Global Information Assurance Certification Paper

Auditing a Squid Web Proxy Server: An Auditor's Report

GSNA Practical Version 3.1 (February 24, 2004)

Steve Mancini
GSEC, CGIH

Abstract

This paper is submitted as the requirement for a practical in the GSNA certification track. The subject of this audit is a Squid web proxy server that is used in a corporate environment. The squid web proxy server enables users behind the company firewall to access the internet without exposing their systems to internet directly. The goal of this practical is to raise awareness to the administrators and management of the server by identifying the high risks which may be associated with offering such a service and by auditing those controls which should be in place to help reduce that risk.

Table of Contents

1 Audit Description (Audit Plan)

1.1 Audit Scope

The scope of this audit is a proxy server that has been established to provide services to meet the needs not obtainable through traditional IT services. The system in questions is an IBM Mpro IntelliStation. The following chart enumerates the hardware details.

Model	IntelliStation Mpro
Manufacturer	IBM
Processor	Pentium II 450 MHz
Number of Processors	2
Memory	128
Network Card	Intel Corporation 82557 Ethernet Pro 100

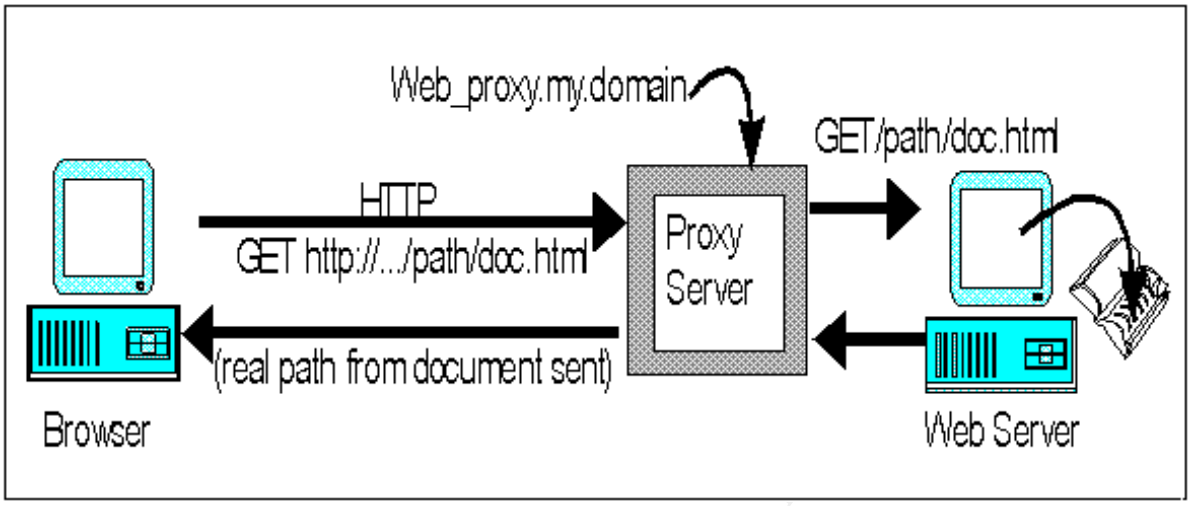
The system is running Red Hat 7.1 and has the following additional software installed:

- ☐ Squid proxy server (2.4STABLE6)
- ☐ Tripwire (2.3.0.47)

One of the critical attributes of this service is that the administrator is unable to disable the service to make upgrades without engaging in an internal change order process. This process requires them to obtain approval from their customers before any administrator initiated outage is permitted. As a result, there a high likelihood that the OS and software are not patched in a timely manner.

Application controls:
Squid is web proxy service that is used by client systems to access information on the intranet. It established outbound (from the company) connections for clients behind the company firewall – thus allowing full access without exposing the client systems to the full threats of the internet:

Table 1.1: HTTP transaction via proxy server[1]



A client will communicate with the outside internet through the squid proxy server utilizing the following protocols:

- http, https
- ftp
- ssl

1.2 Risk Assessment

In assessing a system of this nature, there are numerous configuration settings, behaviors, and infrastructure dependencies that could be audited; the sum of which would far exceed a reasonable amount of time to audit given the constraints of this document. As such, it becomes necessary to decide which areas to focus upon. The best option for doing this is to assess the risks faced by the system and service it provides and to focus upon the highest risks in this initial audit. Industry knowledge defines risk as the likelihood of a successful attack given certain threats and vulnerabilities. This concept is most often depicted by the following formula:

Risk = Threat x Vulnerability x Consequence[2]

To better understand the risks that this system we shall briefly enumerate and rank the threats, vulnerabilities, and consequences of a success attack.

