GIAC Security Essentials Certification (GSEC)
Version 1.4b Option 2
Vulnerabilities & Secure Base Build of AIX 5.1
By Al Un

Abstract

This document covers identifying existing vulnerabilities of AIX 5.1 during a default install and the process of installing a secure base build for AIX 5.1 base operating system (BOS). Installation was done using “installp” to load the base operating system. At each step of the hardening process, we incrementally compared network statistics and running processes against the default build.

The target audience for this paper may not be someone known as the “guru” or “macgyver”, but intended for someone that has never built an AIX operating system or just someone needing to expand the way they should think about building systems.

Introduction

We are undoubtedly an IBM shop on the unix side. The unix servers run on an AIX platform. But, one of the biggest problems working with a group of unix systems administrators was everyone had their own way of building AIX servers. Based on the experience of the system administrator (SA) and the AIX software, variations of builds emerged from the somewhat knowledgeable prejudice approach to the "I have no idea what I'm doing, install everything" approach. There was never a sound base build document that could be used amongst all SAs. If we had such a document, every system could be spawned from the same operating system (OS) image.

This being a problem, I came up with an approach to building servers. This was built on the basic SANS principle that if you are not going to use it, turn it off. The easiest way for our data center to ensure that every server, being introduced into our network, was built on a common foundation was to create the box with all the ports, services, and post-base LPPs stripped from the OS. What would remain would be OpenSSH (port 22) to remotely access the server and syslogd (port 514) to enable system logging.

The goal is to have the base minimum number of services running on the system to:

- disable unused services
- protect against initial vulnerabilities and exploits
- educate SAs on application/software dependencies
- provide a hardened server by design
- understand and simplify the process

Succeeding in these goals would reflect a “netstat” output to look something like this:

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

This is done to protect the network against outdated or insecure protocols/services from the server but also the server against exploits or vulnerabilities that could be compromised from the network. By doing so, I give my team a system that is secure and hardened once the system is online and ready to become part of the network.

Vulnerabilities of AIX 5.1

Let look at these system vulnerabilities. This is a default AIX OS build with the “minimal” package sets to bring the system up and running. The screen capture below show these ports enabled by default.

```
# netstat -a | more
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address      Foreign Address        (state)
tcp4       0      0  *.daytime          *.*                     LISTEN
tcp        0      0  *.ftp              *.*                     LISTEN
tcp        0      0  *.telnet           *.*                     LISTEN
tcp4       0      0  *.smtp             *.*                     LISTEN
tcp4       0      0  *.time             *.*                     LISTEN
tcp4       0      0  *.sunrpc           *.*                     LISTEN
tcp4       0      0  *.smux             *.*                     LISTEN
tcp        0      0  *.exec             *.*                     LISTEN
tcp        0      0  *.login            *.*                     LISTEN
tcp        0      0  *.shell            *.*                     LISTEN
tcp4       0      0  *.rmc              *.*                     LISTEN
tcp4       0      0  *.writesrv         *.*                     LISTEN
tcp4       0      0  localhost.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  localhost.smux   localhost.3276         ESTABLISHED
tcp4       0      0  localhost.32768  localhost.smux         ESTABLISHED
udp4       0      0  *.daytime          *.*
udp4       0      0  *.time             *.*
udp4       0      0  *.sunrpc           *.*
udp4       0      0  *.snmp             *.*
udp4       0      0  *.syslog           *.*
udp4       0      0  *.ntalk            *.*
udp4       0      0  *.32776            *.*
udp4       0      0  *.32794            *.*
udp4       0      0  *.32845            *.*
```

The goal in is to have only have the SSH and syslogd services enabled. From this output, you can see there are additional services enabled by default. You can reference a number of commands (netstat, rpcinfo, lsof) and resources to find the names, services, and possible exploits for each of the ports. Several of the references cited at the end of this paper states are useful in this respect. And remember, www.google.com is your friend.

For some of the higher number ports (32xxx, 49xxx), IBM uses them to run several proprietary services that you do not need in a base build. The services can be found in the /etc/inittab file. You can call 1-800-CALL-AIX for software support for questions that may not be found the any of the references.

A Quick Note About Power

I wanted to talk briefly about something that has burned us several times. Hopefully your hardware will already be racked up with redundant power. Before we started auditing our procedures for hardware installation, we found some common problems in our facility. Here are some things to ask and check out yourself:

- does each server comes ordered with redundant power supplies?
- does each server rack have dual power strips?
- does each power supply connect to different power strips?
- does each power strip connect into different power feeds which are connected into different circuits?

Once your server is racked correctly with power and you have a terminal connected to it, you can start installing the AIX 5.1 Base Operating System. We use an IBM 3151 as a console during installation. Being dependant on the CDE or any of the graphics display requires more additional unnecessary software and can be insecure. This is not discussed in this paper but should be addressed.

