time service
exec	/usr/sbin/inetd	remote execution service
login	/usr/sbin/inetd	rlogin services; use SSH
shell	/usr/sbin/inetd	rsh service; use SSH
ntalk	/usr/sbin/inetd	"new talk" for interactive chat sessions

#mv /etc/inetd.conf /etc/inetd.orig
#chmod 000 /etc/inetd.orig

Strict "000" permissions are given to the file so no one could have access to the file except for root. No modifications should be made to the ".orig" files. Reboot after modifications to */etc/inetd.conf* file. Here are the outputs:

```
# netstat -an | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address        (state)
tcp4       0      0  *.22               *.*                    LISTEN
```

```
tcp4       0      0  *.25                 *.*                    LISTEN
tcp4       0      0  *.111                *.*                    LISTEN
tcp4       0      0  *.199                *.*                    LISTEN
tcp4       0      0  *.657                *.*                    LISTEN
tcp4       0      0  *.1334               *.*                    LISTEN
tcp4       0      0  127.0.0.1.49213      *.*                    LISTEN
tcp4       0      0  *.32769              *.*                    LISTEN
tcp4       0      0  *.32771              *.*                    LISTEN
tcp4       0      0  *.32772              *.*                    LISTEN
tcp4       0      0  x.x.x.x.32769        x.x.x.x.32770      ESTABLISHED
tcp4       0      0  x.x.x.x.32770        x.x.x.x.32769      ESTABLISHED
tcp4       0      0  127.0.0.1.199        127.0.0.1.32768    ESTABLISHED
tcp4       0      0  127.0.0.1.32768      127.0.0.1.199      ESTABLISHED
udp4       0      0  *.111                     *.*
udp4       0      0  *.161                     *.*
udp4       0      0  *.514                     *.*
udp4       0      0  *.32776                   *.*
udp4       0      0  *.32794                   *.*
udp4       0      0  *.32845                   *.*
```

These are the ports shutdown from disabling *inetd*.
tcp ports:
13 -daytime
21 -ftp
23 -telnet
37 -time
512 -exec
513 -login
514 -shell

udp ports:
13 -daytime
7 -time
518 -ntalk

```
# ps -ef
     UID   PID  PPID   C    STIME    TTY  TIME CMD
    root     1     0   0 22:57:42      -  0:00 /etc/init
    root  2812     1   0 23:02:27      -  0:00 /usr/ccs/bin/shlap
    root  3804     1   0 23:02:28      -  0:00 /usr/sbin/srcmstr
    root  4190     1   0 23:02:26      -  0:00 /usr/sbin/syncd 60
    root  4718  3804   0 23:02:31      -  0:00 /usr/sbin/syslogd
    root  5004     1   0 23:02:26      -  0:00 /usr/lib/errdemon
    root  5302     1   0 23:03:00      -  0:00 /usr/sbin/cron
    root  5432  3804   0 23:02:35      -  0:00 sendmail: accepting
connections
    root  5678  3804   0 23:02:38      -  0:00 /usr/sbin/portmap
    root  5934  3804   0 23:02:41      -  0:00 /usr/sbin/inetd
    root  6192  3804   0 23:02:44      -  0:00 /usr/sbin/snmpd
    root  6450  3804   0 23:02:47      -  0:00 /usr/sbin/dpid2
    root  6708  3804   0 23:02:50      -  0:00 /usr/sbin/hostmibd
    root  7744  3804   0 23:02:53      -  0:00 /usr/sbin/biod 6
    root  8008  3804   0 23:03:00      -  0:00 /usr/sbin/rpc.lockd
```

```
   daemon  8260  3804   0 23:02:57      -  0:00 /usr/sbin/rpc.statd
     root  8520     1   0 23:03:00      0  0:00 /usr/sbin/getty
/dev/console
     root  9038     1   0 23:03:07      -  0:00 /usr/sbin/uprintfd
     root  9340  3804   0 23:03:03      -  0:00 /usr/sbin/qdaemon
     root  9556  3804   0 23:03:07      -  0:00 /usr/sbin/writesrv
     root  9810     1   0 23:03:10      -  0:00
/usr/lpp/diagnostics/bin/diagd
     root  8568  3804   0 23:03:10     -   0:00 /usr/sbin/sshd -D
     root 10068     1   0 23:03:07      -  0:00
/usr/bin/AIXPowerMgtDaemon
     root 11094  3804   0 23:03:10      -  0:00 /usr/sbin/rsct/bin/rmcd
-r
   imnadm 11352     1   0 23:03:10      -  0:00
/usr/IMNSearch/httpdlite/httpdlite -r
/etc/IMNSearch/httpdlite/httpdlite.conf
     root 11870  3804   0 23:03:10      -  0:00
/usr/sbin/rsct/bin/ctcasd
     root 12398  3804   0 23:03:13      -  0:00
/usr/sbin/rsct/bin/IBM.ERrmd
     root 12644  3804   0 23:03:13      -  0:00
/usr/sbin/rsct/bin/IBM.ServiceRMd
     root 14192  3804   0 23:03:11      -  0:00
/usr/sbin/rsct/bin/IBM.CSMAgentRMd
```

You may notice that the */usr/sbin/inetd* process is still running. If a service is need from the *inetd.conf* file, *inetd* will fork a process for that service and terminate the service once complete. The startup of *inetd* is controlled by the *rc.tcpip* script thru the */usr/sbin/srcmstr* process. Once we disable the entry in the *rc.tcpip* file, *inetd* will no longer run. You can stop *inetd* using the SRC command:

#stopsrc -s inetd

Even if it did run, the *inetd.conf* file would have no services to start up since we either deleted all the service lines out of the file or removed the file completely.

System Hardening - /etc/inittab

The next step is to edit the */etc/inittab* file. A copy of the original file was created with permissions set to 000.

#cp -p /etc/inittab /etc/inittab.orig
#chmod 000 /etc/inittab.orig
#vi /etc/inittab

Again as a general rule, all disabled lines are removed due to possible root toolkits. These are the lines taken out:

```
powerfail::powerfail:/etc/rc.powerfail 2>&1 | alog -tboot >
/dev/console # Power Failure Detection
```

```
load64bit:2:wait:/etc/methods/cfg64 >/dev/console 2>&1 # Enable 64-bit
execs
fbcheck:23456789:wait:/usr/sbin/fbcheck 2>&1 | alog -tboot >
/dev/console # run /etc/firstboot
rcnfs:23456789:wait:/etc/rc.nfs > /dev/console 2>&1 # Start NFS Daemons
piobe:2:wait:/usr/lib/lpd/pio/etc/pioinit >/dev/null 2>&1  # pb cleanup
qdaemon:23456789:wait:/usr/bin/startsrc -sqdaemon
writesrv:23456789:wait:/usr/bin/startsrc –swritesrv
uprintfd:23456789:respawn:/usr/sbin/uprintfd
shdaemon:2:off:/usr/sbin/shdaemon >/dev/console 2>&1 # High
availability daemon
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6
l7:7:wait:/etc/rc.d/rc 7
l8:8:wait:/etc/rc.d/rc 8
l9:9:wait:/etc/rc.d/rc 9
ctrmc:2:once:/usr/bin/startsrc -s ctrmc > /dev/console 2>&1
pmd:2:wait:/usr/bin/pmd > /dev/console 2>&1 # Start PM daemon
httpdlite:23456789:once:/usr/IMNSearch/httpdlite/httpdlite -r
/etc/IMNSearch/httpdlite/httpdlite.conf & >/dev/console 2>&1
```