[1] http://vms.process.com/~help/helpproxy.html
[2] ISO 1779

1.2.1 Threats

A threat is defined as "Activity that could negatively affect the C/I/A of a given system or service". As per sans there are 3 primary types areas of threat (see below- business goals, validated data, and widely known). Threats can be considered *agents of risk*.[3]

Table 1.2: Threats

Ref	Threat	Damage Capacity
T01	Employee	High
T02	Automated Attack (Internal)	Moderate
T03	Attacker (Internal)	Moderate
T04	Automated Attack (External)	Low
T05	Attacker (External)	Low
T06	"Act of God"	Low

Relevant notes for Threat Table:

- An internal attack is defined as someone/something who is not an employee who has access to the corporate networks without having to penetrate the perimeter security of the corporation.
- An external attack is defined as someone/something who must penetrate the perimeter defenses of the corporation.
- An automated attack would be any program or code that is executed against the server and/or services it supports without the attacker actively controlling its behavior, such as a virus or worm.
- Much like it is used in the insurance industry, an Act of God refers to those natural and statistically improbable events that lack any logical method for prediction except by virtue of the fact that experience shows planes crash, earthquakes happen, floods occur, etc.

1.2.2 Vulnerabilities

Vulnerability is a "weakness in your system or process that allows a threat to occur."[4] Whereas most threats are beyond the control of the security agents and administrators, vulnerabilities can be prevented, and in the case of an audit, corrected. The impact of a vulnerability is based upon the threat that it is coupled with. Connecting your computer to your company's wireless network incurs a certain degree of risk; connecting it to the every-man-for-himself wireless network at Defcon is a significantly greater risk.

[3] SANS Essential 2003, Chapter 7, p 303
[4] SANS Security Essentials 2003, Chapter 7, p 305

Table 1.3: Vulnerabilities

Ref	Vulnerability	Exposure
V01	Inadequate documentation – administrative documentation	Low
V02	Inadequate documentation – disaster recovery plan	Moderate
V03	Improper configuration - hardware	Low
V04	Improper configuration - operating system	High
V05	Improper configuration - software	High
V06	Improper configuration - network	Low
V07	Inadequate security patch maintenance (operating system)	High
V08	Inadequate security patch maintenance (software)	High
V09	Inadequate physical controls in place	Low

Each vulnerability exposes the company to potentially different degrees of impact based upon the access that this vulnerability affords the attacker. In the case of a denial of service, an extended loss of service would be covered in part by the redundant servers deployed across the corporation but would require individual reconfiguration of user applications which are hard-coded to point toward this proxy server. The attacks against confidentiality which bypass the security measures in place through the software (ssl, ssh) would presumably have the greatest impact on the users personal matters which are conducted through the proxy server. In the case of non-compliance with corporate policy (which mandates all company transactions of a sensitive nature be encrypted) then a greater impact would be seen as potentially classified information was disclosed.

Table 1.4: Potential Impact

Ref	Potential Impact
V01	Inadequate documentation – administrative documentation o A low degree of confidence with regard to the integrity of the system/service image. o An increased ambiguity as to what changes were made on the system when it is compromised which then requires the system to be rebuilt from scratch after any incident.
V02	Inadequate documentation – disaster recovery plan o Complicates the ability to recover from a loss of service and aggravates the outage's impact. o Delays in the restoration of service as tribal knowledge surrounding the landing of the proxy server must be re-discovered.
V03	Improper configuration - hardware o Loss of proxy services requiring a replacement/rebuild of the system o Lost confidence in the integrity of the system (potentially requiring the system to be rebuilt), o Jeopardize the confidentiality of the data sent on the system
V04	Improper configuration - operating system o Loss of proxy services, requiring a restart of the services. o Lost confidence in the integrity of the system (potentially requiring the