Caveat

I am not advocating that every server be built from scratch. The idea in being a good systems administrator is to do work more efficiently so we have time to eat those jelly doughnuts. There are easier ways to build operating systems such as cloning from “mksysb” images or using Network Install Manager (NIM). These methods of installation are not covered in this paper.

Hardware Specifications

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.

Loading AIX 5.1 from CD

Back to the installation. 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. You will also need the latest patches. 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.

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's 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 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.

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          Elapsed time
% task complete       (in minutes)
```

This can take approximately 25 minutes based on the hardware specifications.

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 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 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.

Setting the Terminal Type

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

```
AIX Version 5
(C)  Copyrights by IBM and by others  1982, 2000.
#
```

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**.

Applying Latest Patches to AIX 5L

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 all the 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

Time needed to reboot the system and get back to the login prompt: 14 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.
```

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, exploits, and "how to's". You can use any search engine on the intranet 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 the 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.

The differences between AIX version 5.1 and 5.2 is the *prngd* does not use the */dev/random* file to generate the numbers in 5.1. Version 5.2 has included the */dev/random* file.

This is an excerpt from the email from IBM that was sent me.

> The only documentation I found on the details of random number generators used for OpenSSH was in the "What's New in AIX 5.2" class presentation materials. Regarding OpenSSH it states:
>
> Software Dependencies:
> PRNG - pseudorandom number generator- 4.3 only
> 5L using PRNG that ships with OpenSSH
>
> Modifications made by AIX:
> /dev/urandom for entropy & PAM support on 5.2
>
> Attached is the 5L installation "how-to". Please let me know if this information addresses your questions or if there is anything else I can help with.
>
> Thank you,
> Monica Sanchez
> AIX Support Line
> Netcom Group
> 1 800 CALL AIX

What does this all mean? IBM stated the version 5.1 SSH remote login process maybe slower than version 5.2. The seconds that it may take to randomly generate numbers is worth the added wait. Just think in a year, you can upgrade all of your servers to 5.2 and increase another form of security (job security).

Installing SSH

As of 12/13/2002, I have not found, nor could IBM technical support direct me to, an authorized published documents 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:

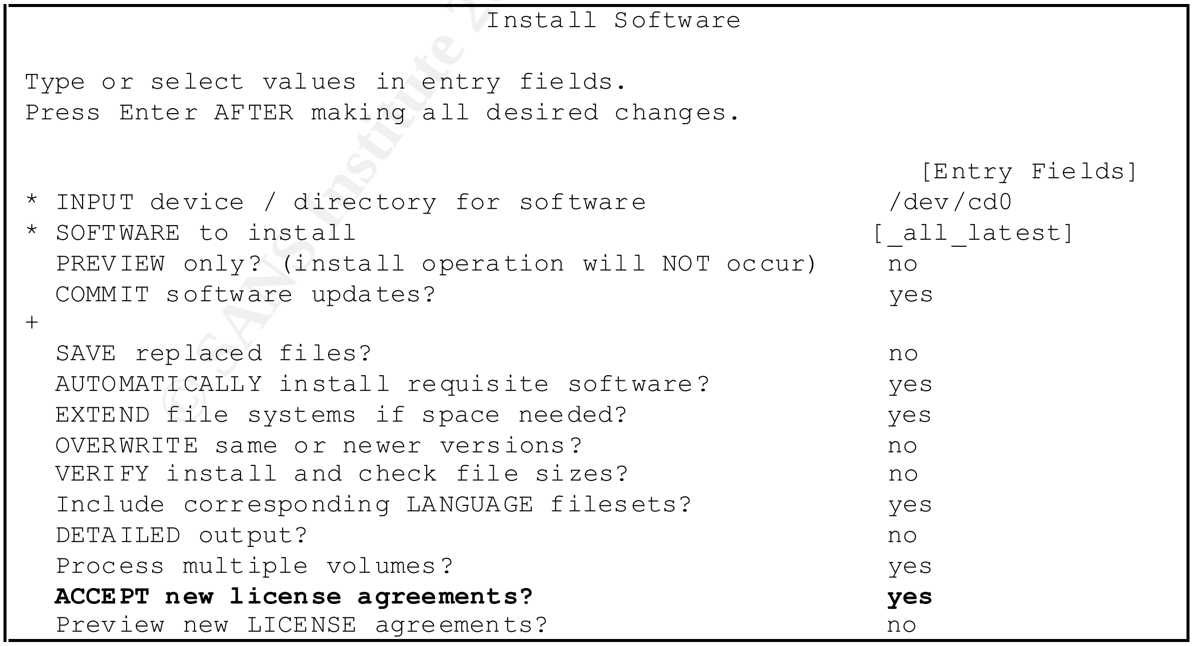
Install the OpenSSL RPM package first. The default install of the AIX V5.1 include the RedHat Package Manager (RPM) LPPs. You will need this to open RPM commands.