The */etc/rc.d/rc* lines were removed because these files are initially empty. Post-build applications may require the system to write their boot scripts in these files. Example, OpenSSH installs its start/stop scripts in the */etc/rc.d/rc2.d* file. These files correspond to the related *inittab* entries and are run during the boot up sequence when *inittab* is called. Therefore, the corresponding line for */etc/rc.d/rc2.d* were included in the modified *inittab* file. Some people may argue to put the SSH boot scripts directly in the *inittab.* At this time, I cannot make a critical judgment on if one method is better than the other. You will have to weigh the arguments for yourself.

System administrators will have to be conscious of the type of application they are installing and whether or not it needs to be started at boot time. This kind of system awareness can only help you become a better system administrator.

The service lines from this */etc/inittab* file were removed because they were either not being used or the full understanding of the service was not understood and thus would not be fully utilized. This also poses a security risk when services are run on the system that no one understands. Therefore, it goes back to the golden rule, “if you are not going to use it, turn it off”. For additional help in this area, you can call IBM for software support.

Inittab entry	associated process	Brief Description
powerfail	no associated process	power failure detection on hardware; signals to *init*
load64bit	/usr/ccs/bin/shlap	for 64 bit kernal; computes for 64bit

		addresses
fbcheck		view */usr/sbin/fbcheck* for brief description of what it does. Looks for the */etc/firstboot* file for changes to the system environment or any customization; associated with NIM installs. If there is no */etc/firstboot* file, this will never run.
rcnfs	biod 6, rpc.statd, rpc.lockd	network file sharing(NFS) processes
piobe	no associated process	printing function
qdaemon	/usr/sbin/qdaemon	printing function
writesrv	/usr/sbin/writesrv	messaging from print subsystem 1334
shdaemon	no associated process	system hang detection mechanism; look in Redbook 5.2 Defenses book for more information.
ctrmc	/usr/sbin/rsct/bin/rmcd ~/ctcasd ~/IBM.Errmd ~/IBM.ServiceRMd ~/IBM.CSMAgentRMd	clustering for SP/Regatta systems; The *ctcasd* is *ctrmc* authentication daemon.
pmd	/usr/bin/AIXPowerMgtDaemon	power management daemon
httpdlite	/usr/IMNSearch/httpdlite /httpdlite -r /etc /IMNSearch/httpdlite /httpdlite.conf	IMN search engine; used for Documentation Library Search

Reboot after modifications to */etc/inittab* file. Here are the outputs:

```
# netstat -an |more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address     Foreign Address        (state)
tcp4       0      0  *.22              *.*                    LISTEN
tcp4       0      0  *.25              *.*                    LISTEN
tcp4       0      0  *.111             *.*                    LISTEN
tcp4       0      0  *.199             *.*                    LISTEN
tcp4       0      0  *.32769           *.*                    LISTEN
tcp4       0      0  x.x.x.x.32769     x.x.x.x.32770        ESTABLISHED
tcp4       0      0  x.x.x.x.32770     x.x.x.x.32769        ESTABLISHED
```

```
tcp4       0      0  127.0.0.1.199      127.0.0.1.32768      ESTABLISHED
tcp4       0      0  127.0.0.1.32768    127.0.0.1.199        ESTABLISHED
tcp        0      0  x.x.x.x.32771      x.x.x.x.25           TIME_WAIT
udp4       0      0  *.111                   *.*
udp4       0      0  *.161                   *.*
udp4       0      0  *.514                   *.*
udp4       0      0  *.32776                 *.*
```

These are the ports shutdown from modifying *inittab*.
tcp ports:
1334 –writesrv
32771 –ctmc
32772 –ctmc

udp ports:
32794 –ctmc
32845 –ctmc

```
# ps -ef
     UID   PID PPID   C    STIME    TTY  TIME CMD
    root     1    0   0 23:16:07      -  0:00 /etc/init
    root  2972    1   0 23:21:15      0  0:00 /usr/sbin/getty
/dev/console
    root  3902    1   0 23:20:51      -  0:00 /usr/lib/errdemon
    root  4190    1   0 23:20:51      -  0:00 /usr/sbin/syncd 60
    root  4460    1   0 23:20:52      -  0:00 /usr/sbin/srcmstr
    root  4744 4460   0 23:20:59      -  0:00 sendmail: accepting
connections
    root  5230 4460   0 23:20:56      -  0:00 /usr/sbin/syslogd
    root  5432    1   0 23:21:15      -  0:00 /usr/sbin/cron
    root  5678 4460   0 23:21:02      -  0:00 /usr/sbin/portmap
    root  5934 4460   0 23:21:05      -  0:00 /usr/sbin/inetd
    root  6192 4460   0 23:21:08      -  0:00 /usr/sbin/snmpd
    root  6450 4460   0 23:21:11      -  0:00 /usr/sbin/dpid2
    root  6708 4460   0 23:21:15      -  0:00 /usr/sbin/hostmibd
    root  6972    1   0 23:21:15      -  0:00
/usr/lpp/diagnostics/bin/diagd
    root  8568 3804   0 23:03:10     -   0:00 /usr/sbin/sshd -D
```

These are the processes disabled by modifying *inittab*.
/usr/sbin/biod 6
/usr/sbin/rpc.lockd
/usr/sbin/rpc.statd
/usr/sbin/uprintfd
/usr/sbin/qdaemon
/usr/sbin/writesrv
/usr/bin/AIXPowerMgtDaemon
/usr/sbin/rsct/bin/rmcd –r
/usr/IMNSearch/httpdlite/httpdlite /etc/IMNSearch/httpdlite/httpdlite.conf
/usr/sbin/rsct/bin/ctcasd
/usr/sbin/rsct/bin/IBM.Ermd

/usr/sbin/rsct/bin/IBM.ServiceRMd
/usr/sbin/rsct/bin/IBM.CSMAgentRMd

Almost half of these are IBM proprietary process that are not need with an FTP or web server. In a lot of case, you may not need these processes running. Consult IBM Redbooks 5.1, 5.2, and technical support for more detailed information on the usefulness of them.

System Hardening - /etc/rc.tcpip
The next step is to edit the */etc/rc.tcpip* file. Again, a copy of the original file was created with permissions set to 000. Again as a general rule, all disabled lines are removed.