	system to be rebuilt), o Jeopardize the confidentiality of the data sent on the system
V05	Improper configuration - software o Loss of proxy services, requiring a restart of the services. o Lost confidence in the integrity of the service applications (potentially requiring the software to be rebuilt), o Jeopardize the confidentiality of the data sent on the system
V06	Improper configuration - network o Loss of proxy services, requiring a restart of the services. o Lost confidence in the integrity of the service applications (potentially requiring the software to be rebuilt), o Jeopardize the confidentiality of the data sent on the system o The system could also be used to gain further access to the intranet of the company.
V07	Inadequate security patch maintenance (operating system) o Loss of proxy services, requiring a restart of the services. o Lost confidence in the integrity of the system (potentially requiring the system to be rebuilt), o Jeopardize the confidentiality of the data sent on the system
V08	Inadequate security patch maintenance (software) o Loss of proxy services, requiring a restart of the services. o Lost confidence in the integrity of the service applications (potentially requiring the software to be rebuilt), o Jeopardize the confidentiality of the data sent on the system (which could expose private employee information and potentially company data which is non-compliant with corporate policy to encrypt all data passing through the firewall)
V09	Inadequate physical controls in place o Loss of proxy services requiring a replacement/rebuild of the system o Lost confidence in the integrity of the system (potentially requiring the system to be rebuilt), o Jeopardize the confidentiality of the data sent on the system (which could expose private employee information and potentially company data which is non-compliant with corporate policy to encrypt all data passing through the firewall)

1.2.3 Consequences

The consequences of an attack detail the impact that a successful exploitation would have on the system and the organization's defined business goals that are associated with the compromised system or service. The below table details the possible consequences if the availability, confidentiality, or integrity of the system is jeopardized. This table makes no assumptions with regard to consequential

impact (example – I compromise the integrity of the system and thereafter breach the servers ability to provide confidential transactions) each aspect is assessed individually.

Table 1.5: Consequences

Ref	Service	Attack	Affect	Severity
C01	Squid	Avail	Unable to access internet web sites	Low
C02		Conf	Employee's web transactions.	High
C03		Integ	Web transactions	High
C04	Squid (ftp)	Avail	Ability of users to transfer files through company firewall	High
C05		Conf	Content of downloads exposed (Potentially passwords)	Mod
C06		Integ	Software Updates/downloads	Mod

1.2.4 Risk (revisited)

Relying upon the above Threat, Vulnerability and Consequence tables we can build a matrix of potential risks. Given the geometric growth of this matrix (5x9x5) we would have 225 potential outcomes and associated risks. For the purpose of this paper, we shall focus on the high ranked attributes and from them distill the risks which should be of greatest concern to the client, this provides for a smaller matrix (1x4x3) which we can work and thus focus our resources and audit accordingly.

Table 1.6: High Risks

Risk	T	V	C	Description
R01	T02	V04	C02	Relying upon improper operating system configurations, an employee successfully compromises the confidentiality of web transactions through the proxy.
R02	T02	V04	C03	Relying upon improper operating system configurations, an employee successfully compromises the integrity of web transactions through the proxy.
R03	T02	V04	C04	Relying upon improper operating system configurations, an employee successfully jeopardizes the ability to download files through the proxy server.
R04	T02	V05	C02	Relying upon improper configurations of the proxy software, an employee successfully compromises the confidentiality of web transactions through the proxy.
R05	T02	V05	C03	Relying upon improper configurations of the proxy software, an employee successfully compromises the integrity of web transactions through the proxy.
R06	T02	V05	C04	Relying upon improper configurations of the proxy

				software, an employee successfully jeopardizes the ability to download files through the proxy server.
R07	T02	V07	C02	Relying upon inadequate patching of the operating system, an employee successfully compromises the confidentiality of web transactions through the proxy.
R08	T02	V07	C03	Relying upon inadequate patching of the operating system, an employee successfully compromises the integrity of web transactions through the proxy.
R09	T02	V07	C04	Relying upon inadequate patching of the operating system, an employee successfully jeopardizes the ability to download files through the proxy server.
R10	T02	V08	C02	Relying upon inadequate patching of the software, an employee successfully compromises the confidentiality of web transactions through the proxy.
R11	T02	V08	C03	Relying upon inadequate patching of the software, an employee successfully compromises the integrity of web transactions through the proxy.
R12	T02	V08	C04	Relying upon inadequate patching of the software, an employee successfully jeopardizes the ability to download files through the proxy server.

