



# Global Information Assurance Certification Paper

Copyright SANS Institute  
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

# **GIAC-GSNA Practical assignment**

## **Audit of Solaris 8 platform**

**Version 2.0**

---

**Azim Ferchichi,**

**January 2002**

© SANS Institute 2000 - 2002, Author retains full rights.

## Table of contents

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
<b>2</b>	<b>State of practice .....</b>	<b>5</b>
2.1	Research technique .....	5
2.2	What can be improved and how .....	6
<b>3</b>	<b>Security requirements .....</b>	<b>8</b>
<b>4</b>	<b>The security audit checklist .....</b>	<b>8</b>
4.1	Scan .....	8
4.1.1	Port scan .....	9
4.1.2	Vulnerability scan .....	10
4.2	Local audit .....	10
4.2.1	Installed packages .....	10
4.2.1.1	Selection of required packages .....	10
4.2.1.2	How to do the test .....	13
4.2.2	OS Installed patches .....	14
4.2.3	Network services, processes and daemons .....	14
4.2.3.1	Network services started by inetd .....	14
4.2.3.2	Boot services .....	15
4.2.3.3	Processes .....	16
4.2.4	Kernel tuning .....	19
4.2.4.1	Network parameters .....	19
4.2.4.1.1	ARP defenses .....	19
4.2.4.1.2	ICMP defenses .....	20
4.2.4.1.3	IP defenses .....	22
4.2.4.1.4	TCP defenses .....	23
4.2.4.1.5	Persistency of network parameters .....	26
4.2.4.2	User stack .....	26
4.2.4.3	File descriptor .....	27
4.2.4.4	Core file .....	28
4.2.5	File system .....	28
4.2.5.1	Partitions and mounted file systems .....	28
4.2.5.2	Basic permissions .....	29
4.2.5.3	World-write files .....	29
4.2.5.4	SUID and SGID files .....	30
4.2.6	Account and password policy .....	31
4.2.7	Console login .....	33
4.2.8	Umask and Cmask .....	33
4.2.9	Path variable .....	35
4.2.10	Cron config files .....	35

4.2.11	TCPwrapper .....	36
4.2.12	SSH security .....	37
4.2.13	Logging.....	40
4.2.14	Integrity check software .....	42
4.2.15	Policy for security patches .....	44
4.2.16	Back-up and recovery policy .....	44
<b>5</b>	<b>Audit results.....</b>	<b>46</b>
5.1	Environment description .....	46
5.2	Scope of the audit .....	47
5.3	Risk analysis .....	47
	Scan from Intranet (internal scan) .....	49
5.5	Port scan of the web server from Internet .....	50
5.6	Vulnerability scan of the web server .....	51
5.7	Results of local audit .....	52
5.7.1	Installed packages .....	52
5.7.2	OS installed patches .....	53
5.7.3	Network services .....	53
5.7.3.1	inetd services .....	53
5.7.3.2	Boot services .....	54
5.7.3.3	processes .....	55
5.7.4	Kernel tuning .....	56
5.7.4.1	Network parameters .....	56
5.7.4.2	User stack .....	58
5.7.4.3	File descriptor .....	58
5.7.4.4	Core file .....	59
5.7.5	File system .....	59
5.7.5.1	partitions.....	59
5.7.5.2	Basic permission .....	60
5.7.5.3	World-write files.....	60
5.7.5.4	SUID files .....	61
5.7.6	Account and Password .....	62
5.7.7	Console.....	63
5.7.8	Umask.....	63
5.7.9	Path.....	64
5.7.10	Cron .....	65
5.7.11	TCPwrapper .....	65
5.7.12	SSH .....	66
5.7.13	Logging.....	67
5.7.14	Integrity software .....	69
5.7.15	Policy for security patches .....	69
5.7.16	Backup and recovery policy .....	70

<b>6</b>	<b>Evaluation of the system .....</b>	<b>71</b>
<b>7</b>	<b>Evaluate the audit .....</b>	<b>74</b>
<b>8</b>	<b>References .....</b>	<b>75</b>
<b>9</b>	<b>Appendix A: aide.sh example file .....</b>	<b>78</b>
<b>10</b>	<b>Appendix B: example of syslog.conf file .....</b>	<b>79</b>
<b>11</b>	<b>APPENDIX C: Matrix of risks assessments .....</b>	<b>81</b>
<b>12</b>	<b>APPENDIX D: results of # pkginfo -I.....</b>	<b>85</b>
<b>13</b>	<b>APPENDIX E: results of # ./PatchCheck .....</b>	<b>95</b>
<b>14</b>	<b>APPENDIX F: /etc/system file .....</b>	<b>97</b>
<b>15</b>	<b>Appendix G: World-write files .....</b>	<b>97</b>
<b>16</b>	<b>APPENDIX H: SUID/SGID files .....</b>	<b>99</b>

© SANS Institute 2000 - 2002, Author retains full rights.

## **PART ONE: Research in Audit, Measurement Practice, and Control**

---

### **1 Introduction**

This audit will consist in evaluating the security of a Solaris 8 platform hosting a web server.

The purpose of this audit is to compare the initial setup/configuration of the OS against a predefined security checklist. To elaborate this checklist, I had to start with security requirements, that have been agreed by the management of our IT department. These requirements were an answer to a growing need for application web servers: we are a Telecom company, and more and more we have to make our services available to Internet users. Our standard way is to offer such services via a web application stored on a web server. This server is located on a screened subnet (i.e. subnet protected by "permissive firewall" from Internet, and separated by another firewall from our Internal backbone). Then, the application on the web server communicates with an application server (hosting the core of the application/service) located on a secure zone.

It's important to understand the scope of this audit. What will be checked, is the OS initial setup and configuration with the web server installed but before the web application is installed (by web application we mean all html pages and the binaries and/or CGI -scripts invoked through these html pages). Furthermore, the configuration of the web server will not be part of this audit. In our security processes, we have decided that before allowing the installation of the web application, the OS must comply to the security audit checklist derived from the security requirements. The web application security and the network elements of the screened subnet (switches, routers and firewall), will not be part of this study.

This audit will be done "manually" i.e. I'll review a step-by-step checklist to report on the OS compliance. As it was the first time we audited such servers against this checklist, doing it manually provides a great opportunity to really understand the inner workings of the system i.e. where and how security is configured, what to check, what else is affected, what security tools are used, etc. Sometimes, automated tools may hide the complexity and subtleties of the system and doing it by hand provides a more accurate judgement on what is secured and what is not.

### **2 State of practice**

In the following paragraphs, I first explain, the research technique I used to find relevant information for the elaboration of my Solaris security audit checklist.

Then for each source, I show what items could be directly used and what are the points that have to be improved or modified accordingly to our specific needs.

#### **2.1 Research technique**

To find relevant information about a Solaris security audit checklist, I started to check the documentation available from known and recognized organizations dealing with IT security (e.g. CIS, SANS,

CERT, NSA, ISSA, etc.). I found two important documents: the CIS Solaris benchmark [1], and the UNIX Security Checklist from the CERT and AusCert. The former provides a clearly defined list of tasks to improve system security which can be performed without jeopardizing mission-critical applications. The latter is a comprehensive security checklist for a wide range of UNIX hosts. Both works have undergone substantial peer review and testing from many different organizations and security experts, which somehow guarantee their reliability and completeness.

Another good and reliable source is the SANS documentation. The support material I had from the SANS courses I attended in London, the free available Solaris checklist (c.f. [3]) and some of the practical works done by GCUX certified students (c.f. <http://www.giac.org/GCUX.php>) constituted a great contribution for the elaboration of my checklist.

Then, a logical step was to look at the vendor's available documentation. For few years, Sun Microsystems, has provided good Solaris security documents published under its famous "BluePrints" series (c.f. [4], [5] and [9]).

Since two years, I've been following the works done around the YASSP tool. The knowledge and experience I gather through the use of this tool and the available documentation about it, have been of great help for my audit (c.f. <http://www.yassp.org/>).

I also performed a Google search with the words "Solaris security checklist" which gave 6'840 hits. By selecting the most relevant ones (c.f. [20], [21] and [22]), I was able to find useful information. I also made more specific searches when I needed precise information on very specific topics.

Finally, I looked at different available books dealing with Unix and Internet security. I just mentioned them here for sake of completeness. However, I did not find them very helpful for my OS security audit. First, because most of the time they include generic information and principles that I already knew. Second, when there is specific information about an OS security, it's sometimes already out-of-date when the book is commercially available. Most relevant books that I found are referenced in [23] and [24].

## **2.2 What can be improved and how**

As a primary source, I used the Solaris Benchmark v1.0.1b from the Center for Internet Security [1]. It's recognized as being a reference in Solaris security by the IT community. Moreover, I appreciated the completeness and the logical structure of the document, that help me to have a guiding line for my checklist. However, this paper is not written as an audit checklist, but more as a manual hardening checklist for Solaris. Therefore, some efforts are required to "translate" it into an audit checklist (i.e. the command used, the tests, etc.). Another point is that this document is written for generic security purposes and doesn't answer to specific security requirements. For example, for the network services they say "If you don't need this service turn it off, if you need it then configure it this way...". Our requirements, will permit to decide whether or not a service must be running. This rationale is not done in the CIS document, because they don't have requirements addressed for a specific situation. In addition, I found that some points were missing or not enough developed. For example, there is no recommendations about the Sun packages to install or de-install. Furthermore, the Kernel tuning parameters are a bit overlooked and finally there is no measures mentioned about the potential danger of SUID files.

The Unix Security Checklist has been published jointly by The Australian Computer Emergency Response Team (AusCERT) and the CERT<sup>®</sup> Coordination Center (CERT/CC) and details steps to improve the security of Unix Operating Systems. This is also a very comprehensive security checklist on which I based my work. However, this paper mixes an auditing process (some points are listed as audit steps: “*ENSURE that...*”), with configuration process (some points are listed as: “*DO this...*”). In addition, this list is not dedicated to Solaris, and sometimes the information provided is UNIX-generic where no specific commands are mentioned. There is also a lack of information on SUID files and Solaris minimum set of required packages. The Sun’s BluePrint on Solaris Operating Environment Security [4], gives good explanations and background on most of the security measures listed in the CIS and CERT documents.

The SANS Solaris checklist [3], helped me to turn the CIS benchmark security measures into a security audit checklist. However, this list doesn’t provide enough detail on how to test every point (e.g. the specific command to execute).

The Sun’s BluePrint on Operating Environment Network Settings for Security [5], gives details on the Kernel Parameters settings, which the other previously mentioned documents lacked for. In addition I used two relevant documents [11] and [17], obtained after sorting the results of a Google search with the words “Solaris Kernel tuning”.

In the above mentioned sources, no valuable information about packages to install or de-install was present. By parsing GIAC practical works, I found some interesting advices about SUN packages to install regarding some security needs. For example, in [6] and [7] one can find what additional packages are needed to support special applications like NTP and SSH. A good source for Solaris 8 core packages used for firewall security requirements can be found at Lance Spitzner’s web site [8]. In addition, the SUN BluePrint about the OS minimization for security [9] explains the role of the different packages.

I’ve been working with Sean Boran ([www.boran.com](http://www.boran.com)) on system hardening (YA SSP tool, manual hardening, etc.) and auditing. Thanks to this experience, I was able to make a list with the minimum set of SUID files required regarding to specific situations. An example of such list can be found at [17]. By doing a Google search, I found interesting information on SUID files at University of Waterloo, California [18].

Finally, I obtained 3 relevant security checklists with the Google web search ([20],[21] and [22]). However, most of them were out-of-date (apply to older version of Solaris), lacked information on SUID files and packages, and were installation checklists. I only used them to make a crosscheck with my own checklist.

To summarize, the main improvements were:

- Compile information from different sources (initial checklist, SUID files, packages, kernel tunings, commands, etc.)
- Turn installation checklist into audit checklist (using the right commands and tools for testing)
- Define what should be left on the machine (files, processes, services, etc.) according to our specific security requirements



### 3 Security requirements

For application web servers described in the introduction chapter, we have agreed in our company that the server should meet these security requirements:

- Req1.** The strict minimum set of services shall be available to Internet, i.e. only HTTP and/or HTTPS (SSL) to reduce the risks from Internet.
- Req2.** The strict minimum set of services shall be available to Intranet, i.e. only HTTP, HTTPS (SSL) and management traffic, to reduce the risks from inside our company (note that in the Intranet there are more 10'000 users!).
- Req3.** The management shall be made from Intranet and never from Internet. The communication between the management station(s) and the server shall be secured in terms of authentication, integrity and confidentiality.
- Req4.** The minimum software shall be installed on the machine.
- Req5.** The minimum processes and services should run on the machine hosting the web server to follow the rule: the less processes are running, the less chance we have to have a vulnerability.
- Req6.** Strong account and password policy shall be in place.
- Req7.** Good logging policy shall be in place to permit to rapidly track break-in attempts (reducing the detection time!).
- Req8.** The configuration shall be tightened following good practice to reduce the effectiveness of exploitable vulnerabilities.
- Req9.** The least privilege principal shall be applied whenever possible.

Based on these security requirements I was able to develop the audit security checklist.

### 4 The security audit checklist

In the following paragraphs, I will describe each item that has to be checked. Then the commands used for the tests are explained each time it's needed.

In addition, I will mention each time if the test is subjective or objective.

When the test is a UNIX command, a “#” will be added at the beginning of the command line to indicate it.

#### 4.1 Scan

Prior to the local audit of the machine hosting the web server, a port scan and remote vulnerability scan are performed for the following reasons:

- it gives a general idea of the degree of exposure of the web server
- it shows network services reachable from outside and thus it helps prioritizing the risks

- can be used to compare the accuracy of the firewall filtering rules with the open services on the server
- it's a good "tool" to convince sysadmin and managers to take action against a potential security hole (if you show them a remote vulnerability scan they are much more impressed than if you show them a hole when you're already logged on the machine).

#### 4.1.1 Port scan

The port scan is done both from Internet and from the internal network (Intranet).

Since I scan only one machine, all TCP and UDP ports are scanned.

The port scan is done by the nmap program, launched from a Linux (version 7.1) machine. The `-O` switch tell nmap to guess the operating system. The `-p` switch followed by a number is used to look for the specified port. When `1 -` is specified (number one followed by minus sign), nmap will scan all 65535 TCP ports. When in addition the switch `-sU` is present nmap will scan all UDP ports.

Machine that initiated the scan	Commands to test	expected results
External machine (Internet)	# nmap -O -p 1- <IP# of server>	The only services in open state shall be http on tcp port 80 and https on port 443. The SSH should be in state filtered. OS should not be guessed
External machine	# nmap -sU -p 1- <IP# of server>	No UDP services shall be opened
Internal machine (Intranet)	# nmap -O -p 1- <IP# of server>	Open services should be limited to port 80 (http), 443 (https), and 22 (ssh). OS should not be guessed.
Internal machine	# nmap -sU -p 1- <IP# of server>	No UDP services shall be opened

Note: sometimes firewalls don't allow to ping the machines they protect. Before doing the nmap scan, I'll ping the machine to see if ping are allowed by the firewall. If ping is stopped, I'll add the `-P0` switch in the nmap command. It will tell nmap not to try to ping the server, before scanning it.

The scan from Internet should show the strict minimum set of available network services i.e. only HTTP and HTTPS. No ports shall be open for remote management from Internet. All remote accesses are centralized via company wide access gateways (it can be either from dial-in connection, or from Internet). Then these connections are routed internally (Intranet).

The scan from Intranet may have other ports opened for management purposes (other than SSH). However, every additional network services used for management shall meet Req3 security requirement, this need shall be justified by the sysadmin and it shall be approved by the management.

Finally, if the system was correctly hardened nmap should not be able to guess the OS.

#### 4.1.2 Vulnerability scan

The vulnerability scanner used is Nessus. The Nessus server is launched from the same machine as nmap i.e. the Linux 7.1. The tests are also made both from Intranet and from Internet. Note, that the latest plugins (corresponding to the latest vulnerabilities) have to be downloaded just before launching the Nessus vulnerability scan against the web server.

From the results of the nmap scan, I will reduce the number of vulnerability to test to the corresponding open ports I found.

From Internet, no vulnerability shall be found. This would imply that the latest security patches have been applied. If a vulnerability is found at this stage it should be immediately fixed without waiting for the end of the audit.

From Intranet, if a vulnerability is found it should be fixed rapidly. While, this is less serious, it can be internally exploited.

### 4.2 Local audit

#### 4.2.1 Installed packages

##### 4.2.1.1 Selection of required packages

Following the Req4 requirement, only the minimum required packages shall be installed on the audited machine. What we need is the minimal set of packages to run the OS, SSH, and HTTP(S).

In addition, few other packages will be needed to run certain utilities that are vital for management.

Regarding the SUN's BluePrint [9], the minimum required packages for the OS to run are:

Solaris 8 OE running in 32 bit mode requires the following packages:

Package Name	Description
SUNWcar	Core Architecture, (Root)
SUNWcsd	Core Solaris Devices
SUNWcs1	Core Solaris, (Shared Libs)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWesu	Extended System Utilities (required for system commands like awk, last, etc.)
SUNWhmd	SunSwift SBus Adapter Drivers used at boot to configure the internal network interfaces
SUNWkvm	Core Architecture, (Kvm)

SUNWlibms	Sun WorkShop Bundled shared libm (system libraries)
SUNWloc	System Localization (used for the system installation with the JASS toolkit)
SUNWnamos	Northern America OS Support
SUNWpd	PCIDrivers for booting
SUNWlibC	Sun Workshop Compilers Bundled libC

In addition if a system is running in a 64 -bit mode the following additional packages are required:

Package Name	Description
SUNWcarx	Core Architecture, (Root) (64 -bit)
SUNWcs1x	Core Solaris Libraries (64 -bit)
SUNWcsxu	Core Solaris (Usr) (64 -bit)
SUNWesxu	Extended System Utilities (64 -bit)
SUNWhmdx	SunSwift SBus Adapter Drivers (64 -bit)
SUNWkvmx	Core Architecture (Kvm) (64 -bit)
SUNWlmsx	Sun WorkShop Bundled 64 -bit shared libm
SUNWlocx	System Localization (64 -bit)
SUNWnamox	Northern America 64 -bit OS Support
SUNWpdx	PCI Drivers (64 -bit)
SUNWlibCx	Sun WorkShop Bundled 64 -bit libC

The following packages are required for patch management:

Package Name	Description
SUNWswmt	Install and Patch Utilities
SUNWgzip	GNU zip (gzip) compression/uncompressing utility
SUNWadmc	System administration core libraries (needed if showrev is used)

The following packages are required for OpenSSH installation:

Package Name	Description
SUNWzlib	The Zip compression library

Following the requirement Req7, the accurate time shall be guaranteed in order to be sure of the date and time of the logged events. For this, the NTP package should be installed. In addition, to be able to send alarm via e-mail the sendmail package should be installed.

Package Name	Description
SUNWntpr	NTP (root)
SUNWntpu	NTP (user)
SUNWsndmr	Sendmail (root)
SUNWsndmu	Sendmail (user)

The use of perl is generally useful for writing scripts. If such scripts are needed for management purposes, then the following packages should be installed:

Package Name	Description
SUNWlibm	Sun WorkShop Bundled libm
SUNWlibms	Sun WorkShop Bundled shared libm

The Man pages are also useful:

Package Name	Description
SUNWdoc SUN Wman	On-line manual pages

If some software have to be compiled (e.g. Apache), then the following packages are required:

Package Name	Description
SUNWarc	Archive Libraries
SUNWarcx	Archive Libraries (64 -bit)
SUNWbtoox	CCS libraries bundled with SunOS (64-bit)
SUNWbtool	CCS tools bundled with SunOS
SUNWscpx	Source Compatibility (Usr) (64 -bit)
SUNWdplx	Developer Profiled Libraries (64 -bit)
SUNWsprox	Sun WorkShop Bundled 64 -bit make library
SUNWhea	SunOS Header Files
SUNWlibm	Sun WorkShop Bundled libm
SUNWdfbh	Dumb Frame Buffer Header Files
SUNWcg6h	GX (cg6) Header Files

#### 4.2.1.2 How to do the test

In addition to the above mentioned packages, it's likely that other packages should be installed. First, we will have the packages for web application. Second, if we have special hardware (e.g. Ethernet card with 4 ports), then additional packages are required. However, what is important is the method adopted for package installation. At minimum, the following steps should have been observed:

- Step1. installing the core packages (mentioned in the previous paragraph)
- Step2. installing the additional packages for the required management tools and application (the one mentioned in previous paragraph)
- Step3. if additional packages are needed for the functioning of the application, they can be installed.
- Step4. if packages are needed for a one time installation (e.g. the compilation packages needed to compile apache), then after their use they should be removed.
- Step5. Sysadmin have to understand what installed packages are for.

To have an idea of the number of installed packages use the following command:

Command (subjective)	expected results
# /usr/bin/pkginfo -i   wc -l	number of packages: quickly gives an idea if only selected packages have been installed

`pkginfo` displays information about software packages that are installed on the system or that reside on a particular device or directory. When used with the `-i` option, it displays information for fully installed packages only. The `wc` utility reads one or more input files and, by default, writes the number of newline characters, words and bytes contained in each input file to the standard output. When used with the `-l` option it counts the number of lines only.

Note, that even if this test uses a command, it's a subjective test, because we can't say "*if the number of packages is equal to..., then the test is OK*". Rather, it gives an idea or a feeling, if a selective policy has been applied for the installation of packages.

To check if a strategy (or policy) is in place:

Test (subjective)	Expected results
Ask sysadmin for package installation strategy	Similar steps as described above (Steps1 -5), shall be observed

To check for installed packages type:

Command to test (objective)	Expected results
-----------------------------	------------------

---

# /usr/bin/pkginfo -i	No unnecessary packages shall be installed.
Check all installed packages and see if they are necessary	

---

## 4.2.2 OS Installed patches

It is very important to check that security patches are up-to-date. Different tools exist, that test if installed patches are currently up-to-date. Some free tools are provided on the net so you can quickly determine the patch level and easily apply missing patches that the vendor has recommended.

CheckPatches is a Bourne shell script for Solaris patch management developed at the University of Waterloo, California, with the help of colleagues on the net (CheckPatches was originally a Perl script posted to Usenet by Bruce Barnett). The scripts rely on the vendor's patch report to construct an incremental list of patches required. These tools have been peer reviewed by Sean Boran ([www.boran.com](http://www.boran.com)) and other participants of the [YASSP](http://www.yassp.org) project, and can help us manage patches on Solaris system. This tool can be downloaded from <http://ist.uwaterloo.ca/security/howto/2000-12-04/patches.tar>.

CheckPatches makes a comparison between installed patches on the system with a Patch Report available from SUN at <http://sunsolve.sun.com/pub/patches/Solaris8.PatchReport>, updated twice a month with the latest available patches.

Sun Microsystems has a tool available at <http://sunsolve.sun.com/pub/cgi/show.pl?target=patchk>, that works on the same principle.

---

Test (objective)	Expected results
Download the tool and the latest patch report from SUN and type: # ./PatchCheck	Patches shall be up-to-date

---

Note: SUN's patch reports will often list patches that cannot apply to the system either because they apply to hardware drivers that are not part of the system or because they apply to packages not installed on the machine. Of course, these patches should not be taken into account.

As part of the audit the patching policy shall also be considered. This point will be covered later in this document (c.f. § 4.2.15)

## 4.2.3 Network services, processes and daemons

### 4.2.3.1 Network services started by inetd

To start a service, the inetd daemon looks for non-commented lines in configuration file /etc/inetd.conf.

Following the requirements Req1 and Req2, HTTP, HTTPS (SSL) and SSH are the only services allowed to run. As these 3 services are normally not running through inetd (and should not), no services shall be started through the inetd daemon. This implies that the whole /etc/inetd.conf file shall be commented, i.e. every line of this file shall start with a “#” symbol. In addition, the inetd demon should not be running.

Commands to test(objective)	Expected results
# <code>egrep -v "^#" /etc/inetd.conf</code>	nothing shall be displayed
# <code>ps -ef   grep inetd</code>	nothing shall be displayed
# <code>pgrep inetd</code>	nothing shall be displayed
Crosscheck with <code>lsof -i</code> and <code>netstat -a</code> commands that no inetd service are running (see § 4.2.3.2 for details on this test)	no inetd service shall be displayed
Crosscheck with <code>ps -ef</code> command that no inetd service are running (see §4.2.3.3 for details on this test)	no inetd service shall be displayed

The `egrep` command search a file for a pattern using full regular expressions. When used with the `-v` switch it prints all lines except those that contain the pattern. The pattern “^#” means line beginning with the “#” character.

The `ps` command is used to display information about processes. The `-e` switch lists information about every process running when the command is typed. The `-f` switch generate a full listing of information on processes (UID, PID, C, STIME, TIME, CMD).

The `pgrep` command find or signal processes by name and other attributes. When used with no switch it returns the process IDs of the active process on the system whose name matches the one specified on the command line.

#### 4.2.3.2 Boot services

As previously mentioned only 3 network services shall be started through RC scripts: the web server daemon, the SSL daemon and SSH daemon.

Both `netstat` and `lsof` command will be used. Only the 3 above mentioned services should appear. If other services are displayed they should be discussed with sysadmins if they are necessary or not. Then the security of these services should be assessed, and based on these results, the managers should agree to let these additional services run.

The `lsof` command lists information about files opened by processes. When the `-i` switch is used, this option selects the listing of files any of whose Internet address matches the address specified after the switch. If no address is specified, this option selects the listing of all Internet and x.25 (HP -UX) network files. In other words, we will see the binary name of the running network service (e.g. `sshd`) along with their connection status and port number (if any).



netstat commands shows the content of network related data structures. When used with the `-a` it shows the status of all sockets. When looking under the TCP and UDP socket tables, we will see the running network services.

Commands to test(objective)	Expected results
# <code>lsof -i</code>	Only SSH, SSL, and HTTP should appear, check that no inetd services are displayed
# <code>netstat -a</code>	Look under TCP and UDP tables. Only SSH, SSL, and HTTP, HTTPS (SSL) should appear, check that no inetd services are displayed
Crosscheck with <code>ps -ef</code> command that not only the mentioned services are running (see §4.2.3.3 for details on this test)	Only SSH, SSL, and HTTP should appear as network daemon
Reboot the system and redo the above tests	Same results shall be observed.

Note: it's important, to verify that when we reboot the machine, no new services are started. This ensure that services have not been disabled manually (just killing a running process), but have been removed from the boot services i.e. removed from the `rc` scripts.

In addition to the above tests, two other subjective tests are important: I have to evaluate the knowledge the sysadmin has with regards to the running network services, and check the procedure (if any) for starting new services.

Test (Subjective)	Expected results
Does the sysadmin know how running service are working, starting, does he know the corresponding config files, the security implications, etc.	Detail knowledge
Check the procedure to start new network services (who gives the authorization, who checks the security implication...)	A written procedure.

#### 4.2.3.3 Processes

With respect to the requirement Req5 , the minimum set of processes shall be running on the machine. Most of them are started through the RC scripts. They are stored under the famous `/etc/rc*.d/` directories.