#cp -p /etc/rc.tcpip /etc/rc.tcpip.orig
#chmod 000 /etc/rc.tcpip.orig
#vi /etc/rc.tcpip

These are the lines removed from */etc/rc.tcpip* file:

```
#start /usr/sbin/dhcpcd "$src_running"
#start /usr/sbin/autoconf6 ""
#start /usr/sbin/ndpd-host "$src_running"
#start /usr/sbin/ndpd-router "$src_running"
#start /usr/sbin/lpd "$src_running"
#start /usr/sbin/routed "$src_running" -q
#start /usr/sbin/gated "$src_running"
qpi=30m  # 30 minute interval
start /usr/lib/sendmail "$src_running" "-bd -q${qpi}"
start /usr/sbin/portmap "$src_running"
start /usr/sbin/inetd "$src_running"
start /usr/sbin/named "$src_running"
#start /usr/sbin/timed "$src_running"
#start /usr/sbin/xntpd "$src_running"
#start /usr/sbin/rwhod "$src_running"
start /usr/sbin/snmpd "$src_running"
#start /usr/sbin/dhcpsd "$src_running"
#start /usr/sbin/dhcprd "$src_running"
start /usr/sbin/dpid2 "$src_running"
start /usr/sbin/hostmibd "$src_running"
#start /usr/sbin/mrouted "$src_running"
#start /usr/sbin/pxed "$src_running"
#start /usr/sbin/binld "$src_running"
```

Here is the assessment of the processes started by the *rc.tcpip* file that do not need to run for our system and is not a dependency of our application.

tcpip entry	associated process	brief description
sendmail	sendmail:	mail services
portmap	/usr/sbin/portmap	RPC services

inetd	/usr/sbin/inetd	*inetd* services
snmpd	/usr/sbin/snmpd	simple network mgt. protocol daemon
dpid2	/usr/sbin/dpid2	outdated SNMP service
hostmibd	/usr/sbin/hostmibd	dpi2 sub-agent daemon associated with SNMP and Management Information Base (MIB)

Since I've determined this is not a sendmail server, I will need to add a line to the *crontab* to regularly flush stranded messages in the sendmail queue. AIX puts it in /usr/sbin**.** The syntax for the *crontab* line is as follows:

minute hour day_of_month month weekday command

In my case, I'm running the sendmail command every hour at 23 minutes past.

#crontab -e
23 * * * * /usr/sbin/sendmail -q

Reboot after modifications to the */etc/rc.tcpip* file. Here are the outputs:

```
# netstat -an | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address        (state)
tcp4       0      0  *.22                 *.*                    LISTEN
tcp4       0      0  x.x.x.x.22           x.x.x.x.1062       ESTABLISHED
udp4       0      0  *.514                *.*
```

These are the ports shutdown from modifying *rc.tcpip*.
tcp ports:
111 –portmapper
199 -SNMP Unix Multiplexer (smux)
32769 –inetd service

udp port:
111 –portmapper
161 –SNMP
32776 –inetd service

```
# ps -ef
     UID   PID PPID   C    STIME    TTY  TIME CMD
    root     1    0   0 00:18:09      -  0:00 /etc/init
    root  2574    1   0 00:22:49      -  0:00 /usr/sbin/syncd 60
    root  3460    1   0 00:22:58      0  0:00 /usr/sbin/getty
/dev/console
    root  3652    1   0 00:22:49      -  0:00 /usr/lib/errdemon
```

```
    root 4038    1   0 00:22:51      -  0:00 /usr/sbin/srcmstr
    root 5186 4038   0 00:22:54      -  0:00 /usr/sbin/syslogd
    root 5426    1   0 00:22:58      -  0:00 /usr/sbin/cron
    root 5682    1   0 00:22:58      -  0:00
/usr/lpp/diagnostics/bin/diagd
    root 6539 6104   0 00:23:43      -  0:00 /usr/sbin/sshd -D
```

These are the process disabled by modifying *rc.tcpip*:
sendmail: accepting connections
/usr/sbin/portmap
/usr/sbin/inetd
/usr/sbin/snmpd
/usr/sbin/dpid2
/usr/sbin/hostmibd

System Hardening /etc/rc.nfs
No *nfs* services need to be running for a FTP/web server in a DMZ. Again, I move and secure the file with its original contents. At any point *nfs* services are called for, a simple copy can be done back in the original name.

#mv /etc/rc.nfs /etc/rc.nfs.orig
#chmod 000 /etc/rc.nfs.orig

Reboot after moving the */etc/rc.nfs* file. The associated processes were disabled with the modifications to the *inittab* file. The three processes are biod 6, rpc.statd, and rpc.lockd. Changing the file name with the ".orig" extension ensures that inittab or any other script cannot call the startup file. The 000 permission help to insure the file will not be tampered by anyone but root.

The After
This is the output of all the processes and services running after hardening.

```
# netstat -an | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address       (state)
tcp4  0      0       *.22               *.*                    LISTEN
tcp4  0      0       x.x.x.x.22         x.x.x.x.12563      ESTABLISHED
udp4  0      0       *.514              *.*
```

```
# ps -ef
     UID   PID PPID   C    STIME    TTY  TIME CMD
    root     1    0   0 00:29:57      -  0:00 /etc/init
    root  2952    1   0 00:34:46      0  0:00 /usr/sbin/getty
/dev/console
    root  3648    1   0 00:34:37      -  0:00 /usr/lib/errdemon
    root  4190    1   0 00:34:37      -  0:00 /usr/sbin/syncd 60
    root  4786    1   0 00:34:39      -  0:00 /usr/sbin/srcmstr
    root  5186 4786   0 00:34:42      -  0:00 /usr/sbin/syslogd
```

```
    root 5426    1   0 00:34:46      -  0:00 /usr/sbin/cron
    root 5682    1   0 00:34:46      -  0:00
/usr/lpp/diagnostics/bin/diagd
    root 6204 4786   0 00:35:32      -  0:00 /usr/sbin/sshd -D
    root 6498 6204   0 00:37:34      -  0:00 /usr/sbin/sshd -D
    root 7024 6498   0 00:37:34  pts/0  0:00 -ksh
```

The final output shows 2 ports enabled. One for SSH connections and the other for *syslogd* which handles system logging. We can also see the established secure remote connection from my desktop to the server using local port 22. The process listing is a trimmer output compared to the default version.

The process listing is only what is needed for our systems and applications to function properly and securely.

Time needed to reboot the default system: 14 minutes
Time needed to reboot a minimal package system: 10 minutes
Time needed to reboot the hardened system: 7 minutes

Modifying /etc/syslog.conf

This can be a touchy subject on what should be reported in the logs. You can get way too much information or too little depending on what type of messages you want to receive from the system. What I'm about to suggest is some basic reporting. You possibly will have to do some homework on changes to the granularity of the logging may be useful to you and log those accordingly. Keep in mind, maintaining system logs is useless if someone or some tool does not review them.

#vi /etc/syslog.conf

```
*.info;mail.none        /var/adm/messages
auth.info               /var/adm/authlog
*.crit;auth.none        *
```

The basic syntax is as follows. The first column states the *facility.priority* and the second column state where the messages should be sent. In my example, any informational priority level messages, excluding any mail facility, will go into the */var/adm/messages* file. Any authorization facility, which include login, su, and getty, at informational priority level or higher will be sent to the */var/adm/authlog* file. Any critical facility excluding, any authorization facility, will go to all users. Here is a helpful URL to get you started.

http://www.unidata.ucar.edu/cgi-bin/man-cgi?syslog.conf+4

You will have to do some tweaking to get the information you are looking for. If you really get ambitious, you can redirect the messages to a centralized syslog server so you can manage the messages from one system.