1.3 Current State of Practice

We start our examination of current states of practice by reviewing the website for the respective open source application. The authors of squid provide a configuration guide which can be found in pdf format off of their support website, http://www.visolve.com. This document proved to be thorough in the descriptions, but weak as far as security considerations or auditing methods.

One of the more concise documents which I discovered while searching for methods of security a squid proxy server is the GSEC certification paper "Security considerations with Squid proxy server" prepared by Eric Galameau. Eric provides a high level examination of some of the obvious risks (such as physical compromise of the system), but more importantly he delves into the configuration options for the squid software itself and highlights several key settings which can reduce the risk of exploitation. His paper can be found in the SANS Reading Room.

Anton Chuvakin wrote an article for security focus in September of 2001 entitled "Anonymizing with Squid Proxy". While this audit does not cover the topic of anonymizing, Chuvakin does recount several industry best practices as it pertains to the securing of a linux squid proxy server. These recommendations focus upon access control the server itself.

When one searches for best practices to secure linux, you will find a staggering 47,500+ relevant links available to you. Rather than sort through all of these sites, I am opting to distill the search to include some well known references that I believe to be industry best practices on the subject of securing linux:

- ✓ The Center for Internet Security's CIS Benchmark for Linux . This includes their benchmarking tool which can provide a quasi-accurate representation of the security posture of a system which is reduced to a score.[5] http://www.cisecurity.org/bench_linux.html
- ✓ Security Consensus Operational Readiness Evaluation (SCORE). Specifically we will incorporate the System Security Plan checklist and the 1.0 for the Auditing Unix checklist.

When searching for unix audits, one of the sites which proved to capture a great number of recurring checklist items is The Unix Auditor's Practical Handbook by KK Mookhey. This audit covers many of the principle areas which are considered industry best methods: physical, operating system, network, user and file system security.

Bypassing the rest of this overwhelming number of sources, I've opted to place heightened focus upon papers submitted to the SANS reading room (not only because of the quality of these papers but also to verify I am not duplicating effort). "Auditing Redhat Linux 7.0" by Mary Laude (GSNA) was submitted in July of 2001 and provides an excellent checklist for several of the core specifications for a secure operating system. Building upon the work by Mary, I also selected the following papers which focused upon a specific service being offers upon a linux operating system, since this parallels the effort of this current audit:

- Sean Beauamann's "Auditing a Linux FTP and DNS Server: An Administrator's Perspective", September 20, 2003. http://www.giac.org/practical/GSNA/Sean_Baumann_GSNA.pdf
- Leigh Haig's "Auditing a CacheFlow Proxy Solution: An Auditor's Perspective", July 4 2003. http://www.giac.org/practical/GSNA/Leigh_Haig_GSNA.pdf
- Eric Tong's "Auditing a Linux Point-to-Point Tunnel Propotocol (PPTP) Virtual Private Network (VPN) Server: An Auditor's Persective." July 2003. http://www.giac.org/practical/GSNA/Eric_Tong_GSNA.pdf

In addition to the literature available externally, the corporation in which this server was established has its own internal security specification which are required for a system to be considered safe enough to be connected to the production network and DMZ. Due to corporate policy, the exact details of these specifications are not publishable so we will focus on the more intuitive requirements that are aligned with industry best practices as found in other documents.

[5] The accuracy of this score is a hotly debated topic – it does however provide the user a measurable method of seeing an improvement a they implement additional controls on the box.