We should see the minimal processes for the kernel, and the processes corresponding to the network services running. Other unexpected processes shall be investigated and their presence understood and justified by the sysadmin. If they are necessary and harmless in terms of security they can be left running.

To look for current running processes I use the command `ps -ef`.

Commands to test(objective)	Expected results
# ps -ef	<p>The following processes may appear:</p> <p><code>sched</code>  <code>/etc/init -</code>  <code>pageout</code>  <code>fsflush</code>  <code>/usr/lib/saf/sac -t 300</code>  <code>/???/??? /bin/httpd (or apache)</code>  <code>/usr/lib/sysevent/syseventd</code>  <code>/usr/lib/sysevent/syseventconfd</code>  <code>/usr/sbin/syslogd -t</code>  <code>/usr/sbin/cron</code>  <code>/usr/sbin/vold</code>  <code>/usr/lib/utmpd</code>  <code>/usr/lib/sendmail -q15m</code>  <code>/usr/local/sbin/sshd</code></p>
Reboot the system and redo the above test	same processes shall be displayed

**The following explanation of the processes are borrowed from [26]:**

*sched* is the first process running. It is referred to as the swapper. This process is responsible for operating system scheduling, and swapping out light weight processes when necessary to run higher priority processes. From this process, the scheduling of and swapping of processes on the system is controlled.

*init* is the process that is responsible for the execution of all processes at their respective run levels. At bootstrap time, *init* is the first process started. From its execution, *init* reads the `/etc/inittab` and `/etc/default/init` and follows the instructions in those files: it starts all other processes, and brings the machine to its default run level (for Solaris, this is run level 3).

*pageout* is the next process in the sequence. It is used to control the paging out of memory to disk, and back in again.

*fsflush* is a daemon responsible for writing data back to the disks. The kernel checks superblocks on a 30 second interval, and the data in the superblock is either idle or unchanged, the kernel uses *fsflush* to clear the superblock and send the information back to the disks.

*sac* is the Service Access Controller, and is started when the system enters multiuser mode. *sac* is a program designed to watch ports on a Solaris system. It can provide statistics on port use, poll for failure, restart port monitors that fail, and a variety of other functions.

*devfseventd* and *devfsadm* we'll cover together, as they're dependent upon one another. These two daemons are part of the Solaris 8 device management package. *devfseventd* is the kernel event notification daemon. This daemon runs on the system, monitoring the kernel for things such as when device nodes are added and removed from the kernel device tree. *devfsadm* is the replacement to the antiquated programs such as *disks*, *tapes*, and *devlinks* (all part of the old *devfs* tools). *devfsadm* builds the links in the */dev* and */devices* directories. All the old *devfs* tools are now symbolic links to *devfsadm*.

*syslogd* is the system logging daemon. This daemon is responsible for monitoring and logging system events, or sending them to users on the system. *syslogd* is a critical application on every system, and is configurable with the */etc/syslog.conf* file.

*vold* is the volume manager. This neat little daemon manages the system cdrom and floppy. When media is inserted into either the cdrom or the floppy drive, *vold* goes to work and mounts the media automatically. Configuration information for this utility is in */etc/vold.conf*. This process shall be left only if a good physical security is ensured for the machine.

*cron* is a system scheduling utility. *Cron* is capable of executing events for specific users on a predetermined time schedule if entries are made into the users *crontab*.

*utmpd* is the *utmp* and *utmpx* monitoring daemon. *utmp* has been obsoleted by *utmpx*, but for reverse compatibility, it exists. To cut to the chase, *utmpx* is used to record the current users on a system. When a user terminates a process or logs out of a system, *utmpd* polls these files to ensure the entries for these events has been removed. Should the entries still exist in the files, *utmpd* removes them.

*ttymon* is a port monitor for terminal ports. This process is usually used in conjunction with *sac*. This facility controls TTY settings to users and services.

*sendmail -q15m* is the Mail Transport Agent used on Solaris 8. With the *-q15m* option specified *Sendmail* is left running in Queue mode: i.e. it will deliver but not accept remote emails for security reasons. We don't want to have a mail server running on this machine (because it's too dangerous). However, the machine might be able to send e-mail especially for alarms to the *sysadmin* official email box.

*httpd* is the daemon for the web server. Depending on the product, the demon might have another name (e.g. *apache*).

*sshd* is the daemon for the SSH server.

Like for network services, I have to evaluate the knowledge the sysadmin has with regards to the running processes/daemons, and check the procedure (if any) for starting new processes/daemons.

Test (Subjective)	Expected results
Does the sysadmin know to what the processes/daemon correspond to? Does he know the security implications	Detail knowledge
Check the procedure to start new processes/daemons (who gives the authorization, who checks the security implication...)	A written procedure.

## 4.2.4 Kernel tuning

### 4.2.4.1 Network parameters

Solaris includes a lot of low level network parameters, that can be adjusted according to specific situations. Amongst them, some may be tuned in order to provide greater protection against various known network attacks. According to the requirement Req8, these parameters should be set to values that guarantee a better level of protection.

Most of the explanations about these parameters have been borrowed from the SUN's BluePrint [5], because the points are clearly detailed.

Note, that I apply the test both to IP and IPv6 network parameters. Even if IPv6 is not yet used, it's safer to have everything secured also for IPv6 in case it's deployed in a near future.

The `ndd` command has to be used to read and sets these configuration parameters in the corresponding kernel drivers. Currently, `ndd` only supports the drivers that implement the TCP/IP Internet protocol family. Each driver chooses which parameters to make visible using `ndd`. When `ndd` command is used with `-set` option it sets the value to the specified value. When no option is specified, the `ndd` command reads the value of the parameters from the driver. I will test parameter values from three drivers: the TCP driver (`/dev/tcp`), the ARP driver (`/dev/arp`) and the IP driver (`/dev/ip`).

#### 4.2.4.1.1 ARP defenses

Here, I check the parameter values that can be set to ensure a certain protection against known ARP attacks.

According to SUN (c.f. p.7 of [5]): “*There are two basic types of attacks possible with ARP: denial of service and spoofing. An attacker can feed a remote system incorrect address information as well. This is known as cache poisoning. Since the ARP layer always trusts the information it receives (all address information received by a system is believed to be accurate.), wrong information can be inserted and current ARP entries can be corrupted. An attacker may use the publish feature of the ARP layer to broadcast incorrect information about other systems. If two ARP replies are received, at least one will be used. It may be the correct one, or it may not. This situation can spread discord through* h-

out systems on the local network and be difficult to diagnose. ARP spoofing attacks are more serious because they are used to compromise remote systems on the local network. By masquerading as another system, it is possible for an attacker to exploit a trust relationship and gain entry to other systems. This attack involves sending false hardware address information to a target system which the system will use to update its ARP tables. Once the false information is implanted, the attacking system changes its IP address and attempts a connection to the target.”

To reduce the effectiveness of these attacks, we can set the interval time between two deletions of the ARP and IP tables to a low value. Thus, if false information has been stored, it will be quickly deleted. The recommended value is one minute for each (note: values are specified in milliseconds).

Check the values for the ARP cache and IP table cleanup interval

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/arp arp_cleanup_interval	60000
# /usr/sbin/ndd /dev/ip ip_ire_arp_interval	60000

#### 4.2.4.1.2 ICMP defenses

ICMP broadcasts are, at times, troublesome. A significant number of replies to a ICMP broadcast from all systems on a network could cause significant network performance degradation. An attacker may use ICMP broadcast requests to initiate a denial of service attack. It is best to disable the ability to respond to all type of ICMP broadcasts.

A well known ICMP broadcast is the echo request sent to a broadcast address, generated by the famous ping command (e.g. # ping 188.105.33.255 ). If all machines of a subnet are configured to respond with an echo reply, it can generate lots of traffic. Even worse, because echo reply will return the same data payload as in the echo request (that’s how the protocol was designed), if this payload is too large, the responding machine will fragment its response across several packets, further increasing the network load. Thus, I have to ensure that the machine doesn’t respond to such requests:

Commands to test(objective)	Expected result
# /usr/sbin/ndd /dev/ip ip_respond_to_echo_broadcast	0

The equivalent of echo request broadcast in IPv6 is echo request multicast. I have to test that it’s also disabled:

Commands to test(objective)	Expected result
# /usr/sbin/ndd /dev/ip ip6_respond_to_echo_multicast	0

Another ICMP broadcast is the timestamp request broadcast. There is no need to answer to such request, because we use the NTP protocol for all systems in our company, for the time synchronization. Ensure that these answers are disabled:

Commands to test(objective)	Expected result
# /usr/sbin/ndd /dev/ip ip_respond_to_timestamp_broadcast	0

Address mask broadcast are also ICMP broadcast request. They are normally used by diskless systems (e.g. printers) sent during a boot. I have to ensure that the server does send response to such requests:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/ip ip_respond_to_address_mask_broadcast	0

Another source of trouble with ICMP protocol is the redirect features.

According to SUN (c.f. p.10 of [5]): *“Redirect errors are used by a router to inform a host sending data, to forward the packets to a different router. Both routers involved in the redirection must be connected to the same subnet. The sending host will then install a new host routing entry in the routing table for the destination host. Unlike ARP entries, these will not time out and be deleted. Most systems check the redirect message for errors and potential problems prior to modifying the routing table. [...] An attacker may forge redirect errors to install bogus routes. This may initiate a denial of service attack if the newly specified router is not a router at all. There are rules governing valid redirect errors, all of which can be spoofed easily.”*

I have to ensure that both sending and accepting ICMP redirect message are not allowed (both for IP and IPv6):

Commands to test(objective)	expected results
# /usr/sbin/ndd /dev/ip ip_ignore_redirect	1
# /usr/sbin/ndd /dev/ip ip6_ignore_redirect	1
# /usr/sbin/ndd /dev/ip ip_send_redirects	0
# /usr/sbin/ndd /dev/ip ip6_send_redirects	0

Finally, as mentioned previously, for time synchronization the NTP protocol is used so we don't need ICMP Timestamp at all. To ensure that answers to ICMP timestamp request are disabled

Commands to test(objective)	Expected results
-----------------------------	------------------

# /usr/sbin/ndd /dev/ip ip_respond_to_timestamp	0
---	---

#### 4.2.4.1.3 IP defenses

In this paragraph I will check that the routing capabilities of the Solaris are turned off, first because the role of the audited machine is hosting a web server and not being router-like machine, and second because most of these features have been exploited to generate attacks.

IP forwarding is a routing functionality that is used to transfer packets from one interface to another. Generally, when servers have two IP interfaces, one is public and the other is private (used for management purposes). This feature can be exploited by an attacker to access the network attached to the private interface.

To ensure that this feature is disabled the following parameters have to be checked:

Commands to test(objective)	expected results
# /usr/sbin/ndd dev/ip ip_forwarding	0
# /usr/sbin/ndd dev/ip ip6_forwarding	0

Another attack is still possible even when IP forwarding is disabled: if an attacker sends a packet to the public interface with a spoofed private address (i.e. the source address of the packet contains an address of the private network), the machine may send the packet to the private interface believing this packet comes from the private network. Solaris has a feature that stops such packets: the system is aware from which interface the packet arrives, and if the source address corresponds to other interface it will drop it. To ensure that this feature is turned on:

Commands to test(objective)	Expected results
# /usr/sbin/ndd dev/ip ip_strict_dst_multihoming	1
# /usr/sbin/ndd dev/ip ip6_strict_dst_multihoming	1

Another routing functionality of the Solaris platform is the possibility to forward directed broadcast. According to SUN (c.f. p.14 of [5]): *"A directed broadcast is a unicast datagram from a system on a remote network addressed to all systems on another network. Once the datagram reaches the router connected to the intended network, the datagram is forwarded to all systems as a data-link layer broadcast. Directed broadcasts can be problematic due to the amount of network traffic generated by broadcasts and the ability to send a packet to all systems on a network."*

An attacker may take advantage of forwarded directed broadcasts to attack and probe systems. [CERT Advisory CA-98.01](#) describes a denial of service attack called the smurf attack after its exploit program. It involves forged ICMP echo request packets sent to broadcast addresses. The source address

*in the forged packet is set to a target. The result is that the target and intermediate routing systems forwarding the directed broadcasts suffer from network congestion .”*

To ensure that this functionality is disabled:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/ip ip_forward_directed_broadcasts	0

Another routing functionality is the possibility to forward source routed packet. These packet includes the route to follow. Normally, there is no need for such packets, as properly configured routers take care of correct routing. However, source routed packet can be used by an attacker to bypass some filtering routers and/or firewalls. Therefore, I should check that this feature is disabled:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/ip ip_forward_src_routed	0
# /usr/sbin/ndd /dev/ip ip6_forward_src_routed	0

#### 4.2.4.1.4 TCP defenses

The SYN flooding attack (<http://www.cert.org/advisories/CA-1996-21.html>) consist in opening a lot of TCP connections (SYN packet) with unreachable source IP addresses. The target machine replies with a SYN+ACK packet and waits for the ACK packet (3-way handshake). Because of false source addresses, the target machine will continue resend the SYN/ACK packet, until a time limit is reached. Because of this, the backlog queue (it's the queue storing the half-open session) may fill up rapidly, and other connection requests may no longer be treated, creating a denial of service. While, it's not possible to completely stops these attacks it's possible to adjust the size of the backlog to a value that render the attack a lot more difficult to realize. The maximum value recommended by SUN is 4096:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_conn_req_max_q0	4096

Likewise, it's possible to make a connection exhaustion, with established connections. An attacker can open many connections to a server and hold them open for long periods of time, effectively pushing the server closer to its connection limit. This attack is however less common, because the connections can be traced back, and because the attacker need great resources. Here again, to mitigate the effectiveness SUN recommends a value of 1024.



Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_conn_req_max_q	1024

The 3 next parameters are more dedicated to web server. Because the HTTP protocol uses TCP connections in a specific manner, some timers have to be adjusted to strengthen the behavior of TCP.

According to SUN's reference manual for TCP/IP tunable parameters [11]: *"On a busy web server, there can be too many TCP connections in TIME\_WAIT state, consuming too much memory. In this situation, we can decrease the value for performance reasons."*

An attacker may generate an attack by improperly shutting down the sockets from the client side, thus creating a DOS.

There is an Internet draft [13] that proposes to avoid the use of this state for busy web server. According to them, performances can be increased up to 50%.

However, with HTTP and Solaris we can not yet bypass this state. According to SUN the minimum time we can set for this state for each connection is 60 seconds. To mitigate the effectiveness of the above mentioned attack and to increase the availability of the server, I have to check that this value has been set to minimum:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_time_wait_interval	60000

In the same way, the timer interval which prohibits a connection to stay in the `FIN_WAIT_2` state forever, may be tuned. The `FIN_WAIT_2` state is reached, if a connection closes actively. The `FIN` is acknowledged, but the `FIN` from the passive side didn't arrive yet - and maybe never will. Usually web servers and proxies actively close connections. A crashed or misbehaving browser may cause a server to use up a precious resource for a long time. Once again, this can be used by an attacker to make a DoS attack. To mitigate, the effectiveness of such exploit, this time should be kept as low as possible.

According to SUN, they advise not to go below 67500 ms. To counter the above mentioned threat and to increase the availability of the server, I have to check that this value has been set to this minimum. Note, that this value has been tested and recommended by IBM [13] for their web servers, and application servers, and by J-S Volkler[11].

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_fin_wait_2_flush_interval	67500

The last timer I have to look at is the TCP keep-alive interval. The *keep-alive* timer becomes significant for web servers, if after the client (browser) initiates a connection (*active open*), it suddenly crashes or terminates without the server knowing about it. This condition can be forced sometimes by

quickly pressing the stop button of Netscape or the Logo of Mosaic. Thus the *keep-alive* probes do make sense for web servers and can be sent with in the HTTP header. However, I have to make sure that the probes stop after a finite time, if a peer does not answer. IBM [13] recommends that this value should be set to as low as 300'000 milliseonds.

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_keepalive_interval	300000

TCP uses a hash table to locate TCP connection control blocks: it contains information on current state of TCP connections. If the size of the table is 512 (default on Solaris8), when Solaris has more than 256 connections, the hash is bypassed and a linear search of memory is required to locate the appropriate TCP data structure. At this stage the performances to process TCP connections decreases. Entries remain in the hash even when the connection is closed and is in the **TIME\_WAIT** and **FIN\_WAIT\_2** state.

To keep up the performance, Oracle [15], IBM [14] and the SYSADMIN journal [16] recommend values from 8192 to 32768. To check this parameter:

Commands to test(objective)	Expected results
# /usr/sbin/ndd /dev/tcp tcp_conn_hash   grep size	tcp_conn_hash_size=8192
# more /etc/system   grep tcp_conn_hash_size	set tcp:tcp_conn_hash_size=8192

note: Since Solaris 2.6, the tcp\_conn\_hash\_size parameter can no more be adjusted with the ndd command. This value can only be set in /etc/system. However, the current used value is shown at the beginning of the TCP hash table obtained with the command ndd /dev/tcp tcp\_conn\_hash.

Another TCP parameter that can be adjusted in Solaris is the method for generating the Initial Sequence Number (ISN).

According to SUN (c.f. p.19 of [5]): *"Predictable ISNs make it possible for attackers to compromise some systems. The TCP three-way handshake discussed previously involves two systems synchronizing sequence numbers prior to data exchange. For each new connection most systems use ISNs that have fixed and predictable counter increments. An attacker uses this knowledge to create a three-way handshake by predicting the required ISN to establish a connection and execute a command."*

On Solaris, 3 different methods exist to generate TCP ISNs. We have to ensure that the one used can guarantee the best randomness and uniqueness of TCP ISN, in order to prevent the above mentioned attack.

Commands to test(objective)	Expected results
# ndd /dev/tcp tcp_strong_iss	2

---

```
# more /etc/default/inetinit | grep TCP_STRONG_ISS= TCP_STRONG_ISS=2
```

---

Currently, the best available method regarding the above mentioned criteria in the method number 2. The value has to be checked both with ndd command, and has also to be set in the file /etc/default/inetinit. This guarantee that the method 2 is permanently used, even after a reboot (the inetinit file is checked at each boot)

The last TCP parameter to check is the one who permits or denies reverse source routing. When TCP source routed packets arrive at the destination machine, they may include a reverse route. The destination machine copies this route to all packets to be sent back to the originating machine. Once again, this routing functionality is useless in well configured networks, and can be exploited to generate DoS attacks. To ensure, this functionality is disabled:

Commands to test(objective)	Expected results
# ndd /dev/tcp tcp_rev_src_routes	0

#### 4.2.4.1.5 Persistency of network parameters

It's important to ensure that the values of these tested network parameters remain the same after rebooting the machine. Typically, a script has to be called at boot time that sets the desired values for the network parameters.

Test(objective)	Expected results
Reboot the machine and re-check the value of the parameters	Same value shall be observed
Check that the correct values of the parameters are set within a script called at boot time (in rc directories)	Such file must exists

#### 4.2.4.2 User stack

To prevent and log stack-smashing attacks (i.e. buffer overflow), it's safe to have the users' stack non-executable. This can be done by setting to 1 the two parameters noexec\_user\_stack and noexec\_user\_stack\_log.

Note the following quote taken from SUN's Blueprint (in [5] page 17):

*"Some security exploitation programs take advantage of the Solaris OE kernel executable system stack to attack the system. These attack programs attempt to overwrite parts of the program stack of a privileged program in an attempt to control it. In Solaris 2.6 OE and later, some of these exploits can be avoided by making the system stack non-executable. [...] This feature does not stop all buffer overflow exploitation programs, and it does not work on Intel x86-based or older SPARC hardware. Some*

overflow exploitation programs work on different principles which non-executable stacks cannot protect against.”

Commands to test(objective)	expected results
# more /etc/system   grep noexec_user_stack	set noexec_user_stack=1
# more /etc/system   grep no-exec_user_stack_log	set no-exec_user_stack_log=1

note: the setting is only available on SunOS 5.6 and later and it's only supported on the sun4u, sun4d and sun4m hardware platforms.

#### 4.2.4.3 File descriptor

In our case, we have a web server that communicates with an application servers. When busy, the server processes may open thousands of files or sockets. In the Solaris operating system, rlim\_fd\_cur sets the soft limit for the number of file descriptors per process, while rlim\_fd\_max is the hard limit.

Many common UNIX -based applications call the stdio fopen() library routine. If the application exceeds the rlim\_fd\_cur limit, the application, and possibly the operating system, will crash, often times with no error logging other than a core file.

Moreover, there is a relation between the TCP/IP stack and rlim\_fd\_cur and rlim\_fd\_max, because these protocols use sockets to make connections between the kernel and external machines. As mentioned previously in § 4.2.4.1.4, the maximum number of established connections (backlog queue) is 1024. Having a limit of 1024 for the file descriptor, guarantee that the server doesn't run out of descriptor for the accepted connections.

Likewise, SUN recommends for busy systems (e.g. iPlanet servers) to set the soft limit to 1024 and the hard limit to 2048.

To test that the file descriptor are correctly set:

Commands to test(objective)	Expected results
# ulimit -Hn	2048
# more /etc/system   grep rlim_fd_max	set rlim_fd_max=2048
# ulimit -Sn	1024
# more /etc/system   grep rlim_fd_cur	set rlim_fd_cur=1024
Reboot the system and redo the above test	Same results shall be observed

Note: a cross check between the value set in /etc/system and the value in the current kernel memory is needed, because the kernel reads the values in /etc/system only at boot time.

ulimit command sets or reads the limits of the available system resources. When ulimit is used with the -a switch it displays all the resources and their values. To specifically verify the file descriptor

limit the `-n` option is used. By default, the soft limit is returned. To have the hard limit, a capital H must be appended with the `-n` switch (`-Hn`).

#### 4.2.4.4 Core file

On a production systems core files should not be needed since debugging should take place only on a test system. On the other hand these files could be used by attacker to make a DoS by generating big core files and consuming the disk space. Furthermore, valuable information is sometimes stored on core files that may be helpful to attackers. To avoid the generation of core files, we can set the size of core file to 0. This is done by setting the `coredumpsize` system variable to 0.

Commands to test(objective)	Expected results
# more /etc/system   grep coredumpsize=	set sys:coredumpsize = 0
# ulimit -c	0
reboot the system and redo the test	Same results shall be observed

Note: The value has still to be effective after a reboot.

#### 4.2.5 File system

##### 4.2.5.1 Partitions and mounted file systems

The partitions can be mounted with different options. The use of these options for dedicated partitions can enhance the security level of the Solaris system.

The `nosuid` option disable SUID programs. In accordance with Req9 requirement, this option shall be set for the partition where no SUID files are expected. The `/home` partition is where users have their home directories. Users shouldn't be allowed to have or install SUID programs in their home directories. Therefore, the `/home` partition should be mounted with the `nosuid` option. The `/var` partition holds the log files. These files generally accessed for read and write, and there is normally no executable file stored there. Therefore we should also mount the `/var` partition with the `nosuid` option set.

The `noatime` option allows mounting file systems without updating inodes at each access to any file. This will significantly speed up services like web caches or news servers, which do a lot of I/O with small files. This option shall be set to `/var` because it contains the all the log files that are accessed frequently. If a special partition is created for the HTTP file of the web server, this partition can also be set with `noatime` option.

The `logging` option keeps a transaction log within the mounted partition. The advantage is an almost instantaneous file system check. The disadvantage is the additional time spent writing the transaction log. The `/usr` and `/home` partition should have this option set

The `size=` shall be used on `/tmp` partition. The value should be set approximatively to 30% of the swap space. This setting can prevent from slowing down the system when the `/tmp` partition is filling up, thus preventing a potential DoS.

The ro option is the read-only option. Mounting file systems read-only provides only a limited protection against Trojans/attackers (if they get root, they can remount read-write). However, it may save time fsck'ing when booting, can improve performance (access times don't need to be updated) and can prevent the sysadmin from making mistakes or help him detecting mistakes (accidentally deleting files etc.). The /usr partition can be set to ro, but in this case it's better to have a separate partition for /usr/local.

Commands to test(objective)	Expected results
# mount -p	the /var partition has the nosuid and noatime options set the /home partition has the nosuid and logging options set the /tmp - partition has the size set to ~30% of swap space the /usr - partition has the logging option set

mount is the command used to mount or unmount file system and remote resources. When invoked with the -p option it prints the list of mounted file systems in the /etc/vfstab format.

#### 4.2.5.2 Basic permissions

The sticky bit shall be set on /tmp and /var/tmp to prevent a user from deleting a file created by another.

Commands to test(objective)	Expected results
# ls -la /   grep tmp	drwxrwxrwt 11 sys sys 2590 Oct 10 10:23 tmp
# ls -la /var   grep tmp	drwxrwxrwt 11 sys sys 3400 Oct 10 10:25 tmp
create a file under an account with permission set to 777, then su to another account and try to delete the file	rm: operation not permitted

#### 4.2.5.3 World-write files

In accordance with Req9 requirement, files should not be world-write unless it's vital for the system. At least, the minimum following rules shall be followed:

- No world-write directory that are in the root search path.
- Sticky bit in world-write directory
- No world-write files in users' home directory.

- sysadmin have to know if world -write access on files are necessary

Following the A reasonable amount of files should be world writable. If too much files are world writable, then I should take closer look to see if an installation has not been made too permissively, and if it's possible to reduce the number of these files.

Commands to test(objective)	Expected results
# find / -perm -o+w   wc -l	Gives an idea of the number of world -write files and dir
Look in home dir if world -write files exist	No files should be found
Sticky bit set on standard world write dir	
# find / -perm -o+w	list of world-write files and directories should have been minimized

Test (subjective)	Expected results
Ask sysadmin for world -write files found	They have to know whether they need it or not.

#### 4.2.5.4 SUID and SGID files

In accordance with Req9 requirement, only the necessary files shall have the SUID bit set. We know that some past attacks have been taking advantages of SUID programs.

During execution of an SUID program, the user ID is set to the user ID of the program owner for the duration of the program. When the owner is root, the user has the superuser privileges when the program is running. This is made to grant normal users to execute commands that require root privilege but without giving them the access to the root account.

Removing, the SUID bit from files owned by root, means that the normal user won't be able to execute the command. Therefore, I should ensure that only binaries that are needed for normal users are left with SUID bit.

For the type of server I'm auditing, we have identify only the following files that can be left with the SUID bit set: utmp\_update, pt\_chmod, login, ping, passwd, su and ssh.

Note that is used to update the /var/adm/utmpx file and is needed to update information on logged user (viewed with the last command) and pt\_chmod is needed when a user is logging remotely (e.g. via ssh). The others are well known commands or programs.

Commands to test(objective)	Expected results
# find / -type f \( -perm -u+s -o -g+s \) -ls	/usr/lib/utmp_update /usr/lib/pt_chmod /usr/bin/login /usr/sbin/ping

---

```
/usr/bin/passwd
/usr/bin/su
/usr/local/bin/ssh
```

---

#### 4.2.6 Account and password policy

In accordance with Req6 requirements a strong account and password policy shall be in place. To comply to such policy, some parameters have to be tested on the system and some procedures shall be verified.