As the root user, you will need to create the various files to send the messages to because syslog does not create them. As you can see, I'm protecting the *authlog* file because I contains various *su* and login information that need to be protected.

#touch /var/adm/messages
#touch /var/adm/authlog
#chmod 600 /var/adm/authlog

Creating a script to rotate logs so they do not fill up */var*. This is adapted from the SANS Security Solaris Step by Step Guide.
#cd /usr/local/bin
#vi rotatelogs.sh

```
#! /bin/sh
#
LOG=messages
cd /var/adm
test -f $LOG.2 && mv $LOG.2 $LOG.3
test -f $LOG.1 && mv $LOG.1 $LOG.2
test -f $LOG.0 && mv $LOG.0 $LOG.1
mv $LOG   $LOG.0
cp /dev/null $LOG
chmod 644    $LOG
#
LOGDIR=/var/adm
LOG=authlog
if test -d $LOGDIR
then
        cd $LOGDIR
        if test -s $LOG
        then
                test -f $LOG.2 && mv $LOG.2  $LOG.3
                test -f $LOG.1 && mv $LOG.1  $LOG.2
                test -f $LOG.0 && mv $LOG.0  $LOG.1
                mv $LOG    $LOG.0
                cp /dev/null $LOG
                chmod 644    $LOG
                sleep 40
        fi
fi
#
kill -HUP `cat /etc/syslog.pid`
```

Make script only readable by root.
#chmod 700 rotatelogs.sh

NOTE: Be very careful when running shell scripts as root. Because root has privileges to remove or modify almost any file, you need to verify that scripts are doing only what it should be doing.

Add into crontab.
#crontab -e
10 3 * * 0 /usr/local/bin/rotatelogs.sh

Changing The Login Screen Welcome Message
You can protect yourself legally by not "welcoming" in unauthorized access. By default, the NOTE: The IBM Redbook hardening doc has incorrect flags. The system will report this usage error `Usage: chsec -f file -s stanza -a "attr=value" ...` when trying to run `chsec -f /etc/security /login.cfg -a default -herald "Unautorized use of this system is prohibited.\n\nlogin: "` from the book. This will work:

#cp –p /etc/security.user /etc/security/user.orig
**#chsec -f /etc/security/login.cfg -s default -a herald **
"Unauthorized use of this system is prohibited.\n\nlogin: "

The "\" is an indication of a continuation to the next line and not a command flag. You can also edit the */etc/security/login.cfg* file if you are unfamiliar with the *chsec* command.

```
default:
herald="Unauthorized use of this system is prohibited\n\nlogin:"
```

Enforce Security policies
Enforce stronger user password policies by running against a dictionary list provided by AIX. You can do this by modifying the default users environment file.

#vi /etc/security/user
default:
minother = 3 (minimum number of non-alphabetic characters)
minlen = 8 (minimum length of password)
dictionlist = /usr/share/dict/words

NOTE: Additional documents on SSH password expiration can be found here:
http://marc.theaimsgroup.com/?l=openssh-unix-dev&m=104409280114747&w=4
http://www.zipworld.com.au/~dtucker/openssh/

Disable Direct Root Access
Do not ever allow direct root login except for on the console. Only allowing users to "su" to root. This will create an audit trail for root activity on the system.

#vi /etc/security/user
root:
rlogin = false

Securing Outside File Access
This will allow files to be own by the user and will deny group or outside access to the files. By default, the *umask* is set to 022.
#vi /etc/security/user
default:
umask = 077

Enabling SAK
I have included this because of an excerpt from the Waterloo document about the Secure Attention Key (SAK). SAK is located in */etc/security/login.cfg* file. The *sak_enabled* variable defines whether the security attention key is enabled and is the **Ctrl-X**, **Ctrl-R** key sequence. A value for true processes the key sequence establishing a trusted path and enables the user to "invoke a trusted shell" at login with the **Ctrl-X**, **Ctrl-R** key sequence.

From a security standpoint, there is a potential security risk when the *tty* login devices are world writeable (0622). This is the default tty mode when SAK is set to false. By setting sak_enabled = true, the *tty* mode is set to 0600 at login which will address this risk. By default this value is set to false.

Here is a helpful URL.

http://www.ncsa.uiuc.edu/UserInfo/Resources/Hardware/IBMp690/IBM/usr/share/man/info/en_US/a_doc_lib/cmds/aixcmds5/tsh.htm

You can also refer to the AIX 5L System Management Guide: Operating System and Devices for additional documentation. Here is the URL.

http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/aixbman/baseadmn/about.htm

Auto Log Off
You want to enable this feature to preventing someone from sitting down on a workstation and jumping onto an existing terminal session as root. An additional level of security would be make sure to that the screen lock is enabled.

#cp -p /etc/security/.profile /etc/security/.profile.orig
#vi /etc/security/.profile
TMOUT=600; TIMEOUT=600; export readonly TMOUT TIMEOUT

Securing Path
Make sure there is no trailing period in your path. If so, remove it. By doing so, you prevent unauthorized execution of files from a current working directory.

#vi /etc/security/.profile
-remove "." from Path

Removing Generic Users/Groups
It is important to remove generic users and groups for several reasons:

1. this system is not going to use services owned by these users
2. removing unused uids/gids reduces the number of potential logins
3. generic accounts are hard to identify the actual user since they do not login using their associated uid first. If you need to use generic accounts, look into using the *sudo* command.

NOTE: All original configuration files have been copied and appended with ".orig" file extension for reference purposes.

#cp -p <passwd.conf> <passwd.orig>

Removing users: uucp, nuucp, lpd, lp, imnadm, guest, invscout, snapp
#rmuser -p <uucp>

Users	user process	why we don't need it
uucp	UUCP network	outdated
nuccp	UUCP network	outdated
lpd	printing subsystem	no needed print services
lp	printing subsystem	no needed print services
imnadm	IMN search engine	we disabled it in the inittab
guest	guest user account	no access granted for generic accounts
invscout	inventory scout	IBM Utility for gathering microcode; do not currently use it

snapp	PDA administration	System Networking, Analysis, and Performance Pilot; enables unix administration tasks on PDA device; do not currently use it

Remove groups: imnadm, uucp, printq, lp (argue staff and nobody entries)
#rmgroup <imnadm>

group	group process	why we don't need it
imnadm	IMN search engine	we disabled it in the inittab
uucp	UUCP network	outdated
printq	printing	no needed print services
lp	printing	no needed print services

Removing the Associated Files for the Generic User/Group
#find / -nouser -ls

Make sure you have cut out full paths and then save into a temporary file.
#find / -nouser -ls | cut -c66-
#find / -nouser -ls | cut -c66- > /tmp/nouser

Tar the files and preserve the symbolic links with the –p flag.
#tar -cvfp /tmp/nouser.tar `cat /tmp/nouser`

Remove all un-owned files and directories
#rm `cat /tmp/nouser`

Double check to make sure you have removed everything.
#find / -nouser -ls

Do same for "-nogroup"
#find / -nogroup -ls

Make sure you have cut out full paths and then save into a temporary file.
#find / -nogroup -ls | cut -c66-
#find / -nogroup -ls | cut -c66- > /tmp/nogroup

Tar the files and preserve the symbolic links
#tar -cvfp /tmp/nogroup.tar `cat /tmp/nogroup`

Remove all un-owned files and directories.