- All documentation will be controlled according to company policies.
- Changes, updates, and corrections to documents will be logged at the beginning of each document.
- Installation and configuration will be completely documented in a highly granular, step-by-step, format.
- Specific system maintenance and operating procedures will be documented.
- An incident response plan will be devised and documented.
- The official company security banner must be display prior to any system access.
- User and administrator passwords will adhere to the company strong password policy.
- All system events and alerts will be centrally logged.
- All encrypted web traffic must be decrypted before reaching the web servers and monitored for intrusions.
- Physical access to the environment must be controlled and audited.
- Physical access to systems must be controlled and audited.
- Software and hardware licenses will be monitored and stored when necessary.
- Software and hardware upgrades and patches will only be accepted from authorized sources.
- The use of any security tools is not authorized by company security.

2 Audit Checklist

2.1 Scope

The scope of this audit is primarily concerned with the squid service that supports the business goals for the establishment of the proxy server. However, in order to evaluate risk properly, other areas will require review, such as the policies, standards and procedures, the physical location of the server, the router configuration and any access control lists (ACLs), and the configuration of the operating systems. As a result our checklist will cover the following 5 areas of control:

- ✓ Physical security
- ✓ Administrative planning
- ✓ OS hardening
- ✓ Network configuration
- ✓ Software (squid) configuration

2.2 Checklist Structure

The checklists are organized by the types of risk as described in Section 1. Each checklist step has the following elements.

Checklist Item:			
Objective:			
References:			
Risk:			
Test:			
Compliance Criteria			
Test Nature:	☐ Objective ☐ Subjective	☐ Intrusive ☐ Passive	☐ Setting ☐ Behavior
Evidence			
Findings:	.		
Results:	☐ Pass ☐ Fail		

Checklist Item
The identifier is a unique name for each step in the checklist.

Objective
The objective is the description and goal of a particular step.

Reference
The reference indicates the source of the step, either from a reference or an original contribution from the auditor.

Risk
The risk identifies which type and element of risk the step is addressing.

Test
The test describes the action taken to determine if the system passes or fails a particular step.

Compliance Criteria
The compliance element describes the criteria for compliance to the step

Test Nature
These entries describe the type of test you are working with. Objective/Subjective describes the decision process used to determine if an auditable item passes or fails. *Intrusive/Passive* describes the type of tests which will be conducted, whether they will affect the system or merely observe a setting[6]. *Setting/Behavior* describe what it is that is being tested- whether is an entry or if you witness the system/administrator reaction in real time.

Evidence
This is a place-marker in the checklist for evidence that is generated by the testing procedure

Findings
Findings are the conclusions that your draw from your evidence and your compliance criteria. By reserving a place in your checklist, you can copy and paste the checklist item into Part #3

Results
Ultimate decision on whether the audit item was considered a pass or fail.

[6] Data collection, which does not directly influence or trespass upon the target, is called a passive attack. Security tests that intrude on the system, and that can be monitored and logged, and could generate alerts are called intrusive attacks. – Open Source Security Testing Methodology Manual by Peter Herzog (2002)

2.3 Checklists

2.3.1 Physical Checklists

Checklist Item:	Physical-01		
Objective:	The proxy server will reside in a location designed to reduce risks associated with physical access to the system.		
References:	Industry BKM: "If a bad guy has unrestricted physical access to your computer, it's not your computer anymore" - 10 Immutable Laws of Security		
Risk:	The major threats from physical access are denial-of-service and inappropriate access to the equipment. Physical access means the system can (potentially) be physically disabled, data on the permanent media storage devices stolen, or the box can be rebooted and administrative access obtained.		
Test:	Is system located in a restricted access data center? Building access is monitored Room access is monitored Room access granted through ACL ACL is reviewed periodically All entrances to room are locked		
Compliance Criteria	1. All entrances to building are locked 2. All entrances to building are monitored 3. Entrances to room are authenticated 4. Entrances to room are monitored 5. Cameras view all entrances monitored by security services 24x7 6. Doors left open for extended period of time are responded to by site security. 7. Room access granted through ACL 8. ACL is reviewed periodically		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Checklist Item:	Physical-02		
Objective:	The proxy server will reside in a location that provides adequate from physical damage due to physical (environmental) threats.		
References:	Corporate Policy on Data Center Criteria		
Risk:	Inadequate protections may result in catastrophic loss of the system and service it provides in the event of an environment incident.		
Test:	Review physical location for the existence of the following 1. Regulated Power 2. Uninterruptible Power Supply 3. Fire protection equipment (detectors/alarms/extinguish) 4. Humidity detection equipment 5. Flood/water detection equipment 6. grounding straps 7. telephone w/ immediate call capability		
Compliance Criteria	Given physical location power and environmental protections should be deployed to protect data center assets. The presence or absence of any of the above should be based upon corporate policy a cost analysis against catastrophic loss.		
Test Nature:	✓ Objective ☐ Subjective	☐ Intrusive ✓ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Checklist Item:	Physical-03
Objective:	The hardware will be configured so as to reduce risks associated with physical access to the system.
References:	"If a bad guy has unrestricted physical access to your computer, it's not your computer anymore" - 10 Immutable Laws of Security
Risk:	The major threats from physical access are denial-of-service and inappropriate access to the equipment. Physical access means the system can (potentially) be physically disabled, data on the permanent media storage devices stolen, or the box can be rebooted and administrative access obtained.