The following parameters shall be checked on the Solaris operating system:

- password length of at least 8 characters
- choice of strong passwords (no n-guessable, mix of numbers, capitals, and special characters)
- password expiration of 8 weeks (with a warning one week in advance)
- no duplicate accounts
- no accounts with no password
- encrypted password shall not be readable: force the use of /etc/shadow file
- minimum set of accounts
- non-users account shall be locked and have a default invalid shell.

Explanation	Commands to test(objective)	Expected results
minimum length of 8 characters	# egrep "PASSLENGTH" /etc/default/passwd	PASSLENGTH=8
	Try to change a password to a length smaller than 8 with passwd command	password can not be changed
password expiration of 8 weeks	# egrep "MAXWEEKS" /etc/default/passwd	MAXWEEKS=8
	# egrep "WARNWEEKS" /etc/default/passwd	WARNWEEKS=1
no duplicate accounts	# logins -d -x -m	no account shall be returned
no accounts with null password	# logins -p -x	no account shall be returned
Solaris' shadow password file shall be enforced	# more /etc/passwd	all accounts shall have an "x" in the password field in /etc/passwd
minimum set of accounts	# more /etc/passwd	Only root, daemon, adm, bin, sys, lp, uucp, nobody, noaccess, and one personal account for sysadmin shall figure.



non-users account shall be locked and have a default invalid shell	# more /etc/passwd	The last argument of the accounts: daemon, adm, bin, sys, lp, uucp, nobody, noaccess, shall have a shell like /sbin/noshell or /sbin/false.
	# passwd -sa	LK shall be displayed in front of the accounts: daemon, adm, bin, sys, lp, uucp, nobody, noaccess.

Then, I have to ensure that the following procedures are consistent and contribute to a strong password and account policy:

Test (Subjective)	Expected results
Ask the sysadmin and users how they choose their password (Is there a method?)	good practice, mix of numbers, characters, special characters, etc.
Can the sysadmin remember efficiently the root password (how many passwords do they have to remember)	copy of password shall be in secure places
What is the procedure when a person having a user account is leaving the company	See text below
What is the procedure when a person having access to the root account is leaving the company	See text below
What is the procedure to add a new user account	See text below
What is the procedure to give access to someone to the root account	See text below

The above mentioned procedures shall exist and must be available through a document. It's important to ask sysadmin if they are able to remember the root password without compromising the security. Often, sysadmins have several machines to manage in parallel. For each, at least one password must be remembered. Adding on top of this, that passwords are changing every 2 months, it's possible that the password be forgotten. To address this problem, sometimes, sysadmin write down somewhere their passwords, and possibly exposing them to unauthorized persons. It's thus important to know, that if there's a copy of passwords, it's stored in a secure place (e.g. in a safe), where only authorized persons have physical access (e.g. the security officer).

Another problem that is often overlooked is when a user is leaving the company. Immediate measures have to be taken to clean the user password and/or to change the root password. It's generally the task of the security officer to ensure that this procedure is enforced.

Likewise, a procedure must exist to allow access to someone to the machine. Generally, the security officer have to maintain an up-to-date list with the persons' name, the account name, and their role regarding the machine. It's also his task to grant sysadmin to create a new user account.

#### 4.2.7 Console login

I have to ensure that root login is only permitted via the console. In addition, login with an empty password shall be prohibited, and the syslog facility shall be invoked to log security events related to login accesses. These measure have to be set in the /etc/default/login file. The parameters and the values to check are the following:

Commands to test(objective)	expected results
# more /etc/default/login   grep CONSOLE	CONSOLE=/dev/console
# more /etc/default/login   grep PASSREQ	PASSREQ=YES
# more /etc/default/login   grep SYSLOG	SYSLOG=YES

In addition the following test shall be made:

Test(objective)	expected results
Login with an empty password	login shall be refused
Check the corresponding log file	the above test shall be logged

#### 4.2.8 Umask and Cmask

The default UMASK for users can be set in /etc/default/login file. The default user umask is used to set the initial permission when a file is created by a user. The umask value represents the complement of the permissions, i.e. a umask of 022 will set the permission to 644 to any created file. Following the least privilege principal (Req9) it is better not to have a too permissive umask. There's no reason for example to let read access to the others by default. A good policy is too have a umask of 027 or 077

Commands to test(objective)	Expected results
# more /etc/default/login   grep UMASK	UMASK=027 (or even better 077)
# umask	027 (or 077)
Login as a user and type:	
# touch testperm	Permission of file testperm shall be
# ls -la testperm	640 or 600

The `/etc/default/login` file is parsed when login occurs via the console or with telnet (and “r” services). However, this file is not taken into account when login is done via SSH program, except if the option `UseLogin` is set to “yes” in `/etc/ssh/sshd_config` which is not recommended (see § 4.2.12 for more detail on this)

When a user is logging in via SSH, the interactive shell specified in the `/etc/passwd` is invoked. Depending on the shell used, different configuration files are executed by the shell. It’s in these files that the `umask` has to set.

If bash shell is used, the following files will be executed (in the right order): `/etc/profile`, `$HOME/.bash_profile`, `$HOME/.bash_login`, `$HOME/.profile`, `$HOME/.bashrc`

If sh shell is used, the following files will be executed (in the right order): `/etc/profile`, `$HOME/.profile`

If csh is used, the following files will be executed (in the right order): `/etc/.login`, `$HOME/.cshrc`, `$HOME/.login`

If ksh is used, the following files will be executed (in the right order): `/etc/profile`, `$HOME/.profile`

Note that for non-root users, instead (or in addition) of invoking `$HOME/.cshrc` and `$HOME/.profile`, sometimes `$HOME/.local.cshrc` and `$HOME/.local.profile` are called.

I have to check that for every used shell, the `umask` is appropriately set in the corresponding files by invoking `umask 027` (or `077`). Particular care should be taken with the root shell and `umask`.

Test(objective)	Expected results
Login as root using SSH and type <code>umask</code>	027 (or 077)
Look what are the shell used by the users in <code>/etc/passwd</code> . Check in the corresponding files that <code>umask</code> is set to 027 or 077	<code>umask 027</code> (or 077)

In addition to the above test, it’s important to check that if a new user account is created with a “not-yet-used” shell, then the corresponding file are setting a `umask` of 027 or 077

Test (Subjective)	Expected results
Ask sysadmin if they ensure a <code>umask</code> of 027 or 077 when new user account is created	sysadmin shall know what files are required to set appropriately the <code>umask</code> for the invoked shell

There is another `umask` variable I have to verify. This the `umask` used by the `init` process at boot time. This is set by the `CUMASK` variable stored in the file `/etc/default/init`. Note that every child process created by `init` inherits the `CUMASK` value. To ensure that these processes don’t create world -write files, the `CUMASK` value shall be at least set to 022.

Test(objective)	Expected results
# grep CMASK /etc/default/init	022

#### 4.2.9 Path variable

The PATH environment variable shall not contain the “.” because known attacks have been taking advantage of this setting.

Commands to test(objective)	Expected results
# echo \$PATH	The “.” shall not be included

#### 4.2.10 Cron config files

In general the utilisation of cron job should be limited to normal users and system accounts shall not be able to schedule cron job. In accordance to Req9 requirement, all accounts except root shall prevented from using cron utility.

The configuration file /etc/cron.d/cron.deny is used to list all accounts that have to be denied the access to execute cron jobs.

Commands to test(objective)	Expected results
# more /etc/cron.d/cron.deny	daemon bin smtp nuucp listen nobody noaccess lp sys adm uucp
Compare the listed accounts in /etc/cron.d/cron.deny with the one in /etc/passwd.	except the root account the lists shall be the same
log on as user account and type: #crontab -l	crontab: you are not authorized to use cron. Sorry.

In the same way the use of the at command shall be reserved to the root account only. This can be set in /etc/cron.d/at.deny.

Commands to test(objective)	Expected results
# more /etc/cron.d/at.deny	daemon bin smtp nuucp listen nobody noaccess lp sys adm uucp
Compare the listed accounts in /etc/cron.d/cron.deny with the one in /etc/passwd.	except the root account the lists shall be the same

Note: even if the at command is not started (see the recommended list of running processes), for sake of consistency, we have to list all account names in the corresponding configuration file.

#### 4.2.11 TCPwrapper

TCPwrapper software should be installed. As said in previous paragraphs, only HTTP, HTTPS and SSH are expected to run as network services. HTTP(S) can not be protected by TCPwrapper. However, SSH can and should.

For our exposed server the /etc/hosts.allow and /etc/hosts.deny files shall be configured to allow only ssh from remote management hosts.

Commands to test(objective)	Expected results
# which tcpd	/usr/local/bin/tcpd
# more /etc/host.deny	ALL:ALL:DENY
# more /etc/hosts.allow,	sshd: <IP# of management host>: ALLOW
Try to connect via ssh from a machine that is not specified in host.allow	The connection shall be refused.

The /etc/host.deny file shall only have the line mentioned above (ALL:ALL:DENY). This is the default rules that deny every connection with network services wrapped by TCPwrapper.

In /etc/hosts.allow file, rule are set for management hosts. Whenever possible, it's better to allow specific hosts rather than a whole subnet.

Even if no services are started through inetd demon, most common services contained in /etc/inet.conf should be wrapped by TCP wrapper security software.

Sometimes, sysadmins are starting inetd services for a temporary period, for troubleshooting. They uncomment the corresponding line in /etc/inetd.conf, and start the inetd demon. If these services were previously wrapped, it can guarantee that the service is only opened for the machines specified in /etc/hosts.allow. It also prevent from accidentally opening a service when editing the /etc/inetd.conf.

To test if a service is wrapped by TCP wrapper, the tcpd program must be invoked at the corresponding line. E.g. for FTP service wrapped by TCP wrapper we should have the following line:

```
ftp      stream tcp nowait      root    /usr/sbin/tcpd      in.ftpd
```

While for a non-wrapped FTP service, the ftp daemon is called directly:

```
ftp      stream tcp nowait      root    /usr/sbin/in.ftpd    in.ftpd
```

At least the following services should wrapped by TCP wrapper: Telnet, FTP, NNTP, r -services, pop, uucp, TFTP, Finger, Systat, Netstat.

Commands to test(objective)	Expected results
# more /etc/inetd.conf	tcpd shall be invoked for the services mentioned above

Finally, with respect to the least privilege principle (Req9 requirement), the 2 configuration files shall have their permissions set to 400 or 600.

Commands to test(objective)	Expected results
# ls -la /etc/hosts.*	-r----- 1 root root 1584 Jul 12 17:53 etc/hosts.allow
	-r----- 1 root root 1584 Jul 12 17:53 etc/hosts.deny

#### 4.2.12 SSH security

First, the version of SSH used shall be up -to-date. This can be verified through the vulnerability scanner results and by directly using the ssh command. The -V switch returns the version of the ssh software installed.

Commands to test(objective)	Expected results
# ssh -V	latest version of OpenSSH (or SSH)
Nessus results	latest version of OpenSSH (or SSH)

To ensure that SSH security software is used in a secure way, appropriate settings have to be implemented in the corresponding configuration file (/etc/sshd\_config).

SSH can use two protocol version (1 and 2). Whenever possible, only the protocol 2 must be used because it's the most secure one (some weaknesses have been reported with protocol 1). If only protocol version is enabled then all parameters related to version 1 shall be disabled. This is the case of "RhostsRSAAuthentication" and "RSAAuthentication" parameters. They both have to be set to "no".

It's also important to log events related to SSH via the syslog tool. To collect enough information the log level must be at least set to info level (check § 4.2.13 for more details on this). The most appropriate facility for logging is AUTH.

The server key length must be at least 768 bit, to guarantee a minimum cryptographic strength (the shorter the key is, the easier the brute force attack is).

Direct root login via SSH can be permitted if no more than 2 sysadmin know the root password. Otherwise, login must be made from a non-root account and su shall be used. This is to be able to trace the account used in case of problems.

We don't want to allow any kind of trusts between the server and other machines. To disallow this feature a number of parameters shall be set to "no" in /etc/ssh/sshd\_config.

First, the "IgnoreRhosts" has to be set to "no" to disallow the use of possible trusts specified in the .rhosts or .shosts files. However, the trusts specified in /etc/hosts.equiv and /etc/shosts.equiv are still effective.

The RhostsAuthentication shall be set to "no". It disables simple trusts specified by /etc/hosts.equiv or /etc/shosts.equiv files.

The RhostsRSAAuthentication uses the trusts specified in .rhosts or .shosts if "IgnoreRhosts" is set to yes, or the trusts specified in etc/hosts.equiv and /etc/shosts.equiv file. In addition it performs a machine authentication based on RSA, using private keys stored in the /etc/ssh/ssh\_host\_key. This key is normally stored in cleartext. This method is used by the protocol version 1, and therefore should be disabled, i.e. set to "no".

The HostbasedAuthentication is similar to RhostsRSAAuthentication, but uses protocol version 2. It should also be disabled as we don't allow trusts of any kind with the server. Therefore it shall be set to "no". Note that this method is the less risky method using trusts, and in some circumstances, it can be allowed. For example, if remote backups are used in conjunction with SSH, they might use trusts and host public key authentication. However, if this option is enabled it shall be justified for a special purpose.

The RSAAuthentication is used for user authentication with SSH. This method is used for protocol version 1 and therefore shall be set to "no".

For protocol version 2 there is two possible user authentication for login with SSH. The password authentication that uses the UNIX password, and the Public key authentication. While both can be enabled, it should be better to allow only one policy (it's easier to control one policy than two!)

The UseLogin option can tell SSH to use the UNIX login program. This should be avoided, because if login program has a vulnerability then SSH can inherit this vulnerability. Therefore it shall be set to “no”.

Here are the recommended minimum parameters to check for this file:

Expected line in sshd_config (objective)		Comments
Protocol 2		It's better to allow only SSH protocol version 2 because it's more secure. Default are set to Protocol 2,1 which means that in first priority protocol version 2 is used, and then Protocol 1 is tried. If SSH clients can handle version 2 of SSH protocol then the value shall be set to 2.
SyslogFacility AUTH		Logs the SSH events to /var/log/authlog
LogLevel INFO		The login level shall be at least set to info.
ServerKeyBits	768	The key length shall be at least 768 bit long
PermitRootLogin no		If only one or two persons have access to the root account to manage the web server, direct root login can be permitted. However, it's better to not allow direct root login, but to log on via a normal user account and to su to root after. In case of problems, we can better track the responsible persons.
PermitEmptyPasswords no		no comment!
StrictModes yes		Strict checks on ownership and access rights for all user files shall be enabled. This avoids attacks based on insufficient protection of such files.
IgnoreRhosts yes		Ignore the trusts between the machines using .rhosts and .shosts files.
RhostsAuthentication no		Ignore trusts using /etc/hosts.equiv and /etc/shosts.equiv
RhostsRSAAuthentication no		This is for the use of trusts between hosts with an RSA authentication. It is based on host keys only and is therefore not recommended. It is no longer supported in the version 2 protocol.
HostbasedAuthentication no		This is for the use of trusts between hosts using a Public key authentication. It's the same as “RhostsRSAAuthentication” but for protocol version 2
RSAAuthentication no		RSAAuthentication authentication is no longer supported in version 2 ssh protocol (It has been replaced by DSA authentication), so it shall not be used.
PubkeyAuthentication no/yes		It's the same “RSAAuthentication” as but for protocol version 2



PasswordAuthentication yes/no	Password authentication shall be enabled
UseLogin no	Specify whether login program is used for interactive SSH login

The permission of the sshd\_config files shall be restricted to 400 or 600 with regards to Req9 requirement.

Commands to test(objective)	Expected results
# ls -la /etc/sshd_config	-r----- 1 root root 1584 Jul 12 17:53 /etc/sshd_config

In addition, to ensure that neither empty password nor root login are permitted the following test shall be made:

Test(objective)	Expected results
Login as root	login shall be refused
Login with an empty password	login shall be refused

#### 4.2.13 Logging

A complete and comprehensive logging policy should be defined and then implemented in order to ensure that enough information is collected, and that sysadmins are informed at the right time when problems occur. This policy has to be a formal document written by security officer (possibly with the help of sysadmin), and approved by the management.

Test (objective)	Expected results
Ask sysadmin, security officer and managers if a written policy exist	A written document

To estimate if a log policy is comprehensive and consistent might be very subjective. However, some few points are essentials. First, the policy should state when the logs have to be viewed: it shall be a task of a dedicated person (generally the sysadmin). Then priority can be set to the different type of log events. For example, the log event related to failed access can be viewed every day, while others can be viewed every week. In addition, for serious presumption of break-in alarm should be triggered. Finally, a procedure shall be triggered in case of emergency with the right persons informed.

Test (subjective)
Check that the policy is comprehensive
Check that logs are viewed regularly

---

Check that right priority are set to view the logs

---

Check that right priority are set for alarms

---

In case of alarms, the right persons must be informed, and if serious breaking has occurs an emergency procedure shall be triggered

---

In addition, I have to ensure that this policy is really enforced, and that right tools are used.

Test (subjective)	Expected results
Check that the policy is enforced	The document must be followed
Check that the right tools are used	At least syslog and a tool to analyze log shall be used

On Solaris system, the syslog shall be used to enforce a part of the log policy described above. This tool permits to log all types of event in a centralized way and therefore shall be used. Syslog works with facilities such as kern, mail, auth, daemon, syslog, lpr, mark, news, uucp, cron. In the configuration file (/etc/syslog.conf), we can specify to log events in a dedicated file (generally stored under /var/log/\*) relative to each facility. It's also possible to specify from which priority level we want to log the events. We have 9 levels of priority (from highest to lowest): emerg, alert, crit, err, warning, notice, info, debug. To have necessary error information reported, all facilities shall be logged from the info priority, to the corresponding files. In addition, emergency errors can be sent to all terminal.

Therefore the syslog.conf file shall be checked for

Test(objective)	Expected results
All facilities shall be logged on separate file under /var/log/	See the Figure 1
Check that the minimum priority level is info.	See the Figure 1
Generate events to trigger syslog to store the event (e.g. ssh login with wrong password)	event must be stored
Check permission for log file (/var/log/*log)	permission shall be set to 600

# All emergency level message are sent to all terminal

\*.emerg \*

# All facilities logged from info level to store sufficient information

```
kern.info      /var/log/kernlog
user.info      /var/log/userlog
mail.info      /var/log/maillog
daemon.info    /var/log/daemonlog
auth.info      /var/log/authlog
lpr.info       /var/log/lprlog
news,uucp.info /var/log/newslog
cron.info      /var/log/cronlog
```

```
# Put all alerts (& higher) into a separate log:
*.err          /var/log/alertlog
```

Figure 1: how the syslog.conf file should look like

Note, that in accordance with Req9 requirement (least privilege), the log files have to be set with a permission of 600.

As mentioned previously, a tool to analyze the generated log files shall be used. Good tools exist such as Logcheck (available at [www.psionic.com/abacus/logcheck](http://www.psionic.com/abacus/logcheck)), that helps to analyze the log files and send alert via e-mail, for error that the sysadmin has defined as serious security violation. I have to ensure that such tool enforces the security policy mentioned above about the prioritization of alarm and viewed events.

Test (Subjective)	Expected results
Check that the policy for alarm prioritization is enforced	Policy and implementation must be consistent

Test(objective)	Expected results
Generate an event that is considered as a serious security violation	Check if an e-mail is sent to the right mailbox

Finally, if administrator have several machines to administer, it might be the case that the logs from all machines are centralized on loghost (central server dedicated to collect log information from other machines). In this case the principle, for log policy is the same, but the log file differ slightly (c.f. Appendix B -§10- for an example).

#### 4.2.14 Integrity check software

To ensure the integrity of important files, and to detect unauthorized change, integrity software such as AIDE or tripwire must be installed.

Test(objective)	Expected results
Ask sysadmin if an integrity check software is installed	AIDE or tripwire must be installed

The minimum system files to be checked for integrity are: /kernel, /usr/bin/\*, /usr/sbin/\*, /usr/local/bin/\*, /etc/passwd, /etc/shadows, /etc/sshd\_conf, /etc/aide.conf, /etc/system, /etc/hosts.allow, /etc/hosts.deny

At least the modification of the files and the permission modification shall be checked. The database of the integrity check software must contain a cryptographic hash (e.g. MD5) of the above mentioned files. At least the modification of the files and the permission modification shall be included in the hash calculation. An example of such setting with AIDE is shown below:

Commands to test (AIDE)	Expected results
# more /etc/aide.conf	Rule=p+u+g+m+md5
	/usr/bin Rule
	/usr/sbin Rule
	/usr/local/bin Rule
	/kernel Rule
	/etc/passwd\$ Rule
	/etc/shadows\$ Rule
	/etc/sshd_conf\$ Rule
	/etc/aide.conf\$ Rule
	/etc/system\$ Rule
	/etc/hosts.allow\$ Rule
	/etc/hosts.deny\$ Rule

Similar rules are expected to be specified if Tripwire is installed instead.

On a regular basis, the database must be recalculated and compared to the old one. If a difference is found, then an alarm should be triggered. As an example a shell script using AIDE is given in Appendix A (§9). This script must be scheduled with the cron program. This script calculates the hash of the specified files and compares the results with the previous database. Any difference is stored in a log file.

In addition a copy of the database should be stored either on a write-once device or with a daily backup.

I should ensure that similar policy is implemented

Test (objective)	Expected results
Verify the integrity check software configuration	Implementation has to be similar to the above mentioned policy.

An example of such script is given in Appendix A (§ 9).

#### 4.2.15 Policy for security patches

A clear policy shall be in place, to decide whether a weakness is worth patching. For our web It shall include a strategy to set priority on patches to apply based on the following statements:

- If the weakness concerns a remotely exploitable weakness in an active network daemon, open to internet, the patch shall be tested and installed immediately. For our web server, it means any security patches related to the HTTP daemon, the SSL daemon, and any third party application used by the web application.
- If a weakness is remotely exploitable from the Intranet, apply it as soon as possible. For our web server, this concern the sshd daemon.
- If a weakness concerns a local exploit on a running daemon install it rapidly. For our web server this concern the list of daemon listed in paragraph 4.2.3.3
- for other weaknesses it may be enough to install the patch together with a bundle at regular intervals (e.g. every 2-3 months).
- Use a patch tool to help find relevant patches and install them. For patches related to Solaris some good tools such as GetApplyPatch, CheckPatches or PatchDiag can be used. More detail about these tools is given in [11], [12], and [13]
- Sysadmin shall subscribe to the vendor's mailing list for new patch announcement. In addition, being on a mailing list of an organization which produces regular summaries of weakness/patches and security news (e.g. for example SecurityFocus, SANS, etc) is recommended.

Test (Subjective)	Expected results
Ask sysadmin for their patching policy	A clear policy addressing the above mentioned points shall be in place.
Ensure that the policy is followed by sysadmin	Procedure described in policy must be followed
Ensure that sysadmin is on right mailing lists	He should subscribe to mailing for Solaris and third party applications if any (e.g. apache)
Ask sysadmin if they verify that secure state is maintained after patch installation	Check after reboot that no new services or processes are started.

In addition, to the stated policy, a kind of patch post-installation procedure should exist. This step is necessary to ensure that no new services, processes or daemons have been installed with the patches.

#### 4.2.16 Back-up and recovery policy

A written backup policy shall be clearly defined and shall address the following points:

- the frequency of incremental backups
- the frequency of full backups
- physical storage of backups (use of safe)
- how long the backups are kept
- who has physical access of backup
- recovery procedure shall be test
- If a special tool is used the tool shall enforce the backup policy.

<b>Test (Subjective)</b>	<b>Expected results</b>
Check that a backup and recovery policy is in place and is enforced	Policy shall address the above mentioned points.

© SANS Institute 2000 - 2002, Author retains full rights.

## PART TWO: Application of Audit Techniques to a Real World System

---

### 5 Audit results

#### 5.1 Environment description

The network topology is the following:

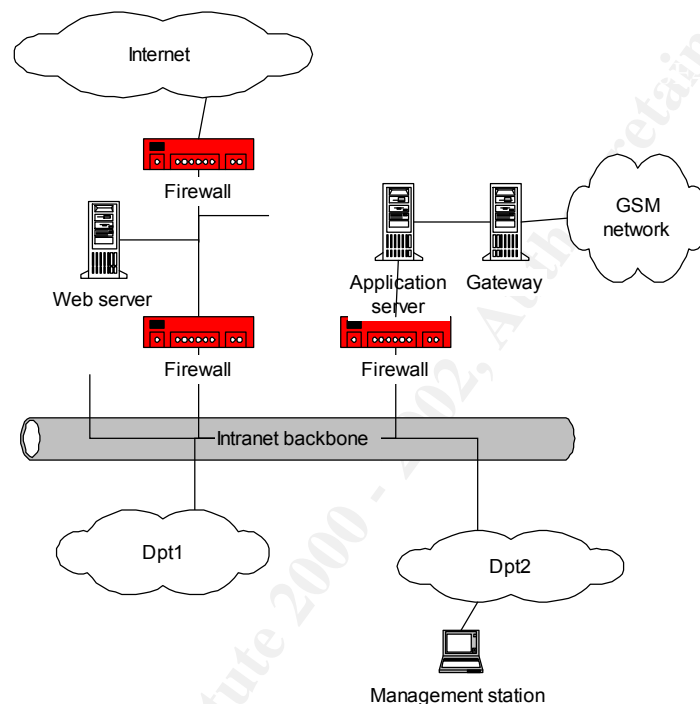


Figure 2: network environment

In the introduction chapter I mentioned that we have to make our service available to Internet users. This is a concrete example: our mobile department has deployed a service to his mobile customers to make “restricted zones of friends with chat rooms”. This service was available with the mobile phones using SMS (Short Message System), and is now ported to the IP world. Internet users will be able to use this service not only with their mobile phone but also from any PC connected to Internet.

This service is offered via a web interface. The users have to log on to the application with HTTP using their mobile phone number as username and a password. Then, the web server makes a query to the application server itself connected to the GSM network through a gateway (that makes the conversion between IP and SS7 protocols).

The web server is located in a screened subnet zone, i.e. is protected from Internet by a “permissive Firewall” and separated from the Intranet backbone by a more restrictive firewall. Note that the firewall rulebase inspection will not be part of this audit.

An important consideration is that our Intranet is considered as “moderately secure”. First, this backbone is shared by more than 12’000 users everyday, spread within more than 10 departments; second, we still have some connections with external networks (e.g. Internet) that are not secured and monitored in a centralized way by competent staff. Even if strong efforts have been made to suppress these “wild” connections, some are still present, especially in testing labs and environments.

## 5.2 Scope of the audit

In this audit I will evaluate the security of the operating system of the machine hosting the web server. The web application and the application that communicates with the application server is not part of this audit. Furthermore, in the project schedule, the applications will be installed on the machine if and only if the OS installation passes the security audit.

The OS that is audited here is Solaris 8 installed on a Sun SPARC station.

An important consideration is that the audit was made in mid -October 2001, and some of the results might be out-of date at the time of writing (like the version of some software).

## 5.3 Risk analysis

Because estimating the risks might be quite subjective, I asked two other colleagues of our security team, Stephane Grundschober (GIAC -GSNA certified) and Stephane Dagonnier, to participate to this task.