#rm -r `cat /tmp/nogroup`

Double check no make sure you have removed everything.
#find / -nogroup -ls

Remote Host Access

There is no *.rhost* files with a default install. If you have a system you need to check for them, you can run this command:
#find / -name .rhost -ls

Securing Crons
From the */var/adm/cron* directory, remove the *cron.deny* and *at.deny* files.
#rm cron.deny at.deny

Create *cron.allow* and *at.allow* and only place root in this file. By doing this, you implicitly deny all users from running a *cron* or *at* job. Secure files with writeable permissions and ownership as root.
#vi cron.allow
root

#vi at.allow
root

#chown root.cron at.allow
#chown root.cron cron.allow
#chmod 640 *.allow

example:
-rw-r--r-- 1 root cron 5 Jan 9 11:28 at.allow
-rw-r--r-- 1 root cron 5 Jan 9 11:28 cron.allow

Baseline Report on Setuid/Setgid Files
Save these files in a secure location. You will be able to do a poor man's integrity check to see if any new setuid/setgid have been created. This will allow you to monitor the system periodically and see if any activity has happened without your knowledge. Setuid/setgid can be used to run scripts as any user based on the file ownership, including root, which would be a way of compromising the system. You should move these files to a secure location, preferably onto an external media.

#find / -perm -4000 -user 0 -ls > /tmp/setuid
#find / -perm -2000 -user 0 -ls > /tmp/setgid

/etc/rc.dt
This script starts the Common Desktop Environment (CDE) GUI but it is not installed with this secure base install. If you have an absolute need for the CDE, you will want to install the CDE packages.

Prevent Unauthorized Monitoring of Remote X server
X11.apps.clients is not installed when using a secure base build.

Final Reboot
Make sure and reboot the system and you can get back to a login prompt cleanly. Verify that *syslogd* is logging to the correct log files. At this point, you are ready to install the third party application.

Ongoing Maintenance

Overview
After the steps for hardening are complete, you will want to make a valid *mksysb* of the operating system. This will serve as a system back up in case anything should corrupt the operating system in the future. You should also have the root volume group mirrored and have backup software to do complete system backups to move data from disk to tape to have off-site tape storage for your backup media and *mksysbs*. If you had the money, replicating to another geographical site would be a great additions and don't forget to give yourself a raise. You will want to keep up on patches and firmware updates to fix any system bugs and vulnerabilities that become known. Lastly, you should make sure important system logs are being redirected to you and also do simple systems checks of critical files to maintain a level of system integrity.

Mksysb
A *mksysb* is a bootable image on various media types that can be used to restore the operating systems and any additional data located in the root volume group. You can find what is currently in the root volume group by running this command.

```
#lsvg -l rootvg
rootvg:
LV NAME  TYPE     LP    PPs   PVs   LV STATE          MOUNT POINT
hd5      boot     1     2     2     closed/syncd      N/A
hd6      paging   30    60    2     open/syncd        N/A
hd8      jfslog   1     2     2     open/syncd        N/A
hd4      jfs      2     4     2     open/syncd        /
hd2      jfs      16    32    2     open/syncd        /usr
hd9var   jfs      11    22    2     open/syncd        /var
hd3      jfs      4     8     2     open/syncd        /tmp
hd1      jfs      1     2     2     open/syncd        /home
```

```
hd10opt  jfs      1     2     2     open/syncd        /opt
```

Mksysbs may also be useful in forensic analysis in case the system gets hacked. You would have an image of what the system configurations looked like before the system was compromised and be able to identify any possible system changes. This can be a valuable source of information to find out how a unauthorized or authorized entity obtained access to the system and what was compromised. Forensic analysis is not in the scope of this paper but I hear that SANS offers a course in this sort of thing.

Here are instructions to create an 8mm tape backup on a 20GB tape drive. The newer IBM p-series servers offer DVD drives that can create the boot images onto DVD media. There is also the option of installing boot images over the network using IBM's Network Install Manager (NIM) and SysBack. I do not have any experience using these methods to create a *mksysb* but you may find it useful in your data center. Our data center have only begin to assess the possible benefits of these other methods but the 8mm tape medias were already being used when I came is organization and like most, I inherited this portion of the project. Here is what to run:

#smitty mksysb

```
                Back Up This System to Tape/File

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

[TOP]                                                [Entry Fields]

Backup DEVICE or FILE                                 [/dev/rmt0]
```

This is the only field you will need to modify. You can type **F-4** or **Esc-4** on the keyboard to get a list of available backup devices.

Press **Return**, after you have modified this field.

You will see the system begin to create the *mksysb*.

```
                          COMMAND STATUS

Command: running       stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

Creating information file (/image.data) for rootvg....
```

```
Creating tape boot image......

Creating list of files to back up.....

Backing up 71642 files...............

5636 of 71642 files (7%).............................

30873 of 71642 files (43%)............................
```

After completion of the *mksysb*, you will want to test the tape to make sure the tape is valid. The best way to validate the tape is to restore from it. Here are the steps:

#smitty restmksysb

Modify the restore device to be the same device that you used to create the *mksysb*. Again, you can press **F-4** or **ESC-4** to pull from a list of available devices.

```
                  Restore Files in a System Image

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

                                                      [Entry Fields]

* Restore DEVICE or FILE                              [/dev/rmt0]
```

Press **Return**, after you have modified this field.

After completion, you should have the same image of the operating system you worked so hard to build.

Mirroring the Root Volume Group

The benefits of a mirrored root volume groups is higher availability of the boot image, operating system, and other critical files you deem worthy of placing there. Disk mirroring will allow for you to lose a physical disk and still be able to keep the operating system functioning. When in disk failure, you will need to contact IBM technical support on verifying the disk failure and replace the disk and re-mirroring the root volume group. If you have the resources, the root volume group containing the operating system should be on a separate disk separated from all of your other data. This will separate non-related system data

from your operating system and help prevent possible system halts from filling up critical file systems and/or data corruption.

Here are the steps to follow.

Gracefully shutdown the server and place the new drive on the appropriate sled into the disk location. Note: As of July 2003, most IBM servers have the ability to hot swap the drives while the system is running. IBM will never recommend this method or be responsible for data loss but I can be a witness to the ability to hot swap drives. If you can bring the system down, this is the safest method.