Test:	Examine build documentation. 1. Is the bios password set by design? 2. Is the boot password set by design? 3. Are the devices (floppy/cd-rom/etc) removed or disabled as bootable devices by design? Insert bootable OS in drive (knoppix[7]) Boot system: 4. Is the bios password set by design? 5. Is the boot password set by design? 6. Are the devices (floppy/cd-rom/etc) removed or disabled as bootable devices by design?		
Compliance Criteria	1. The bios password is set by design. 2. The boot password is set by design. 3. All devices (floppy/cd-rom/etc) are removed or disabled as bootable devices by design. 4. You are prompted for a bios password 5. You are prompted for a boot password 6. The bootable cd OS does not load.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Checklist Item:	Physical-04
Objective:	The operating system will be configured so as to reduce risks associated with physical access to the system.
References:	"If a bad guy has unrestricted physical access to your computer, it's not your computer anymore" - 10 Immutable Laws of Security
Risk:	Physical access means the system can (potentially) be physically disabled, data on the permanent media storage devices stolen, or the box can be rebooted and administrative access obtained.

[7] Knoppix is a bootable linux image. For more information see: http://www.knoppix.org

Test:	1. Access to lilo.conf file is restricted to root.
	`/bin/ls -l /etc/lilo.conf`
	2. The LILO prompt password protected.
	`/bin/grep -A 3 "prompt" /etc/lilo.conf`
	3. Reboot from console w/ Ctrl+Alt+Del is disabled.
	`/bin/grep -A 3 "CTRL-ALT-DELETE" /etc/inittab`
	4. Root password is required to enter single user mode
	`/bin/grep -A 2 "sysinit" /etc/inittab`
	5. Console logins restricted to root and authorized users
	`/bin/grep -A 3 "console login" \`
	`/etc/security/access.conf`
	Attempt to login at console w/ non-authorized account.
	6. Screensaver in enabled and autolocks after 15 minutes idle
	Allow system to remain idle – verify screen saver engages
	7. Screensaver requires password of console logged in user to unlock.
	Attempt to unlock screensaver w/o password. Then with password.		

Compliance Criteria	1. /etc/lilo.conf will be owned by root with permissions set to 600
	2. The 2 directives following the prompt directive will be:
	`password =` <Your_LILO_Password>
	`restricted`
	3. The directive allowing this has been commented out.
	`# Trap CTRL-ALT-DELETE`
	`#ca::ctrlaltdel:/sbin/shutdown -t3 -r now`
	4. The *wait* directive has been added below the *sysinit* directive.
	`# System initialization.`
	`si::sysinit:/etc/rc.d/rc.sysinit`
	`~~:S:wait:/sbin/sulogin`
	5. The line restricting access will be uncommented:
	# Disallow console logins to all but a few accounts.
	#
	-:ALL EXCEPT wheel shutdown sync:LOCAL
	6. Screensaver should enable in 15 minutes.
	7. Only password that should unlock screen is that of the user logged in on the console.		

Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

2.3.2 Administrative Checklists

Item Number:	Administrative-01		
Objective:	Ensure that all administrators are required to attend all necessary information security training courses		
Reference:	Corporate Policy mandates all system admins take courses which cover topics necessary to help reduce risk in the daily operation of their job functions.		
Risk:	Lack of proper education by those responsible for systems increases likelihood of compromise due to ignorance		
Test:	Verify all those with administrative responsibilities have taken all required courses.		
Compliance	All those with administrative access to the system will have taken the corporate courses covering the following: General information security awareness System administration & hardening Resisting social engineering		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Checklist Item:	Administrative-02		
Objective:	An administrative plan of record exists which details the purpose of the system, the policies related to it, the information it will provide in support of those policies.		
Reference:	Corporate Policy		
Risk:	Improper administration practices can result in risks to the squid and socks services. Without proper policies, standards and procedures, unauthorized access may reside in a questionable grey area in the case of an internal (employee) threat and the punitive measures that may be taken.		
Test:	Ask to see administrative documentation for system (hard copy or online)		
Compliance Critieria	Documentation will cover: ✓ System ownership/responsibilities ✓ Purpose of the system ✓ Acceptable Use standards		
Test Nature:	✓ Objective ☐ Subjective	☐ Intrusive ✓ Passive	✓ Setting ☐ Behavior

Evidence	
Findings:	
Results:	☐ Pass ☐ Fail

Item Number:	Administrative-03		
Objective:	Change management documentation must exist for all critical infrastructure systems.		
Reference:	Industry Best Practice, Personal BKM		
Risk:	With proper documentation detailing changes to the system the ability to discern legitimate changes from unauthorized changes is called into question.		
Test:	Ask to see administrative documentation for system (hard copy or online) Question administrators with regard to change procedures for proxy server. Using the unix command *find,* verify no system files have been modified since the last entry in the change control system.		
Compliance Critieria	Documentation will cover: ✓ Original baseline configuration ✓ Any authorized changes to the system ✓ Any detected unauthorized changes to the system, and the steps that were taken to re-secure the system Administrators will demonstrate knowledge of documentation and where to find documentation in case of need.		
Test Nature:	✓ Objective ☐ Subjective	☐ Intrusive ✓ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Administrative-04
Objective:	All software on the system shall be legally obtained, and currently supported by the supplier.
Reference:	Corporate Policy Aubry
Risk:	Legal ramifications for using un-licensed software aside, there is no assurance that vendors will report the applicability of current

	vulnerabilities in software versions which are no longer supported.		
Test:	For each of the following, verify either proof of purchase, or that the software is free of charge (freeware) for corporate use in the manner it has been established: 1. Operating System 2. Squid Proxy Service 3. Tripwire IDS		
Compliance	All software will either be freeware or the administrators can produce a receipt for purchase of a license.		
Test Nature:	✓ Objective ☐ Subjective	☐ Intrusive ✓ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Administrative-05		
Objective:	Appropriate warning banners must be in place and issued upon successful connection to the system		
Reference:	Corporate policy. SANS hardening recommendation.		
Risk:	Protection in case of legal prosecution		
Test:	Connect to machine, observe MOTD. Add the required text to /etc/rc.d/rc3.d/S99local. Note that this file copies the text to /etc/issue and /etc/issue.net upon system boot. If that functionality is removed, make the changes to /etc/issue and /etc/issue.net manually.		
Compliance	The Message of the Day should contain the following information: “Use of this system by unauthorized persons or in an unauthorized manner is strictly prohibited. Keystroke and network logging may be in effect at any time. “		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Administrative-06

Objective:	An incident handling plan must be in place which will provide administrators the procedures and tools necessary to properly handle an event or incident involving the proxy server.		
Reference:	SANS Track 4 Personal Experience		
Risk:	Insufficient planning for an event/incident can result in damage to the evidence of the situation which in turn can complicate your ability to • determine how the breach occurred • take action against those responsible		
Test:	1. Examine the incident handling plan 2. Examine incident handling disk		
Compliance	1. The administrators should have available in printed format an incident handling procedure which covers the major phases of handling an incident: • Who to contact – roles & responsibilities • Steps to take to identify/determine the situation • Steps to take to contain the threat • Steps to take to eradicate the vulnerability/threat • Steps to take to recover from the event. 2. The administrators should have a responders disk prepared in advance with known good binaries for the operating system that will be used in addressing an incident.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Administrative-07
Objective:	A Disaster Recovery plan must exist which will provide administrators the equipment, resources, and instructions to replace system in reasonable amount of time.
Reference:	Beaumann Corporate Policy
Risk:	The inability to recover from a hardware failure for any reason can result in aggravated outage of service.
Test:	1. Ask to see printed disaster recovery plan for the proxy server. Ask to see online version of disaster recovery plan for proxy server 2. Verify the existence of, or replacement procedure for,