For each audit item or group of audit items identified in the first part of this paper, I asked them to fill two columns: one for the likelihood that a break -in succeed if the audited item doesn’t conform to our requirements; and the other for potential damages caused by such a break -in. For the likelihood I defined three levels, 1,2 and 3, 1 meaning low probability that a break -in succeeds, and 3 for high probability. I also defined 3 levels for the damages: 1 represents a non-significant loss, 2 represents an average loss (e.g. loss of image), and 3 represents critical loss (e.g. loss of the service, theft of customers’ data, etc.). Then I calculated the risks by multiplying the likelihood with the damages. The results I obtained from each of us is given in APPENDIX C (§ 11).

Finally, I calculate the average of the risks estimated by each of us. The maximum risk is 9 (=3x3), the minimum being 1(=1x1).

The final results are given below:

<i><b>Audit Item</b></i>	<i><b>Risk</b></i>
Installed packages	7.33
Strategy for Installed packages	2.33
OS patches	9
Inetd service	8
Minimum boot network services running	8



Sysadmin knowledge of running network services	4.67
Procedure to start a new network service	2
Minimum running processes	4
Knowlegde of running processes	3
Procedure to start a new deamon	2
ARP defenses	3
ICMP defenses	4.33
IP defenses	5
TCP defenses	4.67
Persistency after reboot	7.33
User stack protection	6
File descriptor	1.33
Core files	1.67
Partitions	3.67
/tmp sticky bit	1
World-write files and dir	3.33
SUID SGID file	6.33
Password length 8	4.67
Password expiration	6.33
no duplicate accounts	1.33
accounts with null password	9
Shadow file enforced	5.67
minimum set of accounts	2
Non-user account locked and non-valid shell	5.33
Method to choose strong passwd	6
Password storage	8
Procedure when user leaves cpny	7
procedure to add a new user and give authorization to access the machine	5.33
console login	4
Umask	5.33
CMASK	4
Path variable	5.33
cron.deny at.deny	1.33
TCP wrapper installed and configured	6.33
SSH version	9
SSH protocol	8
log of SSH events	3
SSH server key length	3.67
SSH Root login	4.33
SSH empty passwd	9
SSH No trusted hosts	6
RSA or passwd authentication	4.67
400 /etc/sshd_config	3.33
Log policy exist	7.33
log policy is comprehensive	6.33
log policy is enforced	9
log tools installed and configured	9
Integrity check software installed and well co nfigured	7.33
Patching policy exist	7.3
Patching policy enforced	9

Secure state after applying patches	6.67
Backup and recovery policy exist	4.33
Backup and recovery policy enforced	7

High risk values range from 7 to 9. Average risk values range from 4 to 6. Low risk value ranges from 1 to 3. In the above results I have written the highest risk in red.

These results will be used to prioritize the measures to apply. More details on this is given in §xxx.

## 5.4 Scan from Intranet (internal scan)

```
# nmap -O -p 1- xxx.xxx.xxx.xxx

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on www.vvv.dd (xxx.xxx.xxx.xxx):
(The 65529 ports scanned but not shown below are in state: closed)
Port      State      Service
22/tcp    open       ssh
80/tcp    open       http
443/tcp   open       https
7937/tcp  open       unknown
7938/tcp  open       unknown
32768/tcp open       unknown

TCP Sequence Prediction: Class=truly random
                        Difficulty=9999999 (Good luck!)
No 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:
TSeq(Class=TR)
T1 (Resp=Y%DF=Y%W=60DA%ACK=S++%Flags=AS%Ops=NNTNWM)
T1 (Resp=Y%DF=Y%W=60DA%ACK=S++%Flags=A%Ops=NNT)
T2 (Resp=N)
T3 (Resp=N)
T4 (Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T5 (Resp=Y%DF=Y%W=0%ACK=S++%Flags=AR%Ops=)
T6 (Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T7 (Resp=N)
PU (Resp=Y%DF=Y%TOS=0%IPLEN=70%RIPTL=148%RID=E%RIPCK=E%UCK=E%ULEN=134%DAT=E)
```

The port scan was done with Nmap version 2.53 on a Linux platform.

The result of this TCP scan shows what services are visible to Intranet. First, 3 standard ports are actually opened: TCP port 22 for SSH, TCP port 80 for HTTP and TCP port 443 for SSL. As stated in our checklist these service are “authorized to run”. However, 3 other TCP ports are opened: the port 7937, 7938 and 32768. I will have to investigate on these open ports, to see if they are really needed and they are harmless. This will be presented later in this paper when showing the results of the local audit.

Nmap rated the TCP sequence number prediction as almost impossible, which makes hijacking attacks very difficult.

Finally, nmap was not able to guess the type of OS.

```
# nmap -sU -p 1-
starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on www.vvv.dd (xxx.xxx.xxx.xxx):
(The 65533 ports scanned but not shown below are in state: closed)
Port      State      Service
123/udp    open       ntp
7938/udp   open       unknown

Nmap run completed -- 1 IP address (1 host up) scanned in 7377 seconds
```

For the UDP scan, nmap shows two ports open. The UDP port 123 for NTP service, the high port 7938.

If we refer to our checklist, NTP is not part of the accepted running network services. Furthermore, this service has had security vulnerabilities discovered several times. However, maintaining an accurate time and date is necessary, especially for the log analysis. I will show later in this paper how to replace this NTP network service by a simple cron job, which is a much more secure.

The UDP port 7938 is also open (note that I found the TCP port number opened). If I refer to the checklist this port is not authorized to be opened.

I should look further to what applications the non -standard ports correspond to in order to determine if they are needed and harmless. This will be covered in the next chapters dealing with the local audit.

## 5.5 Port scan of the web server from Internet

```
# nmap -p 1- xxx.xxx.xxx.xxx

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on www.vvv.dd (xxx.xxx.xxx.xxx):
(The 65529 ports scanned but not shown below are in state: closed)
Port      State      Service
22/tcp    filtered   ssh
80/tcp    open       http
443/tcp   open       https
7937/tcp   filtered   unknown
7938/tcp   filtered   unknown
32768/tcp filtered   unknown
Nmap run completed -- 1 IP address (1 host up) scanned in 352 seconds
```

The nmap TCP scan result from Internet shows the same open ports on the server. However, the firewall filters SSH, and the 3 high open ports. Even if the firewall rules are not part of this audit it's important to mention that for TCP ports only the strict minimum set of services have been opened on the Internet firewall, in accordance with Req1 requirements.

```
# nmap -sU -p 1-
starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on www.vvv.dd (xxx.xxx.xxx.xxx):
(The 65533 ports scanned but not shown below are in state: closed)
Port      State      Service
123/udp    open       ntp
7938/udp   filtered   unknown

Nmap run completed -- 1 IP address (1 host up) scanned in 8154 seconds
```

The UDP port scan from Internet shows the same ports opened on the server. The high port 7938 is filtered by the firewall. The surprise is that the nmap shows that the UDP port 123 corresponding to NTP service is opened to Internet. This means that the Firewall doesn't filter this port. Even if the audit of the firewall rules is out-of-scope, more investigation is necessary to understand why this port is opened. To our knowledge, we have two NTP reference servers that are internal machine, and time synchronization has to be made with these dedicated servers. I interviewed the sysadmin of the web server to know if he used Internet NTP servers for synchronization. He answered me that he uses our internal reference NTP servers. So I asked the Firewall administrator to immediately close this port. He was surprised and told me that as far as he knew he never opened this port. After checking his rulebase, this port was really not opened. I rescanned the NTP port from Internet, with snoop running on the web server and a sniffer placed on the external interface of the firewall. Then, I realized that no UDP packets were arriving on the server (the firewall was really blocking NTP port), but that the firewall were returning bogus UDP packets in response to our scan!

## 5.6 Vulnerability scan of the web server

The output of nessus is the following:

### Information found on port ssh (22/tcp)

Remote SSH version : ssh-1.99-openssh\_2.2.0p1

### Information found on port http (80/tcp)

The remote web server type is :

IBM\_HTTP\_Server/1.3.12.3 Apache/1.3.12 (Unix)

We recommend that you configure your web server to return bogus versions, so that it makes the cracker job more difficult

The scan was done using Nessus with latest available (26.10.2001) vulnerability plugins. Except the buffer overflow vulnerability that are sometimes dangerous (may cause system crash), all other existing vulnerabilities were tested. The results show that no security vulnerability was found on this server. It was only able to retrieve information about the web server type and version as well as the sshd daemon version and type.

Note that nessus shows an old version of SSH. At the time of the audit the last available version is OpenSSH\_2.9.9p. An upgrade to this version shall be made.

## 5.7 Results of local audit

### 5.7.1 Installed packages

Results	Verdict
# /usr/bin/pkginfo -i   wc -l 649	Fail

This results shows that 650 packages were installed on this machine! We can expect that no strategy has been adopted to limit the number of packages. This will be confirmed in the results below.

Results	Verdict
Package installation strategy: none	Fail

After interviewing, the sysadmin it appears that a default installation has been made, rather than a custom installation. No attention were paid to select only needed packages.

Results	Verdict
# /usr/bin/pkginfo -i The list is given in APPENDIX D (§ 12)	Fail

A lot of installed packages are not needed. For example all packages related a graphical interface or service are not needed and shall be removed. This includes all packages related X11, CDE, Elite3D, OpenWin, etc. Likewise, all packages related to audio device s shall be removed. The printer packages shall also be removed. Some applications packages like uucp, dhcp, ppp, power management, etc. shall also be removed. I have highlighted them in yellow, in the corresponding APPENDIX D (§ 12). They represents more 170 packages that can be immediately removed. It's important to mentioned that before removing them on the production server this should be first done on a test server. After discussion with the sysadmin, I realized that such a server already exist, and is configured exactly like the production server.

For the other packages, more scrutiny is needed. For the hardware drivers packages, we have to know exactly what hardware is present on the machine (e.g. the type of Ethernet cards), before removing packages. There are also the application packages. All packages related to IBMHTTPD server should be left as there are needed by the web server.

The measures can be applied in three steps:

- immediately remove packages highlighted in APPENDIX D (§ 12) on the test server. Reboot and observe the behavior during 2 days. If OK applied on the production server.

- Look with sysadmin what exactly hardware package are needed and which one can be removed.
- Look in IBM documentation and support, what packages are needed for their IBMHTTPD server.

### 5.7.2 OS installed patches

Results	Verdict
# ./PatchCheck See results in Appendix E (§ 13)	Fail

The system has not been patched with up-to-date patches. The results in Appendix E show that many security patches are missing. I highlighted in yellow some patches that are directly related to running processes (cron, xntpd), that are part of the kernel or that apply to TCP/IP drivers. In addition many recommended patches haven't been applied.

The sysadmin agrees to quickly test these patches on the test system and then to apply them on the production server.

Note that some patches that are listed are not relevant for our server because the related packages were not installed.

### 5.7.3 Network services

#### 5.7.3.1 inetd services

Results	Verdict
# ps -ef   grep inetd root 243 1 0 Oct 26 ? 0:00 /usr/sbin/inetd -s -t	Fail
# pgrep inetd 243	Fail

Inetd is running. According to our checklist it should be stopped. I have to look why it's running, i.e. what services are started by inetd.

By looking at non commented lines /etc/inetd.conf we can see which services are started by inetd daemon:

Results	Verdict
# egrep -v "^#" /etc/inetd.conf 300326/4 tli rpc/tcp wait root /platform/SUNW,Ultra-Enterprise-10000/lib/dr_daemon dr_daemon	Fail

The dr\_daemon is a Remote Procedure Call (RPC) program that provides the interface to the Sun Enterprise 10000 Dynamic Reconfiguration (DR) driver. After discussion with the sysadmin, it appears

that this service was not needed, and therefore will be stopped. The corresponding line in `inetd.conf` will be commented, and `inetd` demon will not be started from the `rc` scripts.

### 5.7.3.2 Boot services

Results	Verdict
<pre># lsof -i COMMAND  PID  USER  FD  TYPE  DEVICE  SIZE/OFF  NODE NAME inetd    243  root   11u  IPv4  0x300016aa1c8      0t0  TCP *:32768 (LISTEN) nsrexecd 328  root    5u  IPv4  0x300018ae1d0      0t0  UDP *:7938 (Idle) nsrexecd 328  root    6u  IPv4  0x300019bdd58      0t0  TCP *:7938 (LISTEN) nsrexecd 343  root    6u  IPv4  0x3000114c550      0t0  TCP *:7937 (LISTEN) sshd     334  root    6u  IPv4  0x300018ae6d0      0t0  TCP *:ssh (LISTEN) xntpd    345  root   19u  IPv4  0x300019faf98      0t0  UDP *:ntp (Idle) httpd    4632  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd    4632  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd    4932  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd    4932  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd    6120  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd    6120  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd    6873  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd    6873  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd    8736  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd    8736  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   15474  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   15474  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   21652  root   15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   21652  root   16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   21670  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   21670  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   23432  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   23432  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   23449  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   23449  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN) httpd   27294  nobody 15u  IPv4  0x300055aea40      0t0  TCP *:443 (LISTEN) httpd   27294  nobody 16u  IPv4  0x300055af6c0      0t0  TCP *:80 (LISTEN)</pre>	Pass: only xntp shall be stopped inetd has a l-ready been discussed

The service on the port 32768 started by `inetd`, is the service I found in the previous paragraph (`dr_daemon`), and that should be stopped.

Otherwise, the `lsof` output shows the 3 authorized services HTTP, HTTPS and SSH. However, we have two services that were not foreseen: `xntpd` and `nsrexecd`. As mentioned previously, `xntpd` shall be replaced by a cron job. It's safer to run it as a cron job than letting this service running because of it's bad security reputation. The following line should be added in the root crontab:

```
15 * * * * /usr/sbin/ntpdate -s <IP# ntpserver>
```

Note that here I specified to synchronize the time once per hour at quarter past.

The man pages says: “ *The ntpdate utility sets the local date and time. To determine the correct time, it polls the Network Time Protocol (NTP) servers on the hosts given as arguments. This utility must be run as root on the local host. It obtains a number of samples from each of the servers and applies the standard NTP clock filter and selection algorithms to select the best of these.* ” In addition, the `-s` option permit to log the related events through the `syslog` facility.

Running a cron job that makes a UDP request, it is much more safe than letting run a network daemon because if vulnerability is discovered in NTP it can be remotely exploited.

The UDP and TCP ports 7938 are related to a backup software from Legato system (<http://portall.legato.com/>). nsrexecd is the daemon that authenticates the Backup server's remote execution request and executes the save and savefs commands on the client. After discussion with the sysadmin, it appears that it is their standard way of doing backup. All backups of this department are centralized to a backup server located in secure room, in the basement. As backups are vital, we decided to let this service running.

The output of netstat -a gave the same results. By looking under the UDP and TCP tables I was able to confirm the above results. The output of this command is not shown here as it doesn't provide additional information.

Note: I give a passing grade to this test because only xntp will be stopped (inetd was part of the previous test where I already gave a fail for it).

Results	Verdict
After reboot the same results were observed with lsof -i and netstat -a	Pass

After a reboot of the machine I did not observe additional services that started. The output of lsof -i and netstat -a showed the same results.

Results	Verdict
Sysadmin knowledge network services	Pass
Procedure to start a new service	Fail

After interviewing the sysadmin, I realize that he has in-depth knowledge of the network services.

However, no procedure of control exists for the start of new network services. After discussion with the sysadmin and his manager, it has been agreed that a small procedure will be addressed.

### 5.7.3.3 processes

Results	Verdict
<pre># ps -ef UID      PID     PPID  C   STIME TTY          TIME CMD root      0        0  0   Oct 21 ?        0:17 sched root      1        0  0   Oct 21 ?        4:57 /etc/init - root      2        0  0   Oct 21 ?        0:00 pageout root      3        0  1   Oct 21 ?        555:49 fsflush root    367        1  0   Oct 21 ?        0:00 /usr/lib/saf/sac -t 300 root    278        1  0   Oct 21 ?        0:01 /usr/lib/utmpd root     67        1  0   Oct 26 ?        0:00 /usr/lib/picl/picld root     56        1  0   Oct 21 ?        0:00 /usr/lib/sysevent/syseventd root     58        1  0   Oct 21 ?        0:00 /usr/lib/sysevent/syseventconfd</pre>	Pass-except picld process is not necessary and should be stopped



root	243	1	0	Oct 21 ?	0:00	/usr/sbin/inetd -s -t	xntp and inetd have already been dis- cussed.
root	254	1	0	Oct 21 ?	0:04	/usr/sbin/cron	
root	343	328	0	Oct 21 ?	0:02	/usr/sbin/nsr/nsrexecd -s xxxxx	
root	345	1	0	Oct 21 ?	0:01	/usr/lib/inet/xntpd	note: the “xxxx” replace server name for security reason
root	269	1	0	Oct 21 ?	0:23	/usr/sbin/nscd	
root	288	1	0	Oct 21 ?	0:02	/usr/sbin/vold	
root	253	1	0	Oct 21 ?	0:03	/usr/sbin/syslogd -t	
root	328	1	0	Oct 21 ?	0:02	/usr/sbin/nsr/nsrexecd -s xxxxx	
root	307	1	0	Oct 21 ?	0:00	/usr/lib/sendmail -q15m	
root	334	1	0	Oct 21 ?	0:21	/usr/local/sbin/sshd	
root	14439	1	0	Oct 10 console	0:00	/usr/lib/saf/ttymon -g -h -p xxxxx	
console login: -T sun -d /dev/console -							
nobody	8736	21652	0	Oct 21 ?	0:49	/opt/IBMHTTPD/bin/httpd	
nobody	4632	21652	0	14:18:07 ?	0:02	/opt/IBMHTTPD/bin/httpd	
root	398	367	0	Oct 21 ?	0:00	/usr/lib/saf/ttymon	
nobody	23432	21652	0	Oct 21 ?	0:12	/opt/IBMHTTPD/bin/httpd	
root	2552	334	0	Oct 21 ?	0:17	/usr/local/sbin/sshd	
nobody	21670	21652	0	Oct 21 ?	1:43	/opt/IBMHTTPD/bin/httpd	
nobody	6873	21652	0	15:54:18 ?	0:01	/opt/IBMHTTPD/bin/httpd	
root	21652	1	0	Oct 21 ?	0:02	/opt/IBMHTTPD/bin/httpd	
nobody	27294	21652	0	09:02:18 ?	0:05	/opt/IBMHTTPD/bin/httpd	
nobody	4932	21652	0	Oct 21 ?	0:28	/opt/IBMHTTPD/bin/httpd	
nobody	15474	21652	0	Oct 21 ?	0:19	/opt/IBMHTTPD/bin/httpd	
root	9484	334	0	17:41:00 ?	0:00	/usr/local/sbin/sshd	
root	9355	334	0	17:37:48 ?	0:00	/usr/local/sbin/sshd	
nobody	6120	21652	0	Oct 21 ?	1:17	/opt/IBMHTTPD/bin/httpd	
nobody	23449	21652	0	Oct 21 ?	0:33	/opt/IBMHTTPD/bin/httpd	

The xntp, inetd daemon have already been discussed. A part from these all other daemon are necessary except one: the picld daemon. If we refer to the man pages it says: "*Platform Information and Control Library (PICL) provides a mechanism to publish platform -specific information for clients to access in a platform-independent way. picld maintains and controls access to the PICL information from clients and plug-in modules.*"

This is obviously not a necessary process and therefore in agreement with the sysadmin this it will be removed from the start-up rc script by doing the following:

```
# mv /etc/rcS.d/S95picld /etc/rcS.d/_S95picld
# mv /etc/init.d/picld /etc/init.d/.picld
```

Results	Verdict
Sysadmin knowledge daemon/processes	Pass
Procedure to start new daemon/processes	Fail

After interviewing the sysadmin, I realize that he has in -depth knowledge of daemon/processes.

However, no procedure of control exists for the start daemon/processes. After discussion with the sysadmin and his manager, it has been agreed that a small procedure will be addressed.

## 5.7.4 Kernel tuning

### 5.7.4.1 Network parameters

For the network parameters the results are the followings:

Command	Results
# /usr/sbin/ndd /dev/arp arp_cleanup_interval	60000
# /usr/sbin/ndd /dev/ip ip_ire_arp_interval	60000
# /usr/sbin/ndd /dev/ip ip_respond_to_echo_broadcast	0
# /usr/sbin/ndd /dev/ip ip6_respond_to_echo_multicast	0
# /usr/sbin/ndd /dev/ip ip_respond_to_time_stamp_broadcast	0
# /usr/sbin/ndd /dev/ip ip_respond_to_address_mask_broadcast	0
# /usr/sbin/ndd /dev/ip ip_ignore_redirect	1
# /usr/sbin/ndd /dev/ip ip6_ignore_redirect	1
# /usr/sbin/ndd /dev/ip ip_send_redirects	0
# /usr/sbin/ndd /dev/ip ip6_send_redirects	0
# /usr/sbin/ndd /dev/ip ip_respond_to_timestamp	0
# /usr/sbin/ndd /dev/ip ip_forwarding	0
# /usr/sbin/ndd /dev/ip ip6_forwarding	0
# /usr/sbin/ndd /dev/ip ip_strict_dst_multihoming	1
# /usr/sbin/ndd /dev/ip ip6_strict_dst_multihoming	1
# /usr/sbin/ndd /dev/ip ip_forward_directed_broadcasts	0
# /usr/sbin/ndd /dev/ip ip_forward_src_routed	0
# /usr/sbin/ndd /dev/ip ip6_forward_src_routed	0
# /usr/sbin/ndd /dev/tcp tcp_conn_req_max_q0	4096
# /usr/sbin/ndd /dev/tcp tcp_conn_req_max_q	1024
# /usr/sbin/ndd /dev/tcp tcp_time_wait_interval	240000
# /usr/sbin/ndd /dev/tcp tcp_fin_wait_2_flush_interval	675000
# /usr/sbin/ndd /dev/tcp tcp_keepalive_interval	7200000
# /usr/sbin/ndd /dev/tcp tcp_conn_hash   grep size	tcp_conn_hash_size=512
# more /etc/system   grep tcp_conn_hash_size	Nothing was displayed
# ndd /dev/tcp tcp_strong_iss	2

# more /etc/default/inetinit   grep TCP_STRONG_ISS=	TCP_STRONG_ISS=2
# ndd /dev/tcp tcp_rev_src_routes	0

Globally the results for the network parameters conforms to the values I defined in my checklist. I can give a pass to this test. The only values that don't correspond are shown in red in the above table. Three TCP timers have not been adjusted, and the size of the hash table is under -dimensioned.

The sysadmin agrees to adjust the size of the table by entering the following line in the /etc/system file: set tcp\_connect\_hash\_size=512

For the timer, the following line will be added in the existing script /etc/init.d/nddconfig. This script is called at boot time by /etc/rc2.d/S70nddconfig:

```
/usr/sbin/ndd -set /dev/tcp tcp_time_wait_interval 60000
/usr/sbin/ndd -set /dev/tcp tcp_fin_wait_2_flush_interval 67500
/usr/sbin/ndd -set /dev/tcp tcp_keepalive_interval 300000
```

Results	Verdict
After reboot the network parameters remained the same	Pass
These parameters are set in the /etc/init.d/nddconfig file called by /etc/rc2.d/S70nddconfig at boot. Other are stored in /etc/system. This file is parsed at boot time.	Pass

#### 5.7.4.2 User stack

As shown by the results below the user stack protection is enabled.

Results	Verdict
# more /etc/system   grep noexec_user_stack # set noexec_user_stack=1	Pass
# more /etc/system   grep noexec_user_stack_log # set noexec_user_stack_log=1	Pass

Note: A copy of the /etc/system is given in APPENDIX F (§ 14)

#### 5.7.4.3 File descriptor

The results of the files descriptor are shown below:

Results	Verdict
# ulimit -Hn	Fail ( increase to 2048)

1024	
# more /etc/system   grep rlim_fd_max #	Fail (not set in /etc/system file)
# ulimit -Sn 256	Fail (increase to 1024)
# more /etc/system   grep rlim_fd_cur #	Fail (not set in /etc/system file)
Reboot the system and redo the above test	Don't apply

Both limit will be increased by entering the following line in /etc/system :

```
set rlim_fd_cur = 1024
set rlim_fd_max = 2048
```

Note: A copy of the /etc/system is given in APPENDIX F (§ 14)

#### 5.7.4.4 Core file

Core file are disabled as shown in the results:

Results	Verdict
# more /etc/system   grep coredumpsize= # set sys:coredumpsize = 0	Pass
# ulimit -c 0	Pass
Results are the same after a reboot	Pass

Note: A copy of the /etc/system is given in APPENDIX F (§ 14)

### 5.7.5 File system

#### 5.7.5.1 partitions

Results	Verdict
<pre># df -k Filesystem            kbytes    used  avail c  capacity  Mounted on /dev/md/dsk/d1        4131866 2389000 1701548 59%    / /proc                  0         0      0      0%    /proc fd                     0         0      0      0%    /dev/fd mnttab                 0         0      0      0%    /etc/mnttab /dev/md/dsk/d3        3099287  52502 2984800  2%    /var swap                  1804536   16 1804520  1%    /var/run swap                   524288   16  524272  1%    /tmp /dev/md/dsk/d4        3099287  78642 1 2250881 26%    /opt/valis /dev/md/dsk/d5        6047826  6017 5981331  1%    /var/opt/valis</pre>	Don't apply

# mount -p /dev/md/dsk/d1 - / ufs - no rw,intr,largefiles,logging,onerror=panic,suid,dev=1540001 /proc - /proc proc - no dev=4040000 fd - /dev/fd fd - no rw,suid,dev=4100000 mnttab - /etc/mnttab mntfs - no dev=4200000 /dev/md/dsk/d3 - /var ufs - no rw,intr,largefiles,logging,onerror=panic,suid,dev=1540003 swap - /var/run tmpfs - no dev=1 swap - /tmp tmpfs - no size=512m,dev=2 /dev/md/dsk/d4 - /opt/valis ufs - no rw,intr,largefiles,logging,onerror=panic,suid,dev=1540004 /dev/md/dsk/d5 - /var/opt/valis ufs - no rw,intr,largefiles,logging,onerror=panic,suid,dev=1540005	Fail
--	------

More tightening can be applied on these partitions. The /var and /var/opt /valis should have the nosuid and noatime options set, because there is no SUID program installed on these partitions (according to the sysadmin) and it's not foreseen to install SUID programs on these partition in the future. In addition the

Note that the /opt/valis and /var/opt/valis are the partition dedicated to the application that will be further installed.

#### 5.7.5.2 Basic permission

Results	Verdict
Sticky bit is set on /tmp and /var/temp	Pass
I logged on to a non-root user, then I created a file in /tmp with permission set to 666. I su to another account and tried to delete the file. I got this message:  # rm: operation not permitted	Pass

The sticky bit is set in /tmp directory, and the restrictions work.

#### 5.7.5.3 World-write files

Results	Verdict
# find / -perm -o+w   wc -l # 57	Don't apply
# find / -perm -o+w See results in APPENDIX G (§ 15)	Fail

I found standard world write directory under /var partition (e.g. /var/mail, /var/preserve, /var/tmp, etc.). Most of these files are “normal”, and have the sticky bit set. However, some of these are not needed and can be removed reducing the number of world -write directories. The directory /var/spool/uucppublic can be removed as uucp is not used. Likewise the /var/spool/lp/fifos/public can be removed as no printer server is installed.