#shutdown –Fr

Once the disk has been placed into the disk slot, power the system back on. During boot up sequence, the system will run the Configuration Manager (cfgmgr) which will auto detect any new hardware. If you choose to add the disk while the system is up, you will need to run *cfgmgr* from command line. You can confirm that the new drive has been located by running this command:

```
sloth:/# lspv
hdisk0          000cc26f7d5e5f64                    rootvg
hdisk1          none                                none
```

Add the *hdisk* to the root volume group.

#extendvg <rootvg> hdisk<1>

Quorum should only be used for 3 or more disks. When using 1-2 disks in a volume group, one disk will contain 2 Volume Group Descriptor Areas. (VGDA). The function of quorum protection is to maintain a level of data integrity based on 51 percent of the VGDAs in a volume group being present. This does not work when mirroring 2 disk since one of the disks will contain 2 VGDAs. If you would loss this disk, you would lose 66 percent of the VGDAs and therefore not maintain quorum. This would result in the volume group varying off and make the disks unavailable. Disk quorum does not have a practical use with 2 disk mirroring therefore turn it off.

#chvg -Qn <rootvg>

Mirror the disks and sync any stale partition.

#mirrorvg -S <rootvg>

Update the boot record for the disks in the root volume group

#bosboot -a

Update the boot list to include the additional disk to the root volume group. Your boot list may be different based on your available boot devices.

#bootlist -m normal rmt0 cd0 hdisk0 hdisk1

Verify your boot list.

```
# bootlist -m normal -o
rmt0
cd0
hdisk0
hdisk1
```

Backup Software

You will want to do complete system backup off all the volume groups and file systems using a backup software. I believe that Tivoli, Legato, and Veritas are some major player is this market. You will want to strongly consider placing the installation of these binaries in the root volume group. The reason is in case you have a complete failure of a server and you need restore the system as quickly as possible. Having the backup software in the root volume group will allow you to start backing up all additional data from tape backup after a *mksysb* is completed. Backup software is also handy with just restoring one file or a specific sub-directory. *Mksysbs* are all or nothing.

Because of the different installation and administration guidelines of the various backup software vendors, the scope of setting up 3rd party backup software is not included in this paper but you will need to chose the vendors can provide you with the appropriate software that best suits your need and can scale to your data center. Make sure to get the software sale team to take you out to lunch and you're a size large. In my case, you inherited the backup software that already existed.

Media Rotation and Off-site Storage

You will want to have an appropriate media rotation plan for your *mksysbs* and backup tapes. Check with your company restore policy or contractual agreements with customers on a disaster recovery plan to gauge the frequency of backups and *mksysbs* that you will perform. If you don't have on, you will have to determine what is best practice for your size company.

One thing to consider, if your *mksysbs* are set up properly, the data in root file system should most likely only change with operating system changes, but not limited to, maintenance level upgrades, new devices, and in our case backup software upgrades.

When considering your media rotation plan, you will also need to incorporate off-site storage. Again, you will need to gauge the frequency of off-site storage based on your retention policies and if you have the resources to create clones of

the media or actually need to recycle the original tapes off-site. Off-site storage mean somewhere outside of the building. If it means taking the tapes back to your home, send them to Alaska (unless you work there), or using a 3rd party off-site vendor, you should always plan for the worse case scenario like if the building burned down.

To conclude this section, a high level basic procedure to restore a system, in case of a failure, would be to recover the operating system from the current *mysysb,* ensure that the backup daemon processes are running, and then start to recover the rest of the volumes groups from your tape backups using your backup server. In case of a data center melt down, you would have all off your off-site media to recover from, right? Or even better, a replicated data center on the other side of the country! Ok, back to my reality.

Patches and Firmware

Keeping up-to-date on maintenance and firmware levels is one of the basic practices that can be done to make sure the system is protected against system bugs and vulnerabilities as they become known. Here is the URL for obtaining the most current maintenance level.

https://techsupport.services.ibm.com/server/aix.fdc?toggle=DNLDML

You can be put on an email list from IBM to be notified of any new patch, fixes, and firmware releases to your system. Here is the URL:

https://techsupport.services.ibm.com/server/pseries.subscriptionSvcs

Follow the instruction on page 8 titled “Applying Latest Patches to AIX 5L” to update the system.

You will also need to keep up to date on firmware (also known as microcode) updates. This is done less often but is still important. Here is how you can see what your system firmware levels are.

```
# lscfg -vp | grep -p Firmware

Service Processor Firmware:

ROM Level (alterable).......sc020308

Version.....................RS6K

System Info Specific.(YL)...P2/Y2

System Firmware:

ROM Level (alterable).......NAN02066
```

```
Version.....................RS6K

System Info Specific.(YL)...P2/Y1
```

You can also do a *grep* on "alterable" to find other firmware devices. Thankfully, IBM has a PDF on everything you wanted to know about firmware. Here is the URL.

http://www.ibmlink.ibm.com/usalets&parms=H_203-145

You can call have your IBM field engineer come to your site to do the updates or do it yourself. When you are ready, here is the URL for downloading firmware.

http://techsupport.services.ibm.com/server/mdownload/download.html

System Alerts
(If you have your system logs set up correctly, the system should alert you in some way i.e. emails, reports to console, paging, centralized system log server, or 3rd party monitoring software like BigBrother. I'm not endorsing BigBrother but we do use it. This paper does not cover any aspect of the BigBrother software.

Checking System Files
You can use simple *diff* command on critical system files and compare them to your trusted system files to see if any differences have occurred. Run them in periodic on a *cron* job to do periodic checks on the system. By moving the your trusted baseline reports (i.e. the baseline setuid/setgid procedures) and system files into a secure directory or external media, you can use these as a form of a health check for the system. If you know enough about scripting, you should have it email or even text page you if any of these critical files change. If you are unfamiliar with scripting, find someone in your organization that does. These would include and not limited to:

/etc/inetd.conf	/etc/.rhosts -trusted hosts
/etc/rc.tcpip	/etc/.netrc -ftp remote login
/etc/inittab	/var/adm/cron/at.allow
/etc/rc.nfs	/var/adm/cron/cron.allow
/etc/passwd	/tmp/setuid
/etc/security/passwd	/tmp/setgid
/etc/host.equiv -ssh, r-services	

NOTE: AIX version 5.1 does not create the *.rhosts* file by default. It is very important that you monitor whether or not this file is used because hostname or ip address entries placed into this file allows for un-authenticated/unrestricted access to the server from that host using a rlogin session. This is an easy way for intruders to access your system as root. If they do, game over. A creative way other administrators have dealt with this is by creating a *.rhosts* file and

linking it to */dev/null* which sends it to "la-la land". You will have to decide what method is appropriate.

You will need to make sure to do your homework to really understand the importance of these basic configuration files and processes. This paper does not cover the details of how these files or processes can be exploited. One place to start is the SANS Top 20 List.

URL: http://www.sans.org/top20/

There are many proprietary tools that can help you to monitor and address any changes to your system. If you can find the extra time to install and configure these tools, great. For myself, there are so my hours in a day for setting up these tools and plus my golf game needs more help anyway. Tripwire for system integrity checks, and BigBrother for process monitoring, and a centralized system log server are a good start. The URLs are located on the reference page.