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

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
```

The Before

Here is 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  127.0.0.1.49      *.*                     LISTEN
tcp4       0      0  *.327             *.*                     LISTEN
tcp4       0      0  *.32              *.*                     LISTEN
tcp4       0      0  *.32              *.*                     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
tcp4       0   1596  x.x.x.x.23        x.x.x.x.1128          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 process 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  4991  4672   0   Dec 26  pts/0  0:00 -ksh
    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 12562  5934   0   Dec 26      -  0:00 telnetd -a
    root 12996  3156   1 22:10:15  pts/0  0:00 ps -ef
    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
    root 4504   3990 0   Dec 20        -  0:00 /usr/sbin/sshd -D
    root 4672   4504 0   Dec 20        -  0:01 /usr/sbin/sshd -D
```

System Hardening - /etc/inetd.conf

Lets start hardening the system. I'll start with */etc/inetd.conf* file. The goal it to shutdown as many port needed to run a skeleton base operating system. For the non-privileged ports >1024, I identified the port function and then decided if it could be safely shutdown without affecting the critical components needed to run the system.

After looking at the contents of the *inetd.conf* file, IBM had done some of the hardening work. They had commented out a number of the services. I take this one step further. I move and secure the file with its original contents with an ".orig" extension. At any point *inetd* services are called for, a simple copy can be done back in the original name.

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.

#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. My feeling is there should not be anything running out of *inetd* during the initial base system so I disable *inetd* completely. I will have a copy of the original file if I ever need to enable any services from *inetd*.

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
tcp4       0      0  x.x.x.x.22         x.x.x.x.1073         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                 *.*
```

```
# 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  9978  9614   0 23:06:03  pts/0  0:00 -ksh
    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  9614  8568   0 23:06:03      -  0:00 /usr/sbin/sshd -D
    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.co
    root 11870  3804   0 23:03:10      -  0:00
/usr/sbin/rsct/bin/ctcasd
    root 12146  3178   2 23:08:01  pts/0  0:00 ps -ef
    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
```

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 line 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.

SAs 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 other services in the file were removed because they were either not be used or the full understanding of the service was not clear 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, call IBM for software support.

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
```

```
tcp4       0      48 x.x.x.x.22            x.x.x.x.12539   ESTABLISHED
udp4       0      0  *.111                 *.*
udp4       0      0  *.161                 *.*
udp4       0      0  *.514                 *.*
udp4       0      0  *.32776               *.*
```

```
# 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 8518 8062   0 14:15:47      -  0:00 /usr/sbin/sshd -D

    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 7526 8518   0 23:29:19  pts/0  0:00 -ksh
    root 8062 4460   2 23:29:19      -  0:00 /usr/sbin/sshd -D
```

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"
```

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**.**

#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                *.*
```

```
# 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 6194 4038   0 00:22:59      -  0:00 /usr/sbin/sshd -D
    root 6539 6104   0 00:23:43      -  0:00 /usr/sbin/sshd -D
    root 6462 6708   2 00:24:19  pts/0  0:00 ps -ef
    root 6908 6539   0 00:23:43  pts/0  0:00 -ksh
```

System Hardening /etc/rc.nfs

No *nfs* services should need to be running at this time. 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.
Time needed to reboot the system and get back to the login prompt: 7 minutes

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 4010 4786   0 00:34:46      -  0:00 /usr/sbin/inetd
    root 4190    1   0 00:34:37      -  0:00 /usr/sbin/syncd 60
    root 4394    1   0 00:33:05      -  0:00
/usr/lib/methods/ssa_daemon -l ssa0
    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
    root 7416 7024   1 00:38:16  pts/0  0:00 ps -ef
```

The final output shows the goal of only 2 ports enabled for SSH and *syslogd*. 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.

Summary

A system administrator should be able to build this OS and place it on the network with the other members of the team knowing a basic hardening has been applied to system. All services and ports with regards to SSH and *syslogd* are disabled leaving the SA to decide what post-build services are needed to allow the system to run properly. The end goal was to create system awareness, provide a hardened server while documenting the process, protect the system against some basic known threats, and make it simple and easy to understand what was done to the operating system.

As system administrators, we can take a more active role in the decisions process for system requirements. When we do this, we now become protectors of the servers instead of baby sitters. This can lead us into a place where people like us become eventually known as "gurus" in our field.

This definitely is not an end to a hardening document but the beginnings of one. For my team, this document will provide a baseline for future builds in our data center. There are still many additional steps that will be taken to get this server to production. But, the first steps have been taken to ensure that all future systems are built with a sense of security in mind.

References

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

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

IBM develperWorks: 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

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.

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

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

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.

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

May need to add prngd based on
http://www-1.ibm.com/servers/esdd/tutorials/aix_ssh/3_5.html

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