Then, I found 8 world-write directories under /usr/demo/wbem/. According to SUN “*WBEM is the Web-Based Enterprise Management initiative to standardize management information across platforms. To achieve this, the Common Information Model (CIM) defines a consistent schema for information about objects (e.g. disk drives, printers, applications, operating systems) in the system. This information would be made available to management applications via an open communication standard like eXtensible Markup Language (XML) and HyperText Transport Protocol (HTTP).*” I asked the sysadmin if this tool was effectively used, and the answer was negative. It was then decided to remove these directories.

In addition, I found 17 world-write files under the directory /ctadmin. These files are in fact the results of some packages download after a “tar -xvf” has been made. In agreement with the sysadmin we removed the write access to the other for these files.

Then it lefts 3 world -write dir under /opt/WebSphere/AppServer/ and 8 world -write file. These file correspond to the WebSphere Application server from IBM. This is in relation with the application that will be further installed to communicate with the Application server located in the secure zone (c.f. Figure 2). At the time of the audit it was not possible to determine if these files and directories can have their world -write permission removed.

Results	Verdict
All world directories where not in Home directories, nor in the root search path.	Pass
The standard world directories have the sticky bit set	Pass
Knowledge of sysadmin regarding world -write files and directories found: it was OK except for the WebSphere files and directories	Pass/Fail

#### 5.7.5.4 SUID files

Results	Verdict
# find / -type f \( -perm -u+s -o -g+s \) -ls See APPENDIX H (§ 16) for the results	Fail

I found 92 SUID/SGID files. In fact no measures has been applied to limit the number of SUID/SGID file to the minimum necessary. It has been agreed with the sysadmin to limit these files to the list I

provide in the first part of this document in § 4.2.5.4, on the test server, and it's OK to do the same on the production server.

### 5.7.6 Account and Password

Results	Verdict
# egrep "PASSLENGTH" /etc/default/passwd # PASSLENGTH=8	Pass
I tried to change a password account to a length smaller than 8 with passwd command => not possible (see Figure 3)	
# egrep "MAXWEEKS" /etc/default/passwd # MAXWEEKS=8 # egrep "WARNWEEKS" /etc/default/passwd # WARNWEEKS=1	Pass
# logins -d -x -m #  # logins -p -x #	Pass (nothing was returned)  Pass(nothing was returned)
# more /etc/passwd All accounts have an "x" in the password field in /etc/passwd => shadow file is used # more /etc/passwd root, daemon, adm, bin, sys, lp, uucp, nobody, noaccess, are the only accounts	Pass  Fail
# more /etc/passwd daemon, adm, bin, sys, lp, uucp, nobody, noaccess, have invalid shell	Pass
# passwd -sa LK in front of the accounts: daemon, adm, bin, sys, lp, uucp, nobody, noaccess => all these accounts are blocked	Pass

```

$
$
$
$
$
$ passwd
passwd: Changing password for fortest
Enter login password:
New password:
passwd(SYSTEM): Password too short - must be at least 8 characters.
New password:

```

Figure 3: trying to set a password with a length < 8

All the tests passed except for the minimum number of account: the uucp and lp account can be removed. Otherwise everything conforms to our requirements.

Results	Verdict
I approved the method chosen by the sysadmin to find strong passwords. It included number, capital and special characters .	Pass
The sysadmin ensured me that he was able to remember the password root account without being able to store a written copy somewhere	Pass
Procedure when person is leaving the company: doesn't exist	Fail
Procedure to give access to someone to the machine: doesn't exist	Fail

Apparently, the sysadmin has only three passwords to remember, one being for the root account of the audited machine. In this case it's possible for him not to forget the password and therefore we can admit that a copy of the password is not needed.

When interviewing the sysadmin, I realized that no written control procedure was in place for the attribution of an account on this machine. Furthermore, no control procedure exists when a user is leaving the company.

After discussion with the manager, it was agreed to set up a procedure for both cases, and to maintain a list of users having access to the server, and their role regarding this server.

### 5.7.7 Console

Results	Verdict
# grep CONSOLE= /etc/default/login # CONSOLE=/dev/console	Pass
# grep PASSREQ= /etc/default/login # PASSREQ=YES	Pass
# grep SYSLOG= /etc/default/login # SYSLOG=YES	Pass
I tried to login with an empty password => the login was refused. This attempt was logged on the /var/adm/messages log file	Pass

The direct root login is only authorized on the console. A password is required and the attempted login are logged in /var/adm/messages.

### 5.7.8 Umask



Results	Verdict
# umask 022	Fail Should be 027 or 077
# grep UMASK /etc/default/login # UMASK=022	

The Umask is too permissive, for this type of server. It was agreed with the sysadmin to put it to 027 in /etc/default/login file.

Results	Verdict
I logged with SSH as root and typed # umask 022	Fail Should be 027 or 077
The shell used for root is /bin/sh no umask was set in the shell configuration files	Fail

When we log in with SSH the umask taken into account is the one specified in the shell configuration files. If nothing is specified, the shell takes the system default umask which is 022. After discussion with the sysadmin, it appears that they use only the /bin/sh shell (which is a good choice regarding security). It was then agreed to add the line “umask 027” in the file /etc/profile, and to add a .profile in all home directory with the same line (note that the \$HOME/.profile shall be owned by root, and readable by all.). I also recommend to remove all other shell like bash, csh, ksh, etc.

Results	Verdict
# grep CMASK= /etc/default/init # CMASK=022	Pass

The Cmask is correctly set.

### 5.7.9 Path

Results	Verdict
# echo \$PATH  /usr/bin:/usr/sbin:/usr/ccs/bin:/usr/bin:/bin:/usr/sbin:/sbin :/usr/bin:/usr/sbin:/usr/ucb:/etc:/usr/openwin/bin:/usr/ccs/b in:/usr/local/bin:/usr/bin/nsr:.	Fail the “.” shall be removed

The sysadmin agreed to remove the dangerous “.” in the PATH

### 5.7.10 Cron

Results	Verdict
<pre># more /etc/cron.deny daemon bin smtp nuucp listen nobody noaccess lp sys adm uucp  # more /etc/at.deny daemon bin smtp nuucp listen nobody noaccess root sys adm lp uucp</pre>	Pass

The at.deny and cron.deny have account except root listed. I checked these setting by trying the crontab command from a non root account. The result is shown below:

Results	Verdict
<pre>From root I su to a non -root account and type:  # crontab -l crontab: you are not authorized to use cron. Sorry.</pre>	Pass

### 5.7.11 TCPWrapper

Results	Verdict
<pre># which tcpd no tcpd in /usr/bin /usr/sbin /usr/ccs/bin /usr/bin /bin /usr/sbin /sbin /usr/bin /usr/sbin /usr/ucb /etc /usr/openwin/bin /usr/ccs/bin /usr/local/bin /usr/bin/nsr</pre>	Fail

TCPwrapper is not installed	Fail
-----------------------------	------

I did not find tcpd the binary for TCPwrapper with the “which” command. So I asked the sysadmin whether TCPwrapper was installed or not. The answer was no, but he agreed to install it and configure it very soon.

### 5.7.12 SSH

Results	Verdict
# ssh -V SSH Version OpenSSH_2.2.0p1, protocol versions 1.5/2.0.	Fail update to OpenSSH_2.9.9p

At the time of the audit the last version was OpenSSH\_2.9.9p. The version that I found installed on the server had known bugs. The sysadmin agreed to update it to the last version.

Results	Verdict
# ls -la /etc/sshd_config -rw-r----- 1 root root 1584 Jul 12 17:53 etc/sshd_config	Fail should be 400

The configuration file of sshd server should have more restrictive permission. The sysadmin agreed to set it to 400.

The important configuration parameters of the `etc/sshd_config` file are shown below:

Results	Verdict
Protocol 2,1 SyslogFacility AUTH LogLevel INFO ServerKeyBits 768 PermitRootLogin yes PermitEmptyPasswords no IgnoreRhosts yes StrictModes yes IgnoreRhosts yes RhostsAuthentication no RhostsRSAAuthentication no HostbasedAuthentication no RSAAuthentication no PubkeyAuthentication no PasswordAuthentication yes	Pass (except for protocol)

UseLogin no	
I created an account with an empty password then I tried to log on with an empty password. The login was refused (see Figure 4). This attempt was also logged in /var/adm/messages with the following line: Oct 28 16:26:08 xxxx sshd[813]: [ID 800047 auth.info] Failed password for forttest from xxxxxx port 1303	Pass
I also put a trusted host in /etc/hosts.equiv. I tried to use the trust without and with the public key host based authentication (i.e. with the right pubkey set). The trust was refused.	Pass

It's better to set the protocol only to 2, to prevent a remote client to downgrade the connection to protocol number 1, to perform an attack that exploits a weakness of protocol number 1. The sysadmin wanted to be sure that this protocol was not needed before removing it, and wanted to perform some tests on the test server. I agreed.

Direct root login are possible. The sysadmin explained me that only two users have access to this account, and that there is not so much benefit to disallow direct root login. I agreed.

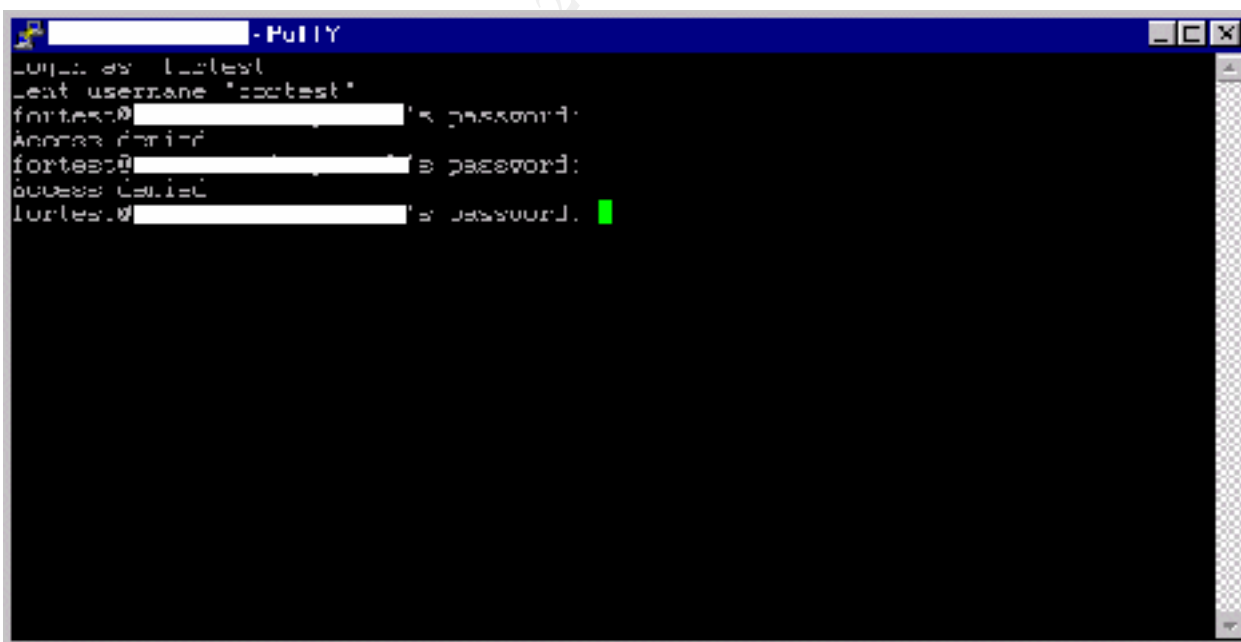


Figure 4: login with an empty password with ssh

### 5.7.13 Logging

Results	Verdict
No written logging policy exists.	Fail
Policy is not comprehensive	Fail
The logs are not viewed on regular basis	Fail/Pass
No priority are defined form alarm	Fail
No procedure exists to inform responsible persons in case of serious alarm	Fail

After interviewing the sysadmin, it happens that no written and approved logging policy existed. I then asked him, if he implicitly uses one. The answer was negative and he admitted that there was a big lack in this matter. Furthermore, I asked him if he checks the logs on a regular basis. He replied me with a hesitating “yes”. So I further asked him, when exactly, and how many times per week/day. I also tried to know if it was an official task of his job (e.g. defined by the management). It appears that the log were actually monitored, but more on a random basis than as a regular task.

No procedure exists to inform responsible persons (e.g. manager) in case of serious problems detected, and no recovery plan has been foreseen.

The syslog.conf file shown below reflects the lack of clear log policy:

Results	Verdict
Syslog is used, but no tool like logcheck is installed to prioritize the alarm, and analyze the logs.	Fail
<pre># more /etc/syslog.conf *.err;kern.notice;auth.notice /dev/console *.alert                        root *.emerg                       * *.debug                       /var/adm/messages</pre>	Fail
Not all facilities are logged under a different log file	Fail
Event shall be logged from at least info level	Fail
/var/log/*log have a permission of 644 (not 600)	Fail

The configuration of the syslog.conf file doesn't include all facilities in a separate file (all facilities from debug level are sent to /var/adm/messages, all emergency messages are sent to all terminal, alert are sent to the local root mail box). This configuration doesn't permit to rapidly search an event, in case of alarm. The sysadmin agreed to change the syslog.conf configuration like it was mentioned in §4.2.13, or like the one mentioned in APPENDIX B(§ 10).

For alarm prioritization and log analysis, the sysadmin has said that he's going to investigate for tools, test and implement them. He also asked me to help him in the task of defining which event has to be considered as serious security alarm.

Results	Verdict
Policy for alarm prioritization not enforced	Fail

#### 5.7.14 Integrity software

Results	Verdict
# which aide # # which tripwire # No integrity software is present	Fail

After interviewing the sysadmin it appears that neither AIDE nor Tripwire (neither any other integrity check software) was installed when the machine was audited. The sysadmin assured me that he will test both software and then implement them rapidly.

#### 5.7.15 Policy for security patches

Results	Verdict
No clear patching policy established	Fail
Policy followed by sysadmin	Fail
Mailing list subscription	Pass/Fail
Verification of secure state	Fail

When I asked the sysadmin for their patching policy the answer was: “ *When we receive a notice, we check if it's worth to install the patch or if it 's necessary...* ”. Then I asked him what he meant by “worth to install” or “necessary”. He remained silently... In fact, no written policy existed, and no rationale was made regarding the priority for patch installation. Then, I proposed him to follow the rules I have listed in § 4.2.15. He agreed saying that it was more or less what they use to do implicitly...

For the mailing list subscription, the sysadmin receives the CERT -advisory mail, and the SUN security bulletins. For the back up software, he doesn't directly receive e -mail from Legato, but the Backup server sysadmin is responsible for this, and sends e -mail, whenever a patch is needed for the client backup software installed on the web server.

The sysadmin should also consider being on the mailing list of IBM for patches concerning, WebSphere, and IBMHTTPD web server.

For the check that the secure state was maintained after a patch installation, the sysadmin agreed, but said that if each time a new patch is installed, we have to conduct an audit like this one, it would take too much time and cost too much money. I did agree, and we decided to make a script to check important things only, or to run checking tool like the CIS benchmark.

#### 5.7.16 Backup and recovery policy

Results	Verdict
Backup policy exists	Pass
Backup policy is enforced	Pass

There is a good backup policy in place. Incremental backups are made every night, and full backups are made every week. In addition, each month a full backup copy is stored in a safe located in the physically protected room in the basement. The backups are made with the Legato software technology. The backup server is located in the same room as the safe. To get into the room access is controlled with the contactless chip card each employee of the company has. Only 3 persons have access to this room, and manage the backup server and tapes.

I asked if they tested the restore function of the backups. They answered that they did it with other servers and that it worked well, but for the web server we are auditing, it has not been tested yet. I suggested them to add in their policy the steps necessary to recover this web server from system failures and other security incidents and to perform a test the restore.

The following picture explain how back ups are made with Legato backup software (this picture is taken from: <http://osb.wff.nasa.gov/osbnet/man/backup/html/SOLBCKUPADMIN/appA.html> )

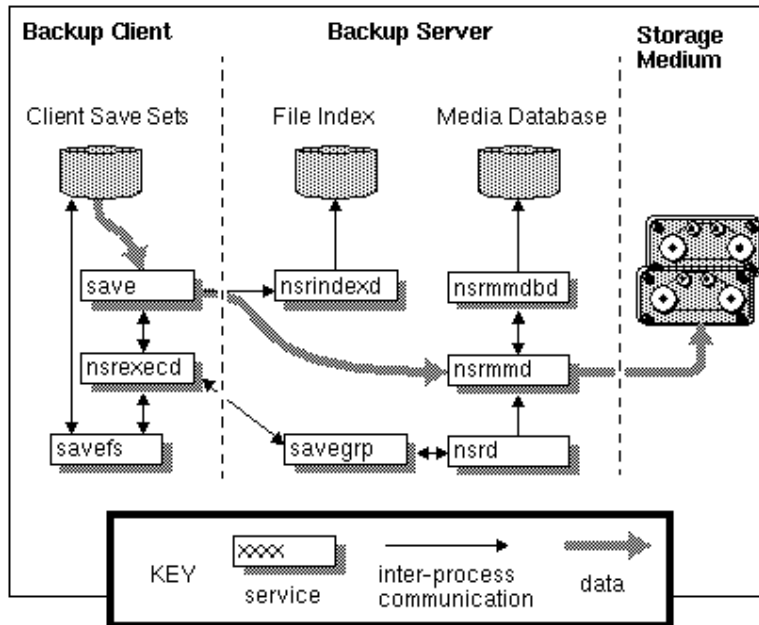


Figure 5: Backup Daemon Processes and Programs Interact During a Scheduled Save

For more detail on how it works please refer to the above mentioned URL.

## 6 Evaluation of the system

To make an evaluation of system I summed the “Pass” weighted with their risk and the “Fail” weighted with their risk. Then I calculated the percentage of Pass and Fail:

$$\begin{aligned}\Sigma_p &= \Sigma \text{Pass}_i \times \text{Risk}_i \\ \Sigma_f &= \Sigma \text{Fail}_j \times \text{Risk}_j \\ \text{Percent of Pass} &= \left[ \frac{\Sigma_p}{(\Sigma_f + \Sigma_p)} \right] \times 100 \\ \text{Percent of Fail} &= \left[ \frac{\Sigma_f}{(\Sigma_f + \Sigma_p)} \right] \times 100\end{aligned}$$

The results are:

$$\Sigma_f = 163.67$$

$$\Sigma_p = 145.67$$

$$\text{Percent of Pass} = 47.1 \%$$

$$\text{Percent of Fail} = 52.9 \%$$

The following results show that the the system comply to our security requirements only at 47.1%. It's obviously not enough. To analyse a bit more the results, I sorted the failed items from the riskiest to less risky item. Here's what we get:



<i>Items that failed</i>	<i>Estimated risks</i>
OS patches	9
SSH version	9
log policy is enforced	9
log tools installed and configured	9
Patching policy enforced	9
Inetd services stopped	8
SSH protocol	8
Installed packages	7.33
Log policy exist	7.33
Integrity check software installed and well configured	7.33
Patching policy exist	7.33
Procedure when user leaves company	7
Secure state after applying patches	6.66
SUID SGID file	6.33
TCP wrapper installed and configured	6.33
Log policy is comprehensive	6.33
procedure to add a new user and give authorization to access the machine	5.33
Umask	5.33
Path variable	5.333
TCP defenses	4.66
Partitions	3.66
World-write files and dir	3.33
400 /etc/sshd_config	3.33
strategy for installed packages	2.33
Procedure to start a new network service	2
Procedure to start a new daemon	2
minimum set of accounts	2
File descriptor	1.33

A quick look at this table shows us that the immediate measures to take are (level 7, 8 and 9): patching the system; establishing a comprehensive patching policy for future patch installation; establishing a comprehensive log policy and implement it with the right tools; upgrading SSH; stopping the unnecessary network services; minimizing the installed packages; integrity check software installed and configured; and the procedure when a user leaves the company.

When I showed this table this helped me to convince the manager, to apply immediately the above mentioned measures. I also gave him an estimation of the costs for such measures.

These costs are summarized in the Table 1.

To patch the system, I estimated that, first, the sysadmin has to do it on the test system and then apply it on the production system. Taking into account that he has to observe the behavior of the system and check that the patches did install or start something we have stopped or disabled, I counted two days of work for the sysadmin.

Measures	Cost calculation	Total
Patching the system.....	two days sysadmin: 2 x 8h x \$85	\$1360
Establish a patching policy.....	half a day sysadmin: 4h x \$85	\$340
	half a day for me: 4h x \$100	\$400
Minimizing the packages.....	three days sysadmin: 3 x 8h x \$85	\$2040
	half a day for me: 4h x \$100	\$400
Establish a log policy.....	half a day sysadmin: 4h x \$85	\$340
	half a day for me: 4h x \$100	\$400
Implement log policy.....	half a day for me: 4h x \$100	\$400
	one day sysadmin: 8h x \$85	\$680
Upgrade SSH and configure it.....	half a day sysadmin: 4h x \$85	\$340
Integrity check SW installation.....	half a day sysadmin: 4h x \$85	\$340
	half a day for me: 4h x \$100	\$400
Procedure when user leaves cp ny...	2 hours manager: 2h x \$110	\$220
	2 hours sysadmin: 2h x \$85	\$170
	2 hours me: 2h x \$100	\$200
Manager approval and supervision	half a day manager: 4h x \$110	\$440
<b>Total</b>		<b>\$8'470</b>

Table 1: cost estimation for the immediate measures

For the log policy I proposed to the manager to do it in collaboration with the sysadmin. As we already have some good examples, this shouldn't take long as far as the sysadmin agrees with such policies. I estimated the time to half a day for the sysadmin and half a day for me. To implement this policy I estimated to half a day for me (to show the sysadmin how to use the tool and how to tune the alarm), and one day for the sysadmin (half a day with me and half a day of fine tuning). Note that the item "*log policy is comprehensive*", is also included here even if its risk level was estimated at 6.33, because when we will establish the log policy we will make it comprehensive as well.

To install a new version of SSH and to configure it to allow only protocol 2, I estimated the time to half a day for the sysadmin.

To minimize the packages, it has first to be applied on the test server. We have to go from the minimum packages required for the OS, and then add the packages for the applications. I estimated to three days of work for the sysadmin. I also included half a day for me, to explain him how to proceed and where to find the information for the required packages.

For the integrity check software, I just counted half a day, because we already have good examples of configuration both for Tripwire and AIDE.

For the procedure when a user leaves a company, we also have some drafts that I can propose to the manager. I estimated that this task could be accomplished in two hours.

Finally, I added the time (estimated to half a day) for the manager, considering that he has to approve the policies and procedures, and that we have to present him the results when the immediate measure will be completed.

All in all for the implementation of the immediate measures it costs \$ 8'470. Note that when these immediate measures will be applied to the system it will meet the requirements at 80%!

After showing these results to the manager, he decided to apply the measures immediately, in order to complete them within a time frame of two weeks, and to present him the status after one week. He also decided not to install the application (the web pages, and the application that communicates with the application server) before these measures are completed. He added that he wanted us to implement the rest of the measures within a time frame of one month.

## **7 Evaluate the audit**

The audit conducted here was valuable because it permitted to discover the weaknesses of the system. Moreover, I was able to quantitatively estimate whether the system complied to the checklist, and therefore to say if the security requirements were met. It was also possible to show to the management what was the root causes why the system did not pass the audit and to give him a cost estimation.

Each item of the checklist was carefully tested except one: the backup software from Legato, was a bit overlooked. I gave a pass to this item based only on the interview I made with the sysadmin. I did not check the configuration on the backup server. More investigation on how this software works and more checks should be done (security history, patches to apply, etc.).

This audit is not sufficient to put the server in production. The scope of this study was to evaluate the system without the web application and the application that communicates with the application server. However, auditing the web server configuration and the web application is as important as the audit that we did here. Just as a reminder, CGI vulnerabilities are in "The Twenty Most Critical Internet Security Vulnerabilities". This is the reason why an audit based on similar principles (security requirements, checklists, risks evaluation) should be conducted as soon as possible.

This audit was done manually because it was the first time we applied the checklist defined in the first part of this paper. However it should have been judicious to apply tools like the CIS benchmark, TITAN, COPS, Tara, etc., and to compare the obtained results. If the checked items and the results are similar then we should consider using these tools for future audits. If these tools can not be used to automate the audit, then we can think of implementing it through scripts or a program. While it takes time at the beginning to code them, it can be very useful if same audits have to be performed. Furthermore, the output can be formatted to a nice HTML report.