It is important to run periodic audits on your system. If you have regularly scheduled downtimes, this is a good time to do these tasks. You should make sure to coordinate with the various teams that need to be notified of the event and inform you management of the tasks you plan on doing to audit the system. Some simple commands and tools are discussed in the next section.

Check Your Configurations

Now it is time to verify our hardened system. Here are some helpful commands that you can run to test which ports are available. I will using *rpcinfo, netstat, lssrc,* and *lsof.* All these utilities, minus *lsof*, come installed with the base operating system.

Rpcinfo Command

If you have any unidentified ports numbers, you can run *rpcinfo* to see if it has been registered via Remote Procedure Call (RPC). As you can see from the output, I had to manually start the portmap subsystem before I was able to call the *rpcinfo* command. This should indicate that the system has taken step to become hardened. This particular system does not require portmapper to be running therefore it was removed out of the */etc/rc.tcpip* file.

```
# rpcinfo -p
rpcinfo: can't contact portmapper: RPC: Rpcbind failure - RPC: Failed
(unspecified error)

# startsrc -s portmap
0513-059 The portmap Subsystem has been started. Subsystem PID is 9366.

# rpcinfo -p
   program vers proto   port  service
```

```
    100000    4   tcp    111  portmapper
    100000    3   tcp    111  portmapper
    100000    2   tcp    111  portmapper
    100000    4   udp    111  portmapper
    100000    3   udp    111  portmapper
    100000    2   udp    111  portmapper

# stopsrc -s portmap
0513-044 The portmap Subsystem was requested to stop.
```

The only service that *rpcinfo* reports is portmapper, which was started manually. Therefore there are no additional services running on the system.

Netstat Command

The *netstat* command can show you active listening and established tcp/udp ports. The available ports should correspond with the previous steps taken to harden the system by removing most services from the system configurations files. As you can see, my SSH session is the only reported connection coming from a foreign address port 12563 which is my PC.

```
 # netstat -an
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address        (state)
tcp4  0      0         *.22             *.*                    LISTEN
tcp4  0      0       x.x.x.x.22         x.x.x.x.12563      ESTABLISHED
udp4  0      0         *.514            *.*
```

Lssrc command

The *lssrc* command is unique to AIX operating systems that will make a request to the System Resource Controller (SRC) to list the status of all the subsystems and groups on the system. You can use this to verify that only *syslogd* (port 514) and *sshd* (port 22) are the only active subsystems and also verify all the inoperative subsystems.

```
# lssrc -a
Subsystem        Group            PID     Status
 syslogd         ras              4006    active
 sshd            ssh              6710    active
 qdaemon         spooler                  inoperative
 writesrv        spooler                  inoperative
…more…
```

Lsof Command

Lsof is a powerful tool that you will be able to list open files, associated processes, owners, and port information. *Lsof* has many command flags and can be used in more ways that is shown in the examples below. Again, extra homework is needed to really utilize this tool. You can go to the IBM website listed below to install the version 5.1 RPM.

Download website:

lsof
http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html

You can either install it via rpm command or by using *smitty*. IBM Technical Support has told me that RPM installs are support by *smitty*. I have not verified this myself so install with caution.

#rpm –i lsof-4.61-3.aix5.1.ppc.rpm

Verify the install.

```
# rpm -aq
mtools-3.9.8-1
cdrecord-1.9-4
mkisofs-1.13-4
lsof-4.61-3
AIX-rpm-5.1.035-2
openssl-0.9.6e-1
```

The first three RPMs are packages utilized for the DVD burner that was included with the hardware setup. We installed the *"openssl"* package earlier and the *"AIX-rpm"* package is needed to install RPMs.

To install via *smitty,* run this command:

#smitty install_all

The input device will be the directory where you copied the file. You can get the "SOFTWARE" name by pressing **F-4** or **Esc-4**. Press **F-7** to select *lsof-4.61.3.* Modify these fields:

```
           Install and Update from ALL Available Software

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

                                                      [Entry Fields]

* INPUT device / directory for software               /tmp

* SOFTWARE to install                                [lsof-4.61 ]

  PREVIEW only? (install operation will NOT occur)     no
```

You will always want to preview the install. Before installing any package. You can toggle the **Tab** key to change the field.

```
                        COMMAND STATUS

Command: running       stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

geninstall -I "a -cgNpqwX -J" -Zp  -d . -f File 2>&1

File:

    R:lsof-4.61
```

Here are some basic *lsof* options you can start with. The *+M* flag indicates to display network based process and their associated RPC names. I am running it for both UDP and TCP protocols. The last command string with the *+L1* flag searches for any unlinked files that can be used by hackers to hide data. It reported nothing of this sort.

```
# lsof +M -i udp

COMMAND  PID USER   FD   TYPE     DEVICE SIZE/OFF NODE NAME

syslogd 6478 root    4u  IPv4 0x7013f600      0t0  UDP *:syslog

#lsof +M -i tcp

COMMAND   PID USER   FD   TYPE     DEVICE SIZE/OFF NODE NAME

sshd    17668 root    3u  IPv4 0x701461e4      0t0  TCP *:22 (LISTEN)

enkidu:/# lsof +L1
```

Nmap

Nmap is a port scan utility that is very helpful to determine the port availability from outside the system. I like Nmap because it runs like most other unix commands and is easy to install via RPM. There is a glorified version of Nmap with additional bells and whistles and is actually considered a network scanner called Nessus. If you have the time, I encourage you to play with it, but Nmap will enable you to easily scan all TCP/UDP ports on the host system to reinforce the *netstat* and *ps* outputs and validate what hardening has done.

There are various operating system platforms you can install Nmap on. Depending on which one, follow the instructions to install the utility from the Nmap home page.

URL: http://www.insecure.org/nmap/nmap_download.html

Lets run Nmap and validate the hardened system. You'll need to familiarize yourself with Nmap but I'll give you some basic flags that you can use to verify your system.

Nmap has an *–O* (that the letter O) option which tries to determine the operating system by sending a series of crafted packets and running it against a database of "signatures". Since we are running this against one known target server that is up, we can use the *–P0* flag to suppress pinging the host.

```
# nmap -P0 -O server_name

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Interesting ports on nietzsche (165.79.148.238):
(The 1600 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
No exact OS matches for host (If you know what OS is running on it, see
http://www.insecure.org/cgi-bin/nmap-submit.cgi).
TCP/IP fingerprint:
SInfo(V=3.00%P=i686-pc-linux-gnu%D=7/26%Time=3F233FDA%O=22%C=1)
TSeq(Class=TR%IPID=I%TS=U)
T1(Resp=Y%DF=N%W=402E%ACK=S++%Flags=AS%Ops=M)
T2(Resp=N)
T3(Resp=Y%DF=N%W=402E%ACK=S++%Flags=AS%Ops=M)
T4(Resp=Y%DF=N%W=4000%ACK=O%Flags=R%Ops=)
T5(Resp=Y%DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T7(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
PU(Resp=Y%DF=N%TOS=0%IPLEN=38%RIPTL=15C%RID=E%RIPCK=F%UCK=0%ULEN=134%DA
T=E)

Nmap run completed -- 1 IP address (1 host up) scanned in 17 seconds
```

As you can see, this hardened system has proven that Nmap cannot identify the operating system. One mark for the good guys!