## 8 References

- [1] Solaris Benchmark v1.0.1b, The Center for Internet Security (CIS), September, 2001.  
<http://www.cisecurity.org/solaris/SolarisBenchmark.pdf>
- [2] UNIX Security Checklist v2.0, Australian Computer Emergency Response Team (AusCERT) and the CERT, 8 Oct. 2001  
[http://www.auscert.org.au/Information/Auscert\\_info/Papers/usc20.html](http://www.auscert.org.au/Information/Auscert_info/Papers/usc20.html)  
[http://www.cert.org/tech\\_tips/usc20\\_full.html](http://www.cert.org/tech_tips/usc20_full.html)
- [3] **Auditing Unix (Solaris)**, Sans institute resources , *Prepared by: Krishni Naidu*  
[http://www.sans.org/checklist/unix\\_check.htm](http://www.sans.org/checklist/unix_check.htm)
- [4] Solaris™ Operating, Environment Security, *Updated for Solaris 8 Operating Environment*  
*By Alex Noordergraaf and Keith Watson*  
*Sun BluePrints™ OnLine - April 2001*  
**<http://www.sun.com/blueprints>**  
Part No.: 816-0428-10  
Revision 01, April 2001  
available at [http://www.sun.com/blueprints/0401/security\\_updt1.pdf](http://www.sun.com/blueprints/0401/security_updt1.pdf)
- [5] Solaris™ Operating Environment Network Settings for Security *By Alex Noordergraaf and Keith Watson – Global, Enterprise Security Service*  
*Sun BluePrints™ OnLine - December 1999*  
**<http://www.sun.com/blueprints>**  
Part No.: 806-4049-10  
Revision 01, December 1999  
available at: [http://www.sun.com/blueprints/1200/network\\_updt1.pdf](http://www.sun.com/blueprints/1200/network_updt1.pdf)
- [6] Securing UNIX Step by Step, K. Koenigsknecht, SANS GCUX practical, November 2001  
[http://www.giac.org/practical/Kurt\\_Koenigsknecht\\_GCUX.doc](http://www.giac.org/practical/Kurt_Koenigsknecht_GCUX.doc)
- [7] Installing And Securing Solaris 8, T. Raborn, SANS GCUX practical, September 2001  
[http://www.giac.org/practical/Timothy\\_Raborn\\_GCUX.zip](http://www.giac.org/practical/Timothy_Raborn_GCUX.zip)
- [8] Solaris8 Core package installation for a Firewall-1, Lance Spitzner  
<http://www.enteract.com/~lspitz/core8.txt>
- [9] Solaris™ Operating Environment Minimization for Security: A Simple, Reproducible and Secure Application

cation Installation Methodology  
*Updated for Solaris 8 Operating Environment*  
*By Alex Noord ergraaf - Enterprise Engineering*  
*Sun BluePrints™ OnLine - November 2000*

**<http://www.sun.com/blueprints>**

Part No. 806 -4050-10

Revision 03, November 2000

[http://www.sun.com/software/solutions/blueprints/1100/minimize\\_updt1.pdf](http://www.sun.com/software/solutions/blueprints/1100/minimize_updt1.pdf)

- [10] IP stack tuning by Rob Thomas.  
[http://www.enteract.com/~robt/Docs/Articles/ip\\_stack-tuning.html](http://www.enteract.com/~robt/Docs/Articles/ip_stack-tuning.html)
- [11] Solaris™ 2.x - Tuning Your TCP/IP Stack and More by [Jens-S. Vöckler](http://www.rvs.uni-hannover.de/people/voeckler/tune/EN/tune.html)  
<http://www.rvs.uni-hannover.de/people/voeckler/tune/EN/tune.html>
- [12] SUN's online reference manual: TCP/IP tunable parameters available at:  
[SUN's reference manual for TCP/IP tunable parameters](http://www.sun.com/documentation/tcpip/tunable.html)
- [13] Avoiding the TCP TIME\_WAIT state at Busy Servers, T. Faber, J. Touch, W. Yue, August 1997  
<http://globecom.net/ietf/draft/draft-faber-time-wait-avoidance-00.html>
- [14] IBM WebSphere Application Server Standard Tuning Guide, Advanced Edition  
<http://www-4.ibm.com/software/webervers/appserv/doc/v35/ae/infocenter/was/0901.html#b173>
- [15] Optimizing HTTP performance, Oracle, 2000  
<http://uisnt1.humboldt.edu/9ias/installdocs/doc/a86828/listener.htm#1007115>
- [16] Successful Solaris performance tuning, SYSADMIN, the journal for Unix system administrator, 2002.  
<http://www.samag.com/documents/s=1323/sam0110e/0110e.htm>
- [17] Limiting SUID files, Sean Boran  
[http://www.boran.com/security/sp/Solaris\\_hardening4.htm#Limiting SUID Files](http://www.boran.com/security/sp/Solaris_hardening4.htm#Limiting_SUID_Files)
- [18] Security Review: Solaris 8 Setuid/Setgid Files, Information Systems and Technology, University of Waterloo, 09-Nov-2000  
[ist.uwaterloo.ca/security/howto/2000-08-17.html](http://ist.uwaterloo.ca/security/howto/2000-08-17.html)
- [19] Solaris 8 Installation Checklist, GCUX Practical assignment Jeff Campione,  
[http://www.sans.org/y2k/practical/Jeff\\_Campione\\_GCUX.htm](http://www.sans.org/y2k/practical/Jeff_Campione_GCUX.htm)

- [20] Securing Solaris Servers - A Checklist Approach, *Paul D. J. Vandenberg and Susan D. Wyess*, November 98  
<http://www.usenix.org/sage/sysadmins/solaris/>
- [21] Solaris Security Checklist, The Singapore Computer Emergency Response team, SingCERT, 1999  
[http://www.singcert.org.sg/archive/security\\_checklists/Solaris\\_checklist.pdf](http://www.singcert.org.sg/archive/security_checklists/Solaris_checklist.pdf)
- [22] Unix Security Checklist, University of Nevada, Reno  
<http://equinox.comnet.unr.edu/homepage/unixweb/security/checklist.html>
- [23] Practical UNIX and Internet Security, 2nd Edition  
*Garfinkel and Spafford*  
*O'Reilly & Associates*, 1996  
ISBN 1-56592-148-8  
<http://www.oreilly.com/catalog/puis/>
- [24] Solaris Security, first edition, Peter H. Gregory, Woodinville, Washington, © Copyright 2000  
ISBN: 0-13-096053-5.
- [25] Solaris implementation of processes, threads, and lightweight processes, by Jim Mauro, August 98.  
<http://sunsite.uakom.sk/sunworldonline/swol-08-1998/swol-08-insidesolaris.html>
- [26] Solaris Default Processes by By [Hal Flynn <hmflynn@earthlink.net>](mailto:hmflynn@earthlink.net), May 29, 2000.  
<http://www.1.securityfocus.com/focus/sun/articles/b3.html>
- [27] Reg Quinton/Bruce Barnett's CheckPatches, CheckPatches.cron, GetApplyPatch, GetApplyPatch.cron  
<http://ist.uwaterloo.ca/~reggers/drafts/>
- [28] A SunSolve Patch Primer at [sunsolve.sun.com/pub-cgi/show.pl?target=content/content1](http://sunsolve.sun.com/pub-cgi/show.pl?target=content/content1)
- [29] Casper Dik's FastPatch at <ftp://www.wins.uva.nl/pub/solaris/auto-install/>

## 9 Appendix A: aide.sh example file

```
#!/bin/sh
#
# Shell script to check integrity and to manage aide db files
# Written by Azim Ferchichi 17.10.2001
#
LogFile=/var/log/warn
BinFile=/usr/bin/aide
AideLogFile=/var/log/aide.log
AideDBFile=/root/AIDE/aide.db
NewAideDBFile=/root/AIDE/aide.db.new
TMPFILE=/root/TMP/aidecheck.tmp

if test -x $BinFile
then
    if test -f $AideLogFile
    then
        $BinFile --check > $TMPFILE
        if test -s $TMPFILE
        then
            echo -e "\n\n----- New Entry----- \r" >> $AideLogFile
            echo `date` >> $AideLogFile
            echo -e "\n" >> $AideLogFile
            more $TMPFILE >> $AideLo gFile
        fi
        rm -f $TMPFILE
    else
        $BinFile --check > $AideLogFile
    fi

    $BinFile --update > /dev/null
    mv $NewAideDBFile $AideDBFile
else

    echo -n `date` >> $LogFile
    echo -e " Warning: aide binary file not found. No integrity check has been done \r" >> $LogFile
fi
#
# end of script
```

## 10 Appendix B: example of syslog.conf file

```
# /etc/syslog.conf      syslog configuration file.
#
# This file is processed by m4 so be careful to quote (') names
# that match m4 reserved words. Also, within ifdef's, arguments
# containing commas must be quoted.
#
# Facilities:   kern          Priorities:   emerg
#              user          (highest first) alert
#              mail          crit
#              daemon          err
#              auth          warning
#              syslog        notice
#              lpr           info
#              mark (timestamps) debug
#
# news
# uucp
# cron          none
# local0..7     [don't send any messages]
#
# Funnies: 0. 'mail.info' logs all mail messages of priority
#           'info' OR HIGHER (i.e. not just priority 'info')!
#           1. you can do *.priority but not facility.* !!
#           2. do "m4 syslog.conf" to check preprocessing
#           3. Must use tabs (not spaces) between selection and action
#           4. "kern,mail.info" logs kern & mail messages of
#              at least priority info. The same for other combinations.
#           5. Long lines don't work.
#           6. You can have MAXIMUM 20 (non comment) lines in this file.
#              (The rest are silently ignored..)
#
# Debugging: . start syslogd with "-d" to enable debug output
#            . send a HUP to syslogd each time you change this file
#            . use /usr/ucb/logger to send test messages to
#              each facility.priority, for example:
#            /usr/ucb/logger -p mail.warn "test from sean"
#
# For lots of messages on the console uncomment this:
#*.err;kern.warning;auth.err;daemon.err           /dev/console
```



```

# For minimal console messages, such as "SU":
auth.err          /dev/console

# To alert logged on root or operator user to import events:
#*.alert;kern.err;daemon.err          operator
#*.alert                                     root

# display emergencies on all terminals (uses WALL)
*.emerg           *

#print time on console every 20mins (not needed if you have contool)
#mark.*           /dev/console

kern.info         ifdef(`LOGHOST', /var/log/kernlog, @loghost)
user.info         ifdef(`LOGHOST', /var/log/userlog, @loghost)
mail.info         ifdef(`LOGHOST', /var/log/maillog, @loghost)
daemon.info       ifdef(`LOGHOST', /var/log/daemonlog, @loghost)
auth.info         ifdef(`LOGHOST', /var/log/authlog, @loghost)
lpr.info          ifdef(`LOGHOST', /var/log/lprlog, @loghost)
news,uucp.info    ifdef(`LOGHOST', /var/log/newslog, @loghost)
cron.info         ifdef(`LOGHOST', /var/log/cronlog, @loghost)

## other "local" messages not yet used
local0,local1.info  ifdef(`LOGHOST', /var/log/local0log, @loghost)
local2,local3,local4.info  ifdef(`LOGHOST', /var/log/local2log, @loghost)
local5,local6,local7.info  ifdef(`LOGHOST', /var/log/local5log, @loghost)

# Put all alerts (& higher) into a seperate log:
*.err             ifdef(`LOGHOST', /var/log/alertlog, @loghost)

```

## 11 APPENDIX C: Matrix of risks assessments

### Thomas' Matrix

Audit Item	Likelihood	consequences, damages	Risk
Installed packages	3	3	9
strategy for Installed packages	2	2	4
OS patches	3	3	9
inetd service	3	3	9
Minimum boot network services running	3	3	9
Sysadmin knowledge of running network services	3	3	9
Procedure to start a new network service	2	2	4
Minimum running processes	2	1	2
Knowledge of running processes	2	2	4
Procedure to start a new daemon	2	2	4
ARP defenses	1	3	3
ICMP defenses	3	3	9
IP defenses	3	3	9
TCP defenses	3	3	9
Persistence after reboot	3	3	9
User stack protection	2	3	6
File descriptor	1	1	1
Core files	1	1	1
Partitions	2	2	4
/tmp sticky bit	1	1	1
World-write files and dir	2	2	4
SUID SGID file	3	3	9
Password length 8	1	2	2
Password expiration	3	3	9
no duplicate accounts	1	1	1
accounts with null password	3	3	9
Shadow file enforced	3	3	9
minimum set of accounts	1	1	1
Non-user account locked and non-valid shell	3	3	9
Method to choose strong passwd	2	3	6
Password storage	3	3	9
Procedure when user leaves company	3	3	9
procedure to add a new user and give authorisation to access the machine	3	3	9
console login	1	3	3
Umask	3	2	6

CMASK	3	2	6
Path variable	1	3	3
cron.deny at.deny	1	1	1
TCP wrapper installed and configured	3	2	6
SSH version	3	3	9
SSH protocol	3	3	9
log of SSH events	3	2	6
SSH server key length	1	1	1
Root login	1	1	1
empty passwd	3	3	9
No trusted hosts	2	3	6
RSA or passwd authentication	2	2	4
400/etc/sshd_config	2	2	4
Log policy exist	3	3	9
log policy is comprehensive	2	3	6
log policy is enforced	3	3	9
log tools installed and configured	3	3	9
Integrity check software installed and well configured	2	2	4
Patching policy exist	3	3	9
Patching policy enforced	3	3	9
Secure state after applying patches	3	3	9
Backup and recovery policy exist	2	3	6
Backup and recovery policy enforced	3	3	9

### Stephane's Matrix

Audit Item	Likelihood of intrusion if audit item failed (1 - low / 2 - medium / 3 - high)	consequences, damages (1 - low / 2 - medium / 3 - high)	Risk
Installed packages	3	3	9
strategy for Installed packages	1	1	1
OS patches	3	3	9
Inetd services	2	3	6
Minimum boot network services running	2	3	6
Sysadmin knowledge of running network services	1	1	1
Procedure to start a new network service	1	1	1
Minimum running processes	2	3	6
Knowledge of running processes	1	1	1
Procedure to start a new daemon	1	1	1
ARP defenses	2	2	4
ICMP defenses	1	2	2
IP defenses	2	2	4
TCP defenses	1	1	1
Persistency after reboot	3	3	9
User stack protection	2	3	6

File descriptor	1	2	2
Core files	1	2	2
Partitions	1	3	3
/tmp sticky bit	1	1	1
World-write files and dir	1	2	2
SUID SGID file	2	3	6
Password length 8	2	3	6
Password expiration	2	2	4
no duplicate accounts	1	1	1
accounts with null password	3	3	9
Shadow file enforced	1	2	2
minimum set of accounts	2	1	2
Non-user account locked and non-valid shell	2	2	4
Method to choose strong passwd	2	3	6
Password storage	3	3	9
Procedure when user leaves cpny	2	3	6
procedure to add a new user and give authorisation to access the machine	2	2	4
console login	1	3	3
Umask	2	2	4
CMASK	1	2	2
Path variable	3	3	9
cron.denyat.deny	1	1	1
TCP wrapper installed and configured	2	2	4
SSH version	3	3	9
SSH protocol	3	3	9
log of SSH events	1	1	1
SSH server key length	1	1	1
Root login	3	3	9
empty passwd	3	3	9
No trusted hosts	3	3	9
RSA or passwd authentication	3	3	9
400 /etc/sshd_config	2	2	4
Log policy exist	3	3	9
log policy is comprehensive	3	3	9
log policy is enforced	3	3	9
log tools installed and configured	3	3	9
Integrity check software installed and well configured	3	3	9
Patching policy exist	3	3	9
Patching policy enforced	3	3	9
Secure state after applying patches	3	3	9
Backup and recovery policy exist	1	3	3
Backup and recovery policy enforced	1	3	3

### Azim's Matrix

Audit Item	Likelihood of intrusion if audit item failed (1 - low / 2 - medium / 3 - high)	consequences, damages (1 - low / 2 - medium / 3 - high)	Risks
------------	---	--	-------

Installed packages	2	2	4
strategy for Installed packages	2	1	2
OS patches	3	3	9
Inetd services	3	3	9
Minimum boot network services running	3	3	9
Sysadmin knowledge of running network services	2	2	4
Procedure to start a new network service	1	1	1
Minimum running processes	2	2	4
Knowledge of running processes	2	2	4
Procedure to start a new daemon	1	1	1
ARP defenses	1	2	2
ICMP defenses	1	2	2
IP defenses	1	2	2
TCP defenses	2	2	4
Persistency after reboot	2	2	4
User stack protection	2	3	6
File descriptor	1	1	1
Core files	1	2	2
Partitions	2	2	4
/tmp sticky bit	1	1	1
World-write files and dir	2	2	4
SUID SGID file	2	2	4
Password length 8	2	3	6
Password expiration	2	3	6
no duplicate accounts	1	2	2
accounts with null password	3	3	9
Shadow file enforced	2	3	6
minimum set of accounts	1	3	3
Non-user account locked and non-valid shell	1	3	3
Method to choose strong passwd	2	3	6
Password storage	2	3	6
Procedure when user leaves cpny	2	3	6
procedure to add a new user and give authorisation to access the machine	1	3	3
console login	2	3	6
Umask	2	3	6
CMASK	2	2	4
Path variable	2	2	4
cron.deny at.deny	1	2	2
TCP wrapper installed and configured	3	3	9
SSH version	3	3	9
SSH protocol	2	3	6
log of SSH events	1	2	2
SSH server key length	3	3	9
Root login	1	3	3
empty passwd	3	3	9
No trusted hosts	1	3	3
RSA or passwd authentication	1	1	1

400 /etc/sshd_config	1	2	2
Log policy exist	2	2	4
log policy is comprehensive	2	2	4
log policy is enforced	3	3	9
log tools installed and configured	3	3	9
Integrity check software installed and well configured	3	3	9
Patching policy exist	2	2	4
Patching policy enforced	3	3	9
Secure state after applying patches	1	2	2
Backup and recovery policy exist	2	2	4
Backup and recovery policy enforced	3	3	9

## 12 APPENDIX D: results of # pkginfo -I

```

system      FJSVhea      SunOS Header Files
system      FJSVmdb      Fujitsu Platform Modular Debugger
system      FJSVmdbx     Fujitsu Platform Modular Debugger (64-bit)
system      FJSVvplr     Fujitsu platform links
system      FJSVvplu     Fujitsu usr/platform links
application IBMHACN     HTTP Server Admin Messages (Simplified Chinese)
application IBMHADE HTTP Server Admin Messages (German)
application IBMHAENU HTTP Server Admin Messages - U.S. English
application IBMHAES  HTTP Server Admin Messages (Spanish)
application IBMHAFR  HTTP Server Admin Messages (French)
application IBMHAIT  HTTP Server Admin Messages (Italian)
application IBMHAJP  HTTP Server Admin Messages (Japanese)
application IBMHAKO  HTTP Server Admin Messages (Korean)
application IBMHAPT  HTTP Server Admin Messages (Portuguese)
application IBMHATW  HTTP Server Admin Messages (Traditional Chinese)
application IBMHL128 HTTP Server LDAP Module (Domestic SSL)
application IBMHMENU HTTP Server Manual Pages (English)
application IBMHS128 HTTP Server SSL Module (128-bit Encryption)
application IBMHSCN  HTTP Server Documentation (Simplified Chinese)
application IBMHSDE  HTTP Server Documentation (German)
application IBMHSENU HTTP Server Documentation - U.S. English
application IBMHSSES HTTP Server Documentation (Spanish)
application IBMHSFCG HTTP Server Fast-CGI
application IBMHSFR  HTTP Server Documentation (French)
application IBMHSIT  HTTP Server Documentation (Italian)
application IBMHSJP  HTTP Server Documentation (Japanese)
application IBMHSKO  HTTP Server Documentation (Korean)
application IBMHSLDP HTTP Server LDAP Module (Export SSL)
application IBMHSMT  HTTP Server MT Module
application IBMHSPT  HTTP Server Documentation (Portuguese)
application IBMHSSCN HTTP Server SSL Messages (Simplified Chinese)
application IBMHSSDE HTTP Server SSL Messages (German)
application IBMHSSEN HTTP Server SSL Messages - U.S. English
application IBMHSSES HTTP Server SSL Messages (Spanish)
application IBMHSSFR HTTP Server SSL Messages (French)
application IBMHSSIT HTTP Server SSL Messages (Italian)
application IBMHSSJP HTTP Server SSL Messages (Japanese)
application IBMHSSKO HTTP Server SSL Messages (Korean)
application IBMHSSNM HTTP Server SNMP Module
application IBMHSSPT HTTP Server SSL Messages (Portuguese)
application IBMHSSRC HTTP Server Source Code
application IBMHSSSB HTTP Server SSL Module Common

```

application	IBMHSSTW	HTTP Server SSL Messages (Traditional Chinese)
application	IBMHSSTW	HTTP Server Documentation (Traditional Chinese)
application	IBMHTTTPA	HTTP Server Administration (Run-time)
application	IBMHTTTPD	HTTP Server Base Run-Time
application	NSCPcom	Netscape Communicator
application	RTCdoc	RTC Hard und Software Dokumentationsscripts
application	SMCgcc	gcc
application	SMCgzip	gzip
application	SMClsof	lsof
application	SMCmd5	md5
application	SMCperl	perl
application	SMCsudo	sudo
application	SMCtop	top
system	SMEvplr	SME platform links
system	SMEvplu	SME usr/platform links
system	SUNW1251f	Russian 1251 fonts
system	SUNW1394h	Sun IEEE1394 Framework Header Files
system	SUNW1394x	Sun IEEE1394 Framework (64-bit)
ALE	SUNW5ttf	Traditional Chinese BIG5 True Type Fonts Package
ALE	SUNW5xmft	Chinese/Taiwan BIG5 X Windows Platform minimum required Fonts Package
system	SUNWaccr	System Accounting, (Root)
system	SUNWaccu	System Accounting, (Usr)
system	SUNWadmap	System administration applications
system	SUNWadmc	System administration core libraries
system	SUNWadmfw	System & Network Administration Framework
system	SUNWadmj	Admin/Install Java Extension Libraries
system	SUNWadmr	System & Network Administration Root
system	SUNWafb	Elite3D Graphics System Software/Device Driver
system	SUNWafbcf	Elite3D Graphics Configuration Software
system	SUNWafbr	Elite3D Graphics System Software (Root)
application	SUNWafbw	Elite3D Graphics Window System Support
system	SUNWafbx	Elite3D Graphics System Software/Device Driver (64-bit)
system	SUNWami	Authentication Management Infrastructure
system	SUNWamir	Configuration files for Authentication Management Infrastructure
system	SUNWamix	Authentication Management Infrastructure (64 bit)
system	SUNWapchd	Apache Web Server Documentation
system	SUNWapchr	Apache Web Server (root)
system	SUNWapchu	Apache Web Server (usr)
system	SUNWapct	ABI Application Certification Tools
system	SUNWapppr	PPP/IP Asynchronous PPP daemon configuration files
system	SUNWapppu	PPP/IP Asynchronous PPP daemon and PPP login service
system	SUNWarc	Archive Libraries
system	SUNWarcx	Archive Libraries (64-bit)
system	SUNWarrf	X11 Arabic required fonts
system	SUNWast	Automated Security Enhancement Tools
system	SUNWatfsr	AutoFS, (Root)
system	SUNWatfsu	AutoFS, (Usr)
system	SUNWauda	Audio Applications
system	SUNWaudd	Audio Drivers
system	SUNWauddx	Audio Drivers (64-bit)
system	SUNWaudh	Audio Header Files
system	SUNWaudio	Audio applications
system	SUNWaudmo	Audio demo programs
system	SUNWbash	GNU Bourne-Again shell (bash)
system	SUNWbcp	SunOS 4.x Binary Compatibility
system	SUNWbnur	Networking UUCP Utilities, (Root)
system	SUNWbnuu	Networking UUCP Utilities, (Usr)
system	SUNWbtool	CCS tools bundled with SunOS
system	SUNWbtoox	CCS libraries bundled with SunOS (64-bit)
system	SUNWbzip	The bzip compression utility
system	SUNWbzipx	The bzip compression library (64-bit)
system	SUNWcar	Core Architecture, (Root)
system	SUNWcarx	Core Architecture, (Root) (64-bit)
system	SUNWcea	Sun GigaSwift Ethernet Adapter Driver 32 bit adb Macros
system	SUNWceax	Sun GigaSwift Ethernet Adapter Driver 64 bit adb Macros
system	SUNWced	Sun GigaSwift Ethernet Adapter (32-bit Driver)

system	SUNWcedu	Sun GigaSwift Ethernet Adapter Driver Headers
system	SUNWcedx	Sun GigaSwift Ethernet Adapter (64-bit Driver)
system	SUNWcg6	GX (cg6) Device Driver
system	SUNWcg6h	GX (cg6) Header Files
system	SUNWcg6x	GX (cg6) Device Driver (64-bit)
ALE	SUNWciu8	Simplified Chinese (EUC) iconv modules for UTF-8
ALE	SUNWciu8x	Simplified Chinese (EUC) iconv modules for UTF-8 (64-bit)
system	SUNWcpc	CPU Performance Counter driver
system	SUNWcpcu	CPU Performance Counter libraries and utilities
system	SUNWcpcux	CPU Performance Counter libraries and utilities (64-bit)
system	SUNWcpcx	CPU Performance Counter driver (64-bit)
system	SUNWcpr	Suspend, Resume package
system	SUNWcprx	Suspend, Resume package (64-bit)
system	SUNWcsd	Core Solaris Devices
system	SUNWcsl	Core Solaris, (Shared Libs)
system	SUNWcslx	Core Solaris Libraries (64-bit)
system	SUNWcsr	Core Solaris, (Root)
system	SUNWcstl	Appttrace Utility
system	SUNWcstlx	Appttrace Utility (64 bit)
system	SUNWcsu	Core Solaris, (Usr)
system	SUNWcsxu	Core Solaris (Usr) (64-bit)
system	SUNWcti2x	Netra ct I2C and System Drivers (64-bit)
system	SUNWctlu	Print utilities for CTL locales
CTL	SUNWctpls	Portable layout services for Complex Text Layout support
CTL	SUNWctplx	Portable layout services for CTL (64-bit)
ALE	SUNWcttf	Simplified Chinese (EUC) True Type Fonts
system	SUNWctu	Netra ct usr/platform links (64-bit)
system	SUNWcvc	Network Console
system	SUNWcvcr	Network Console daemon and rc script
system	SUNWcvcx	Network Console (64-bit)
ALE	SUNWcxmft	Simplified Chinese (EUC) X Windows Platform minimum Required Fonts
system	SUNWdcInt	Solaris Diskless Client Management Application
system	SUNWdcsr	Domain Configuration Server, (Root)
system	SUNWdcsu	Domain Configuration Server
system	SUNWdfb	Dumb Frame Buffer Device Drivers
system	SUNWdfbh	Dumb Frame Buffer Header Files
system	SUNWdhcm	DHCP Manager
system	SUNWdhcsb	Binary File Format Data Module for BOOTP/DHCP Services
system	SUNWdhcsr	BOOTP/DHCP Server Services, (Root)
system	SUNWdhcsu	BOOTP/DHCP Server Services, (Usr)
system	SUNWdial	Buttons/Dials Streams Module
application	SUNWdialh	Buttons/Dials Header Files
system	SUNWdialx	Buttons/Dials Streams Module (64-bit)
system	SUNWdoc	Documentation Tools
system	SUNWdpl	Developer Profiled Libraries
system	SUNWdplx	Developer Profiled Libraries (64-bit)
system	SUNWdrcrx	Dynamic Reconfiguration Modules for Sun Fire 15000 (64-bit)
system	SUNWdrr	Dynamic Reconfiguration Modules for Sun Enterprise 10000
system	SUNWdrrx	Dynamic Reconfiguration Modules for Sun Enterprise 10000 (64-bit)
application	SUNWdsab	Solstice DiskSuite 4.2.1 Collection
system	SUNWdtab	CDE DTBUILDER
system	SUNWdtbas	CDE application basic runtime environment
system	SUNWdtbax	CDE application basic runtime environment (64-bit)
system	SUNWdtcor	Solaris Desktop /usr/dt filesystem anchor
system	SUNWdtct	UTF-8 Code Conversion Tool
system	SUNWdtdem	CDE DEMOS
system	SUNWdtdmn	CDE daemons
system	SUNWdtdst	CDE Desktop Applications
system	SUNWdtdte	Solaris Desktop Login Environment
system	SUNWdtez	Solaris Desktop Extensions Applications
system	SUNWdthe	CDE HELP RUNTIME
system	SUNWdthed	CDE HELP DEVELOPER ENVIRONMENT
system	SUNWdthev	CDE HELP VOLUMES
system	SUNWdthez	Desktop Power Pack Help Volumes
system	SUNWdticn	CDE icons
system	SUNWdtim	Solaris CDE Image Viewer



system	SUNWdtinc	CDE Includes
system	SUNWdtjxt	Java Extensions
system	SUNWdtlog	System boot for Desktop Login
system	SUNWdtma	CDE man pages
system	SUNWdtmad	CDE developer man pages
system	SUNWdtmaz	Desktop Power Pack man pages
system	SUNWdtnc	Netscape Componentization Support for CDE
system	SUNWdtme	CDE Release Documentation
system	SUNWdtscm	CDE Dtpower Schemes (Root)
system	SUNWdtwm	CDE DESKTOP WINDOW MANAGER
system	SUNWebnfs	WebNFS
system	SUNWefclx	Embedded FCode Libraries (64-bit)
system	SUNWefcr	Embedded FCode Interpreter (Root)
system	SUNWefcux	Embedded FCode Interpreter (64-bit)
system	SUNWefcx	Embedded FCode Interpreter Drivers (64-bit)
system	SUNWensqr	Ensoniq ES1370/1371/1373 Audio Device Driver (32-bit), (Root)
system	SUNWensqx	Ensoniq ES1370/1371/1373 Audio Device Driver (64-bit), (Root)
system	SUNWereg	Solaris User Registration Installation ID file
system	SUNWeridx	Sun RIO 10/100 Mb Ethernet Drivers (64-bit)
system	SUNWesu	Extended System Utilities
system	SUNWesxu	Extended System Utilities (64-bit)
system	SUNWeudba	UTF-8 L10N for CDE Base
system	SUNWeudbd	UTF-8 L10N for CDE Dtbuilder
system	SUNWeudda	UTF-8 L10N For CDE Desktop Applications
system	SUNWeudhr	UTF-8 L10N For CDE Help Runtime
system	SUNWeudhs	UTF-8 L10N For CDE Help Runtime
system	SUNWeudis	UTF-8 L10N For CDE Icons
system	SUNWeudiv	UTF-8 L10N For Desktop Imagetool
system	SUNWeudlg	UTF-8 L10N For CDE Desktop Login
system	SUNWeudmg	UTF-8 L10N For Desktop Window Manager
system	SUNWeuez	English UTF-8 L10N For Desktop Power Pack Applications
system	SUNWeugrf	X11 sun_eu_greek fonts
system	SUNWeuluf	UTF-8 L10N For Language Environment User Files
system	SUNWeulux	UTF-8 L10N For Language Environment User Files (64-bit)
system	SUNWeuodf	UTF-8 Core OPENLOOK Desktop Files
system	SUNWeusru	English UTF-8 L10N For Solaris User Registration
system	SUNWeuxwe	UTF-8 X Window Environment
system	SUNWfac	Framed Access Command Environment
application	SUNWfbc	Frame Buffer Configuration Utility
system	SUNWfcip	Sun FCIP IP/ARP over FibreChannel Device Driver
system	SUNWfcipx	Sun FCIP IP/ARP over FibreChannel Device Driver (64 bit)
system	SUNWfcp	Sun FCP SCSI Device Driver
system	SUNWfcpx	Sun FCP SCSI Device Driver (64-bit)
system	SUNWfctl	Sun Fibre Channel Transport layer
system	SUNWfctlx	Sun Fibre Channel Transport layer (64-bit)
system	SUNWfdl	Font Downloader
system	SUNWffb	Creator Graphics System Software/Device Driver
application	SUNWffbcf	Creator Graphics Configuration Software
application	SUNWffbw	Creator Graphics Window System Support
system	SUNWffbx	Creator Graphics System Software/Device Driver (64-bit)
system	SUNWfns	Federated Naming System
system	SUNWfnx	Federated Naming System (64-bit)
system	SUNWfnx5	FNS Support For X.500 Directory Context
system	SUNWfnx5x	FNS Support For X.500 Directory Context (64-bit)
system	SUNWfruid	FRU ID Utility and Library (Usr)
system	SUNWfruip	FRU ID Platform Modules (Usr)
system	SUNWfrui	FRU ID Library (64-bit)
system	SUNWftdur	ftSafe developer utilities package (Root)
system	SUNWftduu	ftSafe developer utilities package (Usr)
system	SUNWftdux	ftSafe developer utilities package (Root) (64-bit)
system	SUNWftpr	FTP Server, (Root)
system	SUNWftpu	FTP Server, (Usr)
system	SUNWfwdcd	IEEE 1394 Video Conferencing Demo (64-bit)
system	SUNWfwdcu	IEEE 1394 Video Conferencing Support, (Usr) (64-bit)
system	SUNWfwdcx	IEEE 1394 Video Conferencing Class Driver (64-bit)
system	SUNWged	Sun Gigabit Ethernet Adapter Driver

system	SUNWgedm	Sun Gigabit Ethernet Adapter Driver Man Pages
system	SUNWgedu	Sun Gigabit Ethernet Adapter Driver Headers
system	SUNWglmr	rasctrl environment monitoring driver for i2c, (Root) (32-bit)
system	SUNWglmx	rasctrl environment monitoring driver for i2c (Root) (64-bit)
system	SUNWglrt	Layout Table Generation Utility
system	SUNWgpcu	The GNU Patch utility
system	SUNWgssdhx	GSS Diffie-Hellman (64-bit)
system	SUNWgss	GSSAPI V2
system	SUNWgssc	GSSAPI CONFIG V2
system	SUNWgssdh	GSS Diffie-Hellman
system	SUNWgssk	kernel GSSAPI V2
system	SUNWgsskx	kernel GSSAPI V2 (64-bit)
system	SUNWgssx	GSSAPI V2 (64-bit)
system	SUNWgzip	The GNU Zip (gzip) compression utility
system	SUNWhea	SunOS Header Files
ALE	SUNWhiu8	Traditional Chinese iconv modules for UTF-8
ALE	SUNWhiu8x	Traditional Chinese (EUC) iconv modules for UTF-8 (64-bit)
system	SUNWhmd	SunSwift SBus Adapter Drivers
system	SUNWhmdu	SunSwift SBus Adapter Headers
system	SUNWhmdx	SunSwift SBus Adapter Drivers (64-bit)
system	SUNWi13rf	X11 ISO-8859-13 required fonts
system	SUNWi15cs	X11 ISO8859-15 Codeset Support
system	SUNWi15rf	X11 ISO-8859-15 required fonts
system	SUNWi1cs	X11 ISO8859-1 Codeset Support
system	SUNWilof	ISO-8859-1 (Latin-1) Optional Fonts
system	SUNWi2cr	Device drivers for I2C devices, (Root, 32-bit)
system	SUNWi2cx	Device drivers for I2C devices, (Root, 64-bit)
system	SUNWi2rf	X11 ISO-8859-2 required fonts
system	SUNWi4rf	X11 ISO-8859-4 required fonts
system	SUNWi5rf	X11 ISO-8859-5 required fonts
system	SUNWi7rf	X11 ISO-8859-7 required fonts
system	SUNWi8rf	X11 iso8859-8 required fonts
system	SUNWi9rf	X11 ISO-8859-9 required fonts
system	SUNWidecr	IDE device drivers
system	SUNWidecx	IDE device drivers- 64bit
system	SUNWider	IDE Device Driver, (Root)
system	SUNWidn	Inter-Domain Network Modules for Sun Enterprise 10000
system	SUNWidnx	Inter-Domain Network Modules for Sun Enterprise 10000 (64-bit)
system	SUNWifb	Sun Expert3D (IFB) Graphics System Software/Device Driver
application	SUNWifbcf	Sun Expert3D (IFB) Graphics Configuration Software
system	SUNWifbr	Sun Expert3D (IFB) Graphics System Software (Root)
application	SUNWifbw	Sun Expert3D (IFB) Graphics Window System Support
system	SUNWifbx	Sun Expert3D (IFB) Graphics System Software/Device Driver (64-bit)
system	SUNWifp	Sun Fibre Channel Arbitrated Loop Device Driver
system	SUNWifph	Sun Fibre Channel Arbitrated Loop Driver Header Files
system	SUNWifpx	Sun Fibre Channel Arbitrated Loop Device Driver (64-bit)
system	SUNWigr	IGS CyberPro2010 Device Driver (ROOT)
application	SUNWigsu	IGS CyberPro2010 DDX (OW) Driver and Utilities
system	SUNWigsx	IGS CyberPro2010 64-bit Device Driver (ROOT)
system	SUNWinst	Install Software
system	SUNWipc	Interprocess Communications
system	SUNWipcx	Interprocess Communications (64-bit)
system	SUNWiscr	Sun ISCRI OCF CT Driver
system	SUNWiscrx	Sun ISCRI OCF CT Driver (64-bit)
system	SUNWislcc	XSH4 conversion for Eastern European locales
system	SUNWislcx	64-bit iconv conversion for Eastern European locales
system	SUNWisolc	XSH4 conversion for ISO Latin character sets
system	SUNWisolx	64-bit iconv conversion for ISO Latin character sets
system	SUNWj2dem	JDK 1.2 demo programs
system	SUNWj2dev	JDK 1.2 development tools
system	SUNWj2man	JDK 1.2 man pages
application	SUNWj2pi	Java Plug-in
system	SUNWj2rt	JDK 1.2 run time environment
system	SUNWj3dev	JDK 1.3 development tools
system	SUNWj3dmo	JDK 1.3 demo programs
system	SUNWj3man	JDK 1.3 man pages

system	SUNWj3rt	JDK 1.3 run time environment
Application	SUNWjass	JASS Toolkit 0.3.1
system	SUNWjcom	Java Communications API
system	SUNWjcomx	Java Communications API (64-bit)
system	SUNWjib	iButton OCF CT Driver
system	SUNWjiu8	Japanese iconv modules for UTF-8
system	SUNWjiu8x	Japanese iconv modules for UTF-8 (64-bit)
system	SUNWjmfpl	Java Media Framework Player
system	SUNWjsnmp	Java SNMP API
system	SUNWjvdm	JavaVM demo programs
system	SUNWjvdev	JavaVM developers package, includes javac, javah, and javap
system	SUNWjvjit	Java JIT compiler
system	SUNWjvman	JavaVM man pages
system	SUNWjvrt	JavaVM run time environment
system	SUNWjxcft	Japanese X Window System common fonts
system	SUNWjxmft	Japanese X Window System Minimum Required Fonts
application	SUNWkcspl	KCMS Optional Profiles
application	SUNWkcspg	KCMS Programmers Environment
application	SUNWkcspx	KCMS Programmers Environment (64-bit)
application	SUNWkcsrt	KCMS Runtime Environment
application	SUNWkcsrx	KCMS 64 bit Runtime Environment
system	SUNWkey	Keyboard configuration tables
ALE	SUNWkiu8	Korean UTF-8 iconv modules for UTF-8
ALE	SUNWkiu8x	Korean (UTF-8) iconv modules for UTF-8 (64-bit)
system	SUNWkmp2r	PS/2 Keyboard and Mouse Device Drivers, (Root, 32-bit)
system	SUNWkmp2x	PS/2 Keyboard and Mouse Device Drivers, (Root, 64-bit)
ALE	SUNWkttf	Korean True Type Fonts
system	SUNWkvmm	Core Architecture, (Kvm)
system	SUNWkvmx	Core Architecture (Kvm) (64-bit)
ALE	SUNWkxmft	Korean UTF-8 X Windows Platform minimum Required Fonts
system	SUNWlccom	Localization common files
system	SUNWlcl	Locale Conversion Library
system	SUNWlclx	Locale Conversion Library (64-bit)
system	SUNWless	The GNU pager (less)
system	SUNWlibC	Sun Workshop Compilers Bundled libc
system	SUNWlibCf	SunSoft WorkShop Bundled libc (cfront version)
system	SUNWlibCx	Sun WorkShop Bundled 64-bit libc
system	SUNWlibm	Sun WorkShop Bundled libm
system	SUNWlibms	Sun WorkShop Bundled shared libm
system	SUNWllc	LLC2 driver and its initialization programs
system	SUNWllcr	LLC2 driver configuration and startup files
system	SUNWllcx	LLC2 64bit driver
system	SUNWlldap	LDAP Libraries
system	SUNWlmsx	Sun WorkShop Bundled 64-bit shared libm
system	SUNWlmx	Sun WorkShop Bundled misc. 64-bit libm files
system	SUNWloc	System Localization
system	SUNWlocx	System Localization (64-bit)
system	SUNWlpmsg	LP Alerts
system	SUNWluxd	Sun Enterprise Network Array sf Device Driver
system	SUNWluxdx	Sun Enterprise Network Array sf Device Driver (64-bit)
system	SUNWluxl	Sun Enterprise Network Array socall Device Driver
system	SUNWluxlx	Sun Enterprise Network Array socall Device Driver (64-bit)
system	SUNWluxop	Sun Enterprise Network Array firmware and utilities
system	SUNWluxox	Sun Enterprise Network Array libraries (64-bit)
system	SUNWlvma	Solaris Volume Management API's
system	SUNWlvmg	Solaris Volume Management Application
system	SUNWlvmr	Solaris Volume Management (root)
system	SUNWm64	M64 Graphics System Software/Device Driver
application	SUNWm64cf	M64 Graphics Configuration Software
application	SUNWm64w	M64 Graphics Window System Support
system	SUNWm64x	M64 Graphics System Software/Device Driver (64-bit)
system	SUNWm64xr	M64XR System Software (Device Driver Config.)
system	SUNWman	On-Line Manual Pages
application	SUNWmc	Solaris Management Console 2.0 (Server Components)
application	SUNWmcc	Solaris Management Console 2.0 (Client Components)
application	SUNWmccom	Solaris Management Console 2.0 (Common Components)

application	SUNWmcdev	Solaris Management Console 2.0 (Development Kit)
application	SUNWmcex	Solaris Management Console 2.0 (Examples)
system	SUNWmdb	Modular Debugger
system	SUNWmdbdm	Modular Debugger Demo Source
system	SUNWmdbx	Modular Debugger (64-bit)
system	SUNWmdg	Solstice DiskSuite Tool
system	SUNWmdi	Sun Multipath I/O Drivers
system	SUNWmdix	Sun Multipath I/O Drivers (64-bit)
system	SUNWmdnr	Solstice DiskSuite Log Daemon Configuration Files
system	SUNWmdnu	Solstice DiskSuite Log Daemon
system	SUNWmdr	Solstice DiskSuite Drivers
system	SUNWmdu	Solstice DiskSuite Commands
system	SUNWmdx	Solstice DiskSuite Drivers (64-bit)
system	SUNWmfdev	Motif UIL Compiler
system	SUNWmfman	CDE Motif Manuals
system	SUNWmfrun	Motif RunTime Kit
system	SUNWmga	Solaris Management Applications
system	SUNWmgapp	Solaris Management Applications
system	SUNWmibii	Solstice Enterprise Agents 1.0.3 SNMP daemon
system	SUNWmipr	Mobile-IP (Root)
system	SUNWmipu	Mobile-IP (Usr)
system	SUNWmkcd	CD creation utilities
system	SUNWmp	MP Print Filter
system	SUNWnamdt	Northern America CDE Support
system	SUNWnamos	Northern America OS Support
system	SUNWnamow	Northern America OW Support
system	SUNWnamox	Northern America 64-bit OS Support
system	SUNWncar	Solaris Network Cache and Accelerator (Root)
system	SUNWncarx	Solaris Network Cache and Accelerator (Root) (64-bit)
system	SUNWncau	Solaris Network Cache and Accelerator (Usr)
system	SUNWncaux	Solaris Network Cache and Accelerator (Usr) (64-bit)
system	SUNWnlsr	Network Information System, (Root)
system	SUNWnlsu	Network Information System, (Usr)
system	SUNWntpr	NTP, (Root)
system	SUNWntpu	NTP, (Usr)
system	SUNWocf	Open Card Framework
system	SUNWocfh	Open Card Framework header files
system	SUNWocfr	Configuration files for Open Card Framework
system	SUNWocfx	Open Card Framework (64 bit)
system	SUNWoladd	OPEN LOOK Alternate Desktop Demos
system	SUNWolaud	OPEN LOOK Audio applications
system	SUNWolbk	OpenWindows online handbooks
system	SUNWoldcv	OPEN LOOK document and help viewer applications
system	SUNWoldem	OPEN LOOK demo programs
system	SUNWoldim	OPEN LOOK demo images
system	SUNWoldst	OPEN LOOK deskset tools
system	SUNWoldte	OPEN LOOK Desktop Environment
system	SUNWolimt	OPEN LOOK imagetool
system	SUNWolinc	OPEN LOOK include files
system	SUNWolman	OPEN LOOK toolkit/desktop users man pages
system	SUNWolrte	OPEN LOOK toolkits runtime environment
system	SUNWolslb	OPEN LOOK toolkit/desktop static/lint libraries
system	SUNWolsrc	OPEN LOOK sample source
system	SUNWosdem	OS demo source
system	SUNWowbcp	OpenWindows binary compatibility
system	SUNWpamsc	PAM Smart Card module
system	SUNWpamsx	PAM Smart Card module (64-bit)
system	SUNWpcelx	3COM EtherLink III PCMCIA Ethernet Driver
system	SUNWpcmci	PCMCIA Card Services, (Root)
system	SUNWpcmdu	PCMCIA Card Services, (Usr)
system	SUNWpcmcx	PCMCIA Card Services (64-bit)
system	SUNWpcmem	PCMCIA memory card driver
system	SUNWpcr	SunSoft Print - Client, (root)
system	SUNWpcser	PCMCIA serial card driver
system	SUNWpcu	SunSoft Print - Client, (usr)
system	SUNWpd	PCI Drivers

system	SUNWpdas	PDA Synchronization for Solaris
system	SUNWpdu	PCI Drivers Headers
system	SUNWpdx	PCI Drivers (64-bit)
system	SUNWpiclh	PICL Header Files (Usr)
system	SUNWpiclr	PICL Framework (Root)
system	SUNWpiclu	PICL Libraries, and Plugin Modules (Usr)
system	SUNWpiclx	PICL Libraries (64-bit)
system	SUNWpl5m	Perl5 On-Line Manual Pages
system	SUNWpl5p	Perl 5.005_03 (POD Documentation)
system	SUNWpl5u	Perl 5.005_03
system	SUNWplow	OpenWindows enabling for Partial Locales
system	SUNWplow1	OpenWindows enabling for Supplementary Partial Locales
system	SUNWpmowm	Power Management OW Utilities Man Pages
system	SUNWpmowr	Power Management OW Utilities, (Root)
system	SUNWpmowu	Power Management OW Utilities, (Usr)
system	SUNWpmr	Power Management config file and rc script
system	SUNWpmu	Power Management binaries
system	SUNWpmux	Power Management binaries (64-bit)
system	SUNWppm	Solaris Print Manager
system	SUNWpppd	Solaris PPP Device Drivers
system	SUNWpppdr	Solaris PPP configuration files
system	SUNWpppdu	Solaris PPP daemon and utilities
system	SUNWpppdx	Solaris PPP Device Drivers (64-bit)
system	SUNWpppg	GNU utilities for PPP
system	SUNWpppk	PPP/IP and IPdialup Device Drivers
system	SUNWpppkx	PPP/IP and IPdialup Device Drivers (64-bit)
system	SUNWpsdpr	PCMCIA ATA card driver
system	SUNWpsf	PostScript filters - (Usr)
system	SUNWpsr	SunSoft Print - LP Server, (root)
system	SUNWpstl	Appttrace Utility
system	SUNWpstlx	Appttrace Utility (64 bit)
system	SUNWpsu	SunSoft Print - LP Server, (usr)
system	SUNWqfed	Sun Quad FastEthernet Adapter Driver
system	SUNWqfedu	Sun Quad FastEthernet Adapter Driver Headers
system	SUNWqfedx	Sun Quad FastEthernet Adapter Driver (64-bit)
system	SUNWqlc	Qlogic ISP 2200/2202 Fibre Channel Device Driver
system	SUNWqlcx	Qlogic ISP 2200/2202 Fibre Channel Device Driver (64 bit)
system	SUNWrdr	On-Line Open Issues ReadMe
system	SUNWrmodu	Realmode Modules, (Usr)
system	SUNWrpm	Utilities for processing RPM archives
system	SUNWrsg	RPCSEC_GSS
system	SUNWrsgk	kernel RPCSEC_GSS
system	SUNWrsgx	RPCSEC_GSS (64-bit)
system	SUNWrtvc	SunVideo Device Driver
application	SUNWrtvc1	SunVideo XIL library support
application	SUNWrtvcu	SunVideo Runtime Support Software
system	SUNWrtvcx	SunVideo Device Driver (64-bit)
system	SUNWsacom	Solstice Enterprise Agents 1.0.3 files for root file system
system	SUNWsadmi	Solstice Enterprise Agents 1.0.3 Desktop Management Interface
system	SUNWsadml	Solstice Launcher.
system	SUNWsadm (64-bit)	Solstice Enterprise Agents 1.0.3 Desktop Management Interface Libraries
system	SUNWsasnm	Solstice Enterprise Agents 1.0.3 Simple Network Management Protocol
system	SUNWsasnx	Solstice Enterprise Agents 1.0.3 Simple Network Management Protocol Libraries (64-bit)
application	SUNWsbuc	Solstice Backup (Backup/Recover) Client
application	SUNWsbum	Solstice Backup (Backup/Recover) Man
system	SUNWschbp	SPARCCompilers Binary Compatibility Libraries
system	SUNWscgui	Solaris Smart Card Administration GUI
system	SUNWscmr	Init script & links for Sun Fire 15000 Key Management daemon
system	SUNWscmu	Key Management daemon for Sun Fire 15000
system	SUNWscmx	Key Management Modules for Sun Fire 15000 (64-Bit)
system	SUNWscmos	SCM Microsystems SmartOS
system	SUNWscmsc	Sun SCRI OCF CT Driver
system	SUNWscplp	SunSoft Print - Source Compatibility, (Usr)
system	SUNWscpr	Source Compatibility, (Root)

system	SUNWscpu	Source Compatibility, (Usr)
system	SUNWscpx	Source Compatibility (Usr) (64-bit)
system	SUNWses	SCSI Enclosure Services Device Driver
system	SUNWsex	SCSI Enclosure Services Device Driver (64-bit)
system	SUNWsfdr	Sun Fire 880 DR Daemon
system	SUNWsfdr	Sun Fire 880 DR Daemon init script
system	SUNWsior	SuperIO 307 (plug-n-play) device drivers, (Root) (32-bit)
system	SUNWsiox	SuperIO 307 (plug-n-play) device drivers, (Root) (64-bit)
system	SUNWslpr	SLP, (Root)
system	SUNWslpu	SLP, (Usr)
system	SUNWslpx	SLP (64-bit)
system	SUNWsndmr	Sendmail root
system	SUNWsndmu	Sendmail user
system	SUNWsolnm	Solaris Naming Enabler
system	SUNWspl	Spell Checking Engine - Base Release (English)
system	SUNWsprot	Solaris Bundled tools
system	SUNWsprox	Sun WorkShop Bundled 64-bit make library
system	SUNWsra	Source Compatibility Archive Libraries
system	SUNWsregu	Solaris User Registration
system	SUNWsrh	Source Compatibility Header Files
system	SUNWssad	SPARCstorage Array Drivers
system	SUNWssadx	SPARCstorage Array Drivers (64-bit)
system	SUNWssaop	SPARCstorage Array Utility
system	SUNWstcx	SUN ISCR Kernel Driver - (64-bit)
system	SUNWsutl	Static Utilities
system	SUNWswmt	Install and Patch Utilities
application	SUNWsx	SX/CG14 Shareable Library
application	SUNWsxow	SX/CG14 Window System Support
system	SUNWtcsh	Tenex C-shell (tcsh)
application	SUNWtcxow	TCX Window System Support
system	SUNWter	Terminal Information
system	SUNWtiu8	Thai UTF-8 iconv modules for UTF-8
system	SUNWtiu8x	Thai UTF-8 iconv modules for UTF-8 (64-bit)
system	SUNWtleu	Thai Locale Environment User Files
system	SUNWtleux	Thai Language Environment user files (64-bit)
system	SUNWtltk	ToolTalk runtime
system	SUNWtltkd	ToolTalk developer support
system	SUNWtltkm	ToolTalk manual pages
system	SUNWtltkx	ToolTalk library (64-bit)
system	SUNWtnfc	TNF Core Components
system	SUNWtnfcx	TNF Core Components (64-bit)
system	SUNWtnfd	TNF Developer Components
system	SUNWtoo	Programming Tools
system	SUNWtoox	Programming Tools (64-bit)
system	SUNWtxfnt	Thai X Windows Platform required Fonts Package
system	SUNWucbt	Apptrace support objects for ucblib
system	SUNWucbt	Apptrace support objects for ucblib (64 bit)
system	SUNWudf	Universal Disk Format 1.50, (Usr)
system	SUNWudfr	Universal Disk Format 1.50
system	SUNWudfrx	Universal Disk Format 1.50 (64-bit)
system	SUNWuiu8	Iconv modules for UTF-8 Locale
system	SUNWuiu8x	Iconv Modules for UTF-8 Locale (64-bit)
system	SUNWuium	Iconv Man Pages for UTF-8 Locale
system	SUNWulcf	UTF-8 Locale Environment Common Files
system	SUNWulcfx	UTF-8 Locale Environment Common Files (64-bit)
system	SUNWulocf	UTF-8 Locale Environment OpenWindows Common Files
system	SUNWusb	USB Device Drivers
system	SUNWusbu	USB Headers
system	SUNWusbx	USB Device Drivers (64-bit)
system	SUNWusoc	Sun Universal SOC+ Device Driver
system	SUNWusocx	Sun Universal SOC+ Device Driver (64-bit)
system	SUNWusx	UltraSPARC CPU Device Driver (64-bit)
system	SUNWuxfl1	SUNW,Ultra-1 FLASH PROM Update
system	SUNWuxfl2	SUNW,Ultra-2 FLASH PROM Update
system	SUNWuxfl4	SUNW,Ultra-4 FLASH PROM Update
system	SUNWuxfle	SUNW,Ultra-Enterprise FLASH PROM Update

system	SUNWuxflr	Sun4u FLASH PROM update generic components, (Root)
system	SUNWuxflu	Sun4u FLASH PROM Update generic components, (Usr)
system	SUNWuxlcf	UTF-8 X Locale Environment Common Files
system	SUNWuxlcx	UTF-8 X Locale Environment Common Files (64-bit)
system	SUNWvld	Sun Ethernet Vlan Utility Routines
system	SUNWvldu	Sun Ethernet Vlan Utility Headers
system	SUNWvldx	Sun Ethernet Vlan Utility Routines (64-bit)
system	SUNWvolg	Volume Management Graphical User Interface
system	SUNWvolr	Volume Management, (Root)
system	SUNWvolu	Volume Management, (Usr)
system	SUNWvolux	Volume Management (Usr) (64-bit)
system	SUNWwbapi	WBEM API
system	SUNWwbcor	WBEM Services (root)
system	SUNWwbcou	WBEM Services (usr)
application	SUNWwbdev	Sun WBEM SDK
application	SUNWwbdoc	Sun WBEM SDK - Documentation
application	SUNWwbmc	Solaris Management Console 2.0 (WBEM Components)
system	SUNWwsr2	Solaris Product Registry & Web Start runtime support
system	SUNWwsrv	Solaris Product Registry Viewer
system	SUNWxcu4	XCU4 Utilities
system	SUNWxcu4t	XCU4 make and scs utilities
system	SUNWxcu4x	XCU4 Utilities (64-bit)
system	SUNWxi18n	X Window System Internationalization Common Package
system	SUNWxi18x	X Window System Internationalization Common Package (64-bit)
application	SUNWxilcg	SX/CG14 XIL Support
application	SUNWxildh	XIL Loadable Pipeline Libraries
application	SUNWxilh	XIL API Header Files
application	SUNWxilow	XIL Deskset Loadable Pipeline Libraries
application	SUNWxilrl	XIL Runtime Environment
application	SUNWxilvl	VIS/XIL Support
system	SUNWxim	X Window System X Input Method Server Package
system	SUNWximx	X Window System X Input Method Server Package (64-bit)
system	SUNWxwacx	AccessX client program
system	SUNWxwcft	X Window System common (not required) fonts
system	SUNWxwcl	X Window System Display Postscript CID support library
system	SUNWxwdem	X Window System demo programs
system	SUNWxwdim	X Window System demo images
system	SUNWxwdv	X Windows System Window Drivers
system	SUNWxwdvx	X Windows System Window Drivers (64-bit)
system	SUNWxwdxm	DPS motif library
system	SUNWxwfa	X Window System Font Administrator
system	SUNWxwfnt	X Window System platform required fonts
system	SUNWxwfs	Font server
system	SUNWxwhl	X Window System & Graphics Header links in /usr/include
system	SUNWxwice	ICE components
system	SUNWxwicx	X Window System ICE library (64-bit)
system	SUNWxwinc	X Window System include files
system	SUNWxwkey	X Windows software, PC keytables
system	SUNWxwman	X Window System online user man pages
system	SUNWxwmod	OpenWindows kernel modules
system	SUNWxwmox	X Window System kernel modules (64-bit)
system	SUNWxwoft	X Window System optional fonts
system	SUNWxwopt	nonessential MIT core clients and server extensions
system	SUNWxwplt	X Window System platform software
system	SUNWxwplx	X Window System library software (64-bit)
system	SUNWxwpmn	X Window System online programmers man pages
system	SUNWxwpsr	Sun4u-platform specific X server auxiliary filter modules
system	SUNWxwrtl	X Window System & Graphics Runtime Library Links in /usr/lib
system	SUNWxwrtx	X Window System Runtime Compatibility Package (64-bit)
system	SUNWxwslb	X Window System static/lint libraries
system	SUNWxwslx	X Window System lint libraries (64-bit)
system	SUNWxwsrc	X Window System sample source
system	SUNWypr	NIS Server for Solaris (root)
system	SUNWypu	NIS Server for Solaris (usr)
system	SUNWzip	The Info-Zip (zip) compression utility
system	SUNWzlib	The Zip compression library



system	SUNWzlibx	The Info-Zip compression library (64-bit)
system	SUNWzsh	Z shell (zsh)
system	TSBWvplr	Toshiba platform links
system	TSBWvplu	Toshiba usr/platform links
system	TSIpgx	PGX32 (Raptor GFX) System Software/Device Driver
application	TSIpgxw	PGX32 (Raptor GFX) X Window System Support
system	TSIpgxx	PGX32 (Raptor GFX) System Software/Device Driver (64-bit)
system	TWSvplr	TWS platform links
system	TWSvplu	TWS usr/platform links
application	gsk4bas	gsk4bas
application	itj000001	IBMWebAS.base - WASicon
application	itj000002	IBMWebAS.base - server
application	itj000003	IBMWebAS.base - tivoli
application	itj000004	IBMWebAS.base - IBMApache
application	itj000005	IBMWebAS.base - ITJ Info