We will now run a TCP, UDP, and RPC scans to see what Nmap reports. The flags are *–sF* for TCP, *-sU* for UDP, and *-sR* for RPC.

```
# nmap -sF -sU -sR -P0 server_name

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Interesting ports on nietzsche (165.79.148.238):
(The 3067 ports scanned but not shown below are in state: closed)
Port       State       Service (RPC)
22/tcp     open        ssh
514/udp    open        syslog

Nmap run completed -- 1 IP address (1 host up) scanned in 4 seconds
```

Nmap reported only port 22 and 514 that validates the only ports open in the previous *netstat* output. Another mark for the good guys!

This document is not an end to building a secure system but will give insight on how to begin to approach stopping some of the basic system threats to an AIX system. Talking amongst your peers, keeping in touch with current system events, and continued training in system security all provide the most valuable assets to learning all there is to know about AIX.

References

AIX toolbox for Linux applications: Open Source packages available for AIX 4.3.3 & AIX 5L.
URL: http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html

AIX 5L Version 5.1 Installation Guide, Version 1. IBM Corporation April 2002.
URL: http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base/aix51.htm

System Management Guide: Operating System and Devices.
URL:
http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/aixbman/baseadmn/about.htm

Batten, D., Joglar, A., St. Clair, L., Schreitmueller, S., Sanchez, R. "Strengthening AIX Security: A System-Hardening Approach." 26 March 2002.
URL: www.ibm.com/servers/aix/whitepapers/aix_security.html

Big brother Homepage:
URL: http://bb4.com/

Parmar, Harpal. Building a Cost Effective Syslog Server using Solaris For Intel and SunScreen Lite. 2003.
URL: http://www.giac.org/practical/GCUX/Harpal_Parmar_GCUX.pdf

Commands Reference, Volume 5 - tsh Command.
URL:
http://www.ncsa.uiuc.edu/UserInfo/Resources/Hardware/IBMp690/IBM/usr/share/man/info/en_US/a_doc_lib/cmds/aixcmds5/tsh.htm

Fix Delivery Center for AIX Version 5: Download maintenance packages.
URL: https://techsupport.services.ibm.com/server/aix.fdc?toggle=DNLDML

Genty, Denise. "IBM AIX network security developer." January 2003.
URL: http://www-1.ibm.com/servers/esdd/articles/openssh_updated.html?Open&ca=daw-home-news02

IBM developerWorks: Toolbox subscription.
URL: http://www-106.ibm.com/developerworks/toolbox/guest.html

IBM developerWorks: Open source projects. "OpenSSH on AIX Images Project: Files." 24 July 2002.
URL:
http://www-124.ibm.com/developerworks/downloads/index.php?group_id=108

IBM e-server pSeries & RS/6000 Microcode Updates.
URL:
http://techsupport.services.ibm.com/server/mdownload/download.html#update

IBM e-server pSeries Support: Alert for pSeries Customers.
URL: https://techsupport.services.ibm.com/server/pseries.subscriptionSvcs

IBM e-server pSeries Support: Fix Delivery Center for AIX Version 5.
URL: https://techsupport.services.ibm.com/server/aix.fdc?toggle=DNLDML

IBM e-server Certification Study Guide – pSeries AIX System Administration.
IBM Redbooks, December 2001.

"Installing OpenSSH for AIX 5.1" IBM Corporation, 31 October 2002.

Nmap Home Page.
URL: http://www.insecure.org/nmap/nmap_download.html

Quinton, Reg. "Security Review: AIX 4.3 Network Hardening." 15 January 2001
URL: http://ist.uwaterloo.ca/security/howto/2001-01-15/

Rae, K., Un, A. "Unix: OS Installation." Version 1.2 (unpublished). 22 April 2002.

SANS Top 20 List
URL: http://www.sans.org/top20/

Securing File Transfer Protocol (FTP) using Transport Layer Security (TLS).
URL: http://cs.engr.uky.edu/~prasad/MastersProject.html.

Solaris Security Step by Step Version 1.0. The SANS Institute. 1999.

Spencer, Bruce. "AIX Tip of the Week: Choosing Between a 32 vs 64 Bit Kernel in AIX 5." 12 May 2002.
URL: http://silcon.silcon.com/~baspence/AIXtip/aix5_kernel.htm

Tripwire Homepage: Academic Source Release.
URL: http://www.tripwire.com/products/tripwire_asr/

Un, A. "Vulnerabilities & Secure Base Build of AIX 5.1." 31 December 2002.

URL: http://www.giac.org/practical/GSEC/Al_Un_GSEC.pdf

UNIX man pages: syslog.conf (4).
URL: http://www.unidata.ucar.edu/cgi-bin/man-cgi?syslog.conf+4

Vetter, S., Chaudry, A., de Klerk, A., Kong, Y., Reid, E., Singh, N.P. IBM Certification Study Guide AIX V4.3 System Administration. IBM Corporation, May 1999.

Appendix A

Installing OpenSSH for AIX 5.1

Contents

About this document

OpenSSH is a set of client and server software that allows you to encrypt telnet, ftp, and remote copy traffic between two machines.

This document describes the procedure for installing OpenSSH at AIX 5.1

IMPORTANT: The procedure below assumes that there are no other (third-party) versions of OpenSSH already installed on the system. If a third-party version of OpenSSH is currently installed, you will need to remove it before proceeding with this installation.

Obtaining necessary software

1. Obtain rpm.rte from your AIX Base Install media
2. Download OpenSSL software from AIX Toolbox for Linux Applications - Cryptographic Content
3. Download OpenSSH software from DeveloperWorks Website

   Click on the OpenSSH package corresponding to your OS level
   - For AIX 5.1
     - Click on "3.4p1_5.1"
     - Read the Release Notes

        - Scroll to the bottom and download openssh34p1_51.tar.Z
    - For AIX 5.2
        - Click on "3.4p1_52"
        - Read the Release Notes
        - It is imperative that you set up /etc/pam.conf as documented in the Release Notes else OpenSSH will not work. This is only for AIX 5.2)

Installing OpenSSL

1. Update AIX-rpm database (This may take several minutes to complete)
2.
3. # /usr/sbin/updtvpkg
4. Install OpenSSL software
5.
6. # cd <directory containing rpm images>
7. # rpm -i openssl-0.9.6e-2.aix4.3.ppc.rpm

Installing OpenSSH

1. Install openssh filesets
2.
3. # cd <directory containing **uncompressed** openssh filesets>
4. # rm .toc
5. # smitty install_latest
6. (use '.' as your input directory and _all_latest for the "SOFTWARE to install")

Testing your OpenSSH installation

1. Connect to sshd from a client
2.
3. # ssh root@server_name
4. Enter "yes" when asked if you want to continue connecting
5. Enter root's password