### 13 APPENDIX E: results of # ./PatchCheck

#### Missing Security Patches for Solaris8

108528-12 SunOS 5.8: kernel update patch

108773-13 \* SunOS 5.8: IIIM and X Input & Output Method patch

108869-12 SunOS 5.8: snmpdx/mibiisa/libssasnmp/snmplib patch

108875-10 SunOS 5.8: c2audit patch

108909-12 \* CDE 1.4: Smart Card Administration GUI patch

108949-07 CDE 1.4: libDtHelp/libDtSvc patch

108975-05 SunOS 5.8: /usr/bin/rmformat and /usr/sbin/format patch

108987-07 SunOS 5.8: Patch for patchadd and patchrm

108991-18 SunOS 5.8: /usr/lib/libc.so.1 patch

108993-05 SunOS 5.8: nss and ldap patch

109005-03 \* SunOS 5.8: /sbin/su.static and /usr/bin/su patch

109134-24 \* SunOS 5.8: WBEM patch

109149-02 \* SunOS 5.8: /usr/sbin/mkdevmaps and /usr/sbin/mkdevalloc patch

109154-10 \* SunOS 5.8: PGX32 Graphics

109202-03 \* SunOS 5.8: /kernel/misc/gld and /kernel/misc /sparcv9/gld patch

109234-07 \* SunOS 5.8: Apache and NCA patch

109238-02 SunOS 5.8: /usr/bin/sparcv7/ipcs and /usr/bin/sparcv9/ipcs patch

109279-18 SunOS 5.8: /kernel/drv/ip patch

109320-04 SunOS 5.8: LP patch

109322-09 SunOS 5.8: libnsl patch

109324-04 SunOS 5.8: sh/jsh/rsh/pfsh patch

109326-06 SunOS 5.8: libresolv.so.2 and in.named patch

109354-13 \* CDE 1.4: dtsession patch

109667-04 SunOS 5.8: /usr/lib/inet/xntpd and /usr/sbin/ntpdate patch

109695-03 \* SunOS 5.8: /etc/smartcard/opencard.pro perties patch

109805-04 SunOS 5.8: /usr/lib/security/pam\_krb5.so.1 patch

109887-10 \* SunOS 5.8: smartcard patch

109888-13 SunOS 5.8: platform drivers patch

109896-07 \* SunOS 5.8: USB and Audio Framework patch

109898-05 SunOS 5.8: /kernel/drv/arp patc h

109951-01 SunOS 5.8: jserver buffer overflow

110286-04 OpenWindows 3.6.2: Tooltalk patch

110416-03 \* SunOS 5.8: ATOK12 patch

110668-02 SunOS 5.8: /usr/sbin/in.telnetd patch

110898-03 SunOS 5.8: csh/pfcsh patch



110903-02 SunOS 5.8: edit, ex, ved it, vi and view patch  
 110957-02 SunOS 5.8: /usr/bin/mailx patch  
 111085-02 SunOS 5.8: /usr/bin/login patch  
 111332-04 \* SunOS 5.8: /usr/lib/dcs patch  
 111504-01 SunOS 5.8: /usr/bin/tip patch  
 111596-02 SunOS 5.8: /usr/lib/netsvc/yp/rpc.yppasswdd patch  
 111606-02 SunOS 5.8: /usr/sbin/in.ftpd patch  
 111626-01 OpenWindows 3.6.2: Xview Patch  
 111647-01 \* BCP libmle buffer overflow  
 111659-03 SunOS 5.8: passwd and pam\_unix.so.1 patch  
 111826-01 SunOS 5.8: /usr/sbin/sparcv7/whodo & /usr/sbin/sparcv9/whodo patch  
 111874-02 SunOS 5.8: usr/bin/mail patch  
 111881-01 SunOS 5.8: /usr/kernel/strmod/telmod patch  
 112039-01 \* SunOS 5.8: usr/bin/ckitem patch  
 112218-01 SunOS 5.8:: pam\_ldap.so.1 patch

#### Missing Recommended Patches for Solaris8

108434-04 32-Bit Shared library patch for C++  
 108435-04 64-Bit Shared library patch for C++  
 108528-12 SunOS 5.8: kernel update patch  
 108652-46 X11 6.4.1 Xsun patch  
 108725-06 SunOS 5.8: st driver patch  
 108727-09 SunOS 5.8: /kernel/fs/nfs and /kernel/fs/sparcv9/nfs patch  
 108827-12 SunOS 5.8: /usr/lib/libthread.so.1 patch  
 108869-12 SunOS 5.8: snmpdx/mibiisa/libssasnmplib patch  
 108875-10 SunOS 5.8: c2audit patch  
 108949-07 CDE 1.4: libDtHelp/libDtSvc patch  
 108974-17 SunOS 5.8: dada, uata, dad, sd and scsi drivers patch  
 108975-05 SunOS 5.8: /usr/bin/rmformat and /usr/sbin/format patch  
 108981-07 SunOS 5.8: /kernel/drv/hme and /kernel/drv/sparcv9/hme patch  
 108987-07 SunOS 5.8: Patch for patchadd and patchrm  
 108991-18 SunOS 5.8: /usr/lib/libc.so.1 patch  
 108993-05 SunOS 5.8: nss and ldap patch  
 109007-06 SunOS 5.8: at/atrm/batch/cron patch  
 109147-12 SunOS 5.8: Linker patch  
 109238-02 SunOS 5.8: /usr/bin/sparcv7/ipcs and /usr/bin/sparcv9/ipcs patch  
 109277-02 SunOS 5.8: /usr/bin/iostat patch  
 109279-18 SunOS 5.8: /kernel/drv/ip patch  
 109318-21 SunOS 5.8: suninstall patch  
 109320-04 SunOS 5.8: LP patch  
 109322-09 SunOS 5.8: libnsl patch  
 109324-04 SunOS 5.8: sh/jsh/rsh/pfsh patch  
 109326-06 SunOS 5.8: libresolv.so.2 and in.named patch  
 109657-06 SunOS 5.8: isp driver patch  
 109667-04 SunOS 5.8: /usr/lib/inet/xntpd and /usr/sbin/ntpdate patch  
 109805-04 SunOS 5.8: /usr/lib/security/pam\_krb5.so.1 patch  
 109882-05 SunOS 5.8: eri header files patch  
 109888-13 SunOS 5.8: platform drivers patch  
 109898-05 SunOS 5.8: /kernel/drv/arp patch  
 109904-05 SunOS 5.8: /etc/default/mpathd and /sbin/in.mpathd patch  
 109951-01 SunOS 5.8: jserver buffer overflow

```

110283-05    SunOS 5.8: mkfs and newfs patch
110286-04    OpenWindows 3.6.2: Tooltalk patch
110380-04    SunOS 5.8: ufssnapshots support, libadm patch
110460-13    SunOS 5.8: fruid/PICL plug -ins patch
110662-06    SunOS 5.8: ksh patch
110668-02    SunOS 5.8: /usr/sbin/in.telnetd patch
110723-04    SunOS 5.8: /kernel/drv/sparcv9/eri patch
110898-03    SunOS 5.8: csh/pfcsch patch
110903-02    SunOS 5.8: edit, ex, vedit, vi and view patch
110934-05    SunOS 5.8: pkgtrans, pkgadd, pkgchk and libpkg.a patch
110945-04    SunOS 5.8: /usr/sbin/syslogd patch
110951-02    SunOS 5.8: /usr/sbin/tar and /usr/sbin/static/tar patch
110957-02    SunOS 5.8: /usr/bin/mailx patch
111085-02    SunOS 5.8: /usr/bin/login patch
111177-06    SunOS 5.8: /usr/lib/lwp/libthread.so.1 patch
111293-04    SunOS 5.8: /usr/lib/libdevinfo.so.1 patch
111327-05    SunOS 5.8: libsocket patch
111504-01    SunOS 5.8: /usr/bin/tip patch
111596-02    SunOS 5.8: /usr/lib/netsvc/yp/rpc.yppasswdd patch
111606-02    SunOS 5.8: /usr/sbin/in.ftpd patch
111626-01    OpenWindows 3.6.2: Xview Patch
111659-03    SunOS 5.8: passwd and pam_unix.so.1 patch
111826-01    SunOS 5.8: /usr/sbin/sparcv7/whodo & /usr/sbin/sparcv9/whodo patch
111874-02    SunOS 5.8: usr/bin/mail patch
111881-01    SunOS 5.8: /usr/kernel/strmod/telmod patch
112138-01    SunOS 5.8:: usr/bin/domainname patch
112218-01    SunOS 5.8:: pam_ldap.so.1 patch

```

For more information see 'Solaris8.PatchReport'

## 14 APPENDIX F: /etc/system file

```

forceload: misc/md_stripe
forceload: misc/md_mirror
forceload: drv/pcipsy
forceload: drv/simba
forceload: drv/glm
forceload: drv/sd
rootdev:/pseudo/md@0:0,1,blk
set md:mddb_bootlist1="sd: 6:16 sd:6:1050 sd:7:16 sd:7:1050 sd:14:16"
set md:mddb_bootlist2="sd:14:1050 sd:15:16 sd:15:1050"
set sys:coredumpsize=0
set nfssrv:nfs_portmon=1
set noexec_user_stack_log=1
set noexec_user_stack=1

```

## 15 Appendix G: World-write files

```

>>>> 57 World writeable files....
-rw-rw-rw-  1 root    root          0 Oct 16 16:10 /var/sadm/install/.pkg.lock
-rw-rw-rw-  1 root    bin           0 Oct 16 16:13 /var/adm/spellhist
drwxrwxrwt  3 root    mail        512 Oct 25 17:48 /var/mail
drwxrwxrwt  2 root    bin        512 Oct 16 16:13 /var/preserve

```

drwxrwxrwt	7	root	bin	512	Oct	25	17:09	/var/spool/pkg
drwxrwx-wx	2	lp	lp	512	Oct	16	16:20	/var/spool/lp/fifos/public
-rw-rw-rw-	1	lp	lp	0	Oct	17	07:25	/var/spool/lp/fifos/FIFO
drwxrwxrwt	2	uucp	uucp	512	Oct	16	16:44	/var/spool/uucppublic
drwxrwxrwt	2	root	sys	512	Oct	26	14:55	/var/tmp
drwxrwxrwt	2	root	root	512	Oct	16	16:22	/var/dt/dtpower/schemes
-rw-rw-rw-	1	root	root	8	Sep	24	1999	/var/dt/dtpower/_current_scheme
drwxrwxrwt	2	root	root	512	Oct	17	07:25	/var/dt/tmp
-rw-rw-rw-	1	root	other	0	Oct	17	07:43	/var/nsr/tmp/product.res.lck
-rw-rw-rw-	1	root	other	0	Oct	17	07:43	/var/nsr/tmp/nsrla.res.lck
drwxrwxrwx	2	root	other	512	Oct	17	07:43	/var/nsr/applogs
drwxrwxrwx	3	root	other	1024	Oct	26	14:45	/opt/WebSphere/AppServer/properties
drwxrwxrwx	2	root	other	512	Oct	25	17:09	/opt/WebSphere/AppServer/deployableEJBs
-rwxrwxrwx	1	root	other	6715	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Access.jar
-rwxrwxrwx	1	root	other	6641	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Account.jar
-rwxrwxrwx	1	root	other	4265	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Audit.jar
-rwxrwxrwx	1	root	other	3942	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Hello.jar
-rwxrwxrwx	1	root	other	4781	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/History.jar
-rwxrwxrwx	1	root	other	3953	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Increment.jar
-rwxrwxrwx	1	root	other	5129	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Leave.jar
-rwxrwxrwx	1	root	other	5995	Oct	25	17:49	/opt/WebSphere/AppServer/deployableEJBs/Transfer.jar
drwxrwxrwx	2	root	other	512	Oct	26	15:23	/opt/WebSphere/AppServer/logs
----rw-rw-	1	root	sys	42849	May	24	18:26	/usr/sadm/lib/wbem/prodregapi.jar
-rw-rw-rw-	1	bin	bin	137	Jun	2	1999	/usr/dt/appconfig/jmf/jmf.properties
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/applet
drwxrwxrwx	7	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client/enumeration
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client/logging
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client/misc
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client/namespace
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/client/systeminfo
drwxrwxrwx	6	root	sys	512	Oct	16	16:52	/usr/demo/wbem/provider
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/provider/jni
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/provider/sfl
drwxrwxrwx	2	root	sys	512	Oct	16	16:52	/usr/demo/wbem/provider/sip
-rw--w--w-	1	bin	bin	0	Jan	6	2000	/usr/oasys/tmp/TERRLOG
-rw-rw-rw-	1	root	other	12673730	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/was35_adv_ptf_3.jar
-rw-rw-rw-	1	root	other	3891825	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/was35_adv_ptf_3.log
-rwxrwxrwx	1	25226	24641	5843	Feb	28	2001	/ctadmin/2001-10-25/3.5_fixpack_3/was35_adv_ptf_3.readme
-rw-rw-rw-	1	root	other	13080295	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/ihs_ptf_3.jar
-rw-rw-rw-	1	root	other	113848	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/ihs_ptf_3.log
-rw-rw-rw-	1	root	other	896078	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/jdk_ptf_3.jar
-rw-rw-rw-	1	root	other	9311	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/jdk_ptf_3.log
-rwxrwxrwx	1	25226	24641	7174	Mar	5	2001	/ctadmin/2001-10-25/3.5_fixpack_3/install.sh
-rw-rw-rw-	1	root	other	14304535	May	16	07:55	/ctadmin/2001-10-25/3.5_fixpack_4/was35_adv_ptf_4.jar
-rw-rw-rw-	1	root	other	4157558	May	16	07:55	/ctadmin/2001-10-25/3.5_fixpack_4/was35_adv_ptf_4.log
-rwxrwxrwx	1	25226	24641	5846	May	15	19:12	/ctadmin/2001-10-25/3.5_fixpack_4/was35_adv_ptf_4.readme
-rw-rw-rw-	1	root	other	11177446	May	16	07:57	/ctadmin/2001-10-25/3.5_fixpack_4/ihs_ptf_4.jar
-rw-rw-rw-	1	root	other	30563	May	16	07:57	/ctadmin/2001-10-25/3.5_fixpack_4/ihs_ptf_4.log
-rw-rw-rw-	1	root	other	21393039	May	16	08:00	/ctadmin/2001-10-25/3.5_fixpack_4/jdk_ptf_4.jar
-rw-rw-rw-	1	root	other	109435	May	16	08:00	/ctadmin/2001-10-25/3.5_fixpack_4/jdk_ptf_4.log
-rwxrwxrwx	1	25226	24641	7789	May	16	08:00	/ctadmin/2001-10-25/3.5_fixpack_4/install.sh

## 16 APPENDIX H: SUID/SGID files

```
-r-s--x--x 1 root bin 19620 Feb 26 2001 /usr/lib/lp/bin/netpr
-r-sr-xr-x 1 root bin 13840 Jan 6 2000 /usr/lib/fs/ufs/quota
-r-sr-xr-x 1 root bin 83008 Jan 24 2001 /usr/lib/fs/ufs/ufsdump
-r-sr-xr-x 1 root bin 907796 Dec 21 2000 /usr/lib/fs/ufs/ufsrestore
---s--x--x 1 root bin 4104 Jan 6 2000 /usr/lib/pt_chmod
-r-sr-xr-x 1 root bin 7068 Jan 6 2000 /usr/lib/utmp_update
-r-sr-xr-x 1 root bin 107408 Apr 6 2001
/usr/lib/fbconfig/SUNWifb_config
-r-sr-xr-x 1 root bin 752512 Nov 10 2000 /usr/lib/sendmail
-rwsr-xr-x 1 root adm 5040 Jan 6 2000 /usr/lib/acct/accton
---s--x--x 1 uucp uucp 5964 Jan 6 2000 /usr/lib/uucp/remote.unknown
---s--x--x 1 uucp uucp 166448 Jul 12 04:53 /usr/lib/uucp/uucico
---s--x--x 1 uucp uucp 33620 Jul 12 04:53 /usr/lib/uucp/uusched
---s--x--x 1 uucp uu cp 82932 Jul 12 04:53 /usr/lib/uucp/uuxqt
-rwsr-xr-x 1 root bin 68856 Jul 12 04:51 /usr/openwin/bin/xlock
-r-sr-sr-x 1 root bin 18144 Dec 9 1999 /usr/openwin/bin/ff.core
-rwsr-xr-x 1 root bin 44096 Sep 28 2000 /usr/openwin/bin/sys -suspend
-rwsr-sr-x 1 root bin 24292 Nov 11 1999
/usr/openwin/bin/kcms_configure
-rwsr-sr-x 1 root bin 89792 Nov 11 1999
/usr/openwin/bin/kcms_calibrate
-rwsr-sr-x 1 root bin 31952 Nov 10 1999
/usr/openwin/bin/sparcv9/kcms_configure
-rwsr-xr-x 1 root bin 27620 Dec 16 1999 /usr/openwin/lib/mkcookie
-r-sr-sr-x 1 root sys 22808 Dec 2 1999 /usr/dt/bin/dtaction
-r-sr-xr-x 1 root bin 34036 Dec 2 1999 /usr/dt/bin/dtappgather
-r-sr-sr-x 1 root daemon 304176 Dec 2 1999 /usr/dt/bin/sdtcm_convert
-r-sr-xr-x 1 root bin 358340 Nov 8 2000 /usr/dt/bin/dtprintinfo
-r-sr-xr-x 1 root bin 166336 May 17 15:35 /usr/dt/bin/dtsession
-r-sr-xr-x 1 root sys 28196 Mar 16 2000 /usr/bin/sparcv7/ps
-r-sr-xr-x 2 root bin 11368 Jan 6 2000 /usr/bin/sparcv7/uptime
-r-sr-xr-x 2 root bin 11368 Jan 6 2000 /usr/bin/sparcv7/w
-rwsr-xr-x 1 root sys 37780 Jan 24 2001 /usr/bin/at
-rwsr-xr-x 1 root sys 13732 Jan 24 2001 /usr/bin/atq
-rwsr-xr-x 1 root sys 12692 Jan 24 2001 /usr/bin/atrm
-r-sr-xr-x 1 root bin 17072 Jan 24 2001 /usr/bin/crontab
-r-sr-xr-x 1 root bin 13808 Jan 6 2000 /usr/bin/eject
-r-sr-xr-x 1 root bin 26372 Jan 6 2000 /usr/bin/fdformat
-r-sr-xr-x 1 root bin 29312 Feb 26 2001 /usr/bin/login
-rwsr-xr-x 1 root sys 7328 Jan 6 2000 /usr/bin/newgrp
-rwsr-xr-x 1 root sys 7764 Mar 16 2000 /usr/bin/newtask
-r-sr-sr-x 3 root sys 101744 Jan 6 2000 /usr/bin/passwd
-r-sr-xr-x 1 root bin 6508 Jan 6 2000 /usr/bin/pfexec
-r-sr-xr-x 1 root bin 21008 Jan 6 2000 /usr/bin/rcp
-r-sr-xr-x 1 root bin 55480 Jan 6 2000 /usr/bin/rdist
-r-sr-xr-x 1 root bin 16012 Jan 6 2000 /usr/bin/rlogin
-r-sr-xr-x 1 root bin 8964 Jan 6 2000 /usr/bin/rsh
-r-sr-xr-x 1 root sys 17564 May 2 2001 /usr/bin/su
```

```

-r-s--x--x 1 uucp bin 55228 Jan 6 2000 /usr/bin/tip
-r-sr-sr-x 3 root sys 101744 Jan 6 2000 /usr/bin/yppasswd
-r-s--x--x 1 root sys 340260 Nov 2 2000 /usr/bin/admintool
-r-s--x--x 1 root lp 22460 Feb 26 2001 /usr/bin/lp
-r-s--x--x 1 root lp 9736 Jan 6 2000 /usr/bin/cancel
-r-s--x--x 1 root lp 7116 Jan 6 2000 /usr/bin/lpset
-r-s--x--x 1 root lp 21704 Feb 26 200 1 /usr/bin/lpstat
-r-sr-sr-x 1 root sys 37096 Mar 16 2000 /usr/bin/sparcv9/ps
-r-sr-xr-x 2 root bin 15392 Jan 6 2000 /usr/bin/sparcv9/uptime
-r-sr-xr-x 2 root bin 15392 Jan 6 2000 /usr/bin/sparcv9/w
-r-sr-xr-x 1 root sys 41708 Jan 6 2000 /usr/bin/chkey
-r-sr-sr-x 3 root sys 101744 Jan 6 2000 /usr/bin/nispasswd
-r-sr-xr-x 1 root bin 38740 Jan 24 2001 /usr/bin/rmformat
-r-sr-xr-x 1 root bin 5980 Jan 6 2000 /usr/bin/volcheck
-r-sr-xr-x 1 root bin 12580 Feb 26 2001 /usr/bin/volrmmount
---s--x--x 1 root uucp 69784 Jan 6 2000 /usr/bin/ct
---s--x--x 1 uucp uucp 83808 Feb 26 2001 /usr/bin/cu
---s--x--x 1 uucp uucp 67176 Jul 12 04:53 /usr/bin/uucp
---s--x--x 1 uucp uucp 22588 Jul 12 04:53 /usr/bin/uuglist
---s--x--x 1 uucp uucp 19568 Jan 6 2000 /usr/bin/uuname
---s--x--x 1 uucp uucp 62012 Jul 12 04:53 /usr/bin/uustat
---s--x--x 1 uucp uucp 71032 Jul 12 04:53 /usr/bin/uux
-r-sr-xr-x 1 root bin 229416 Feb 21 2001 /usr/bin/pppd
-r-sr-xr-x 1 root bin 12916 Jan 6 2000 /usr/sbin/sparcv7/whodo
-rwsr-xr-x 3 root bin 17616 Jan 6 2000 /usr/sbin/allocate
-rwsr-xr-x 1 root bin 9800 Jan 6 2000 /usr/sbin/mkdevalloc
-rwsr-xr-x 1 root bin 10032 Apr 13 2000 /usr/sbin/mkdevmaps
-r-sr-xr-x 1 root bin 48028 Jan 6 2000 /usr/sbin/ping
-rwsr-xr-x 1 root sys 22640 Jan 6 2000 /usr/sbin/sacadm
-r-sr-xr-x 1 root bin 35652 Jan 6 2000 /usr/sbin/traceroute
-rwsr-xr-x 3 root bin 17616 Jan 6 2000 /usr/sbin/deallocate
-rwsr-xr-x 3 root bin 17616 Jan 6 2000 /usr/sbin/list_devices
-r-sr-xr-x 1 root bin 61508 Dec 9 1999 /usr/sbin/afbconfig
-r-s--x--x 1 root lp 6856 Feb 26 2001 /usr/sbin/lpmove
-r-sr-xr-x 1 root bin 17408 Jan 6 2000 /usr/sbin/sparcv9/whodo
-r-sr-xr-x 1 root bin 58980 Dec 9 1999 /usr/sbin/ffbconfig
-r-sr-xr-x 1 root bin 37260 Nov 1 1999 /usr/sbin/igsconfig
-r-sr-xr-x 1 root bin 28784 Apr 23 2001 /usr/sbin/m64config
-r-sr-xr-x 1 root bin 29640 Jun 5 15:24 /usr/sbin/pmconfig
-r-sr-xr-x 1 root bin 5584 Jan 6 2000 /usr/sbin/aspppls
-r-sr-xr-x 1 root bin 767844 Nov 30 2000 /usr/sbin/static/rcp
-r-sr-xr-x 1 root sys 22988 Jan 6 2000 /usr/ucb/sparcv7/ps
-r-sr-xr-x 1 root sys 31544 Jan 6 2000 /usr/ucb/sparcv9/ps
-r-sr-sr-x 1 bin bin 9836 Jan 9 2000 /usr/vmsys/bin/chkperm
---s--x--x 1 root root 60004 Sep 26 1999 /usr/local/bin/sudo
-rwsr-xr-x 1 root other 85 0552 Oct 17 08:02 /usr/local/bin/ssh
-r-sr-xr-x 1 lp lp 203 Dec 16 1999 /etc/lp/alerts/printer
-rwsr-sr-x 1 root other 408 Jun 8 2000
/etc/tivready/monitorslfs/IBM_HTTP_Server_1.3.6_for_Solaris.slf

```