	damaged hardware. 3. Ask to see the last system backup of the proxy server.		
Compliance	1. There is no difference between the written and online content of the plan. This plan will include detailed instruction necessary to replace the current proxy server without the need of any "tribal knowledge". 2. Hardware will either be pre-existing or obtainable within sufficient time to meet the company service level agreement. 3. System backup should be less than 1 week old or should not deviate in configuration from current running system.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

2.3.3 OS Hardening Checklists

Item Number:	Hardening-01
Objective:	Authentication data and procedures will be implemented to reduce risk of unauthorized access.
Reference:	Personal Experience, CIS Level-1 Benchmark and Scoring Tool for Linux". URL: http://www.cisecurity.org/bench_linux.html (15 June 2002).
Risk:	An attacker could obtain access to the system through the use of a weak or non-existent passwords or brute force an encrypted

	password string acquired through exposed password file.		

Test:	1. Check for the use of shadow passwords:
	`/bin/sudo /bin/more /etc/password`
	`ls –l /etc/shadow`
	`/bin/sudo /bin/more /etc/shadow`
	2. Check for the existence of empty passwords:
	`awk –F: '($2 == "") {print $1}' /etc/shadow`
	3. Check for the number of accounts assigned to uid 0:
	`awk -F: '($3 == 0) {print $1}' /etc/passwd`
	4. Passwords should be configured to age and have a minimum password length:
	`/bin/grep PASS_MIN_LEN /etc/login.defs`
	`/bin/grep and PASS_MAX_DAYS /etc/login.defs`
	5. Obtain copy of password file and execute john the ripper in brute force mode for 90 days[8].		

Compliance Criteria	1. Shadow passwords will be configured on the system such that: • passwd file should show no password entries in encrypted string field in the second stanza `root:x:0:0:root:/root:/bin/bash` • /etc/shadow should not be readable by anyone but root (600) • All entries should be composed of either 1. The hash of the password 2. A character disabling the password, usually * 2. There should be no passwords with a blank password. 3. There should be only 1 account with uid 0 4. The PASS_MAX_DAYS should not be greater than 90 and PASS_MIN_LEN 8 5. John the Ripper should not be able to brute force any passwords.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-02

[8] Instructions for the installation and use of john the ripper can be obtained at http://www.openwall.com/john

Objective:	All system account passwords will be disabled through use of no usable password and a disabled login shell.		
Reference:	Chuvakin,		
Risk:	Unauthorized users could access system accounts and their use can go un-noticed.		
Test:	1. Examine password file for use of interactive login shells `/bin/grep –v nologin /etc/password` 2. /bin/cat /etc/security/access.conf 3. /bin/grep “pam.access.so” /etc/pam.d/login		

Compliance Criteria	1. Results of grep should reveal no system accounts except root
	2. The access.conf file could contain entries such as:
	`-:ALL EXCEPT root:LOCAL`
	`-:named smmsp:ALL`
	3. The grep should yield the following entry:
	`account required /lib/security/pam_access.so`		

Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-03
Objective:	Disable un-necessary services
Reference:	SANS Securing Linux step by Step CIS Level-1 Benchmark and Scoring Tool for Linux Laude
Risk:	Un-necessary services result in additional unjustifiable exposure to potential unauthorized access or denial-of-service.

Test:	1. Verify none of the following services are configured through xinetd: (finger, ntalk, rexec, rlogin, rsh, telnet, talk)
	`/bin/ls /etc/xinetd/*`
	2. Review which services are started as daemons on system:
	`/bin/grep “disable” /etc/xinetd.d/*`
	3. Verify only necessary services are run at boot:
	`/path/to/chkconfig –list | /bin/grep “:on”`
	4. Verify there are no un-ncessary services listening
	`/bin/netstat –anp | /bin/grep –I “listen”`
	5. There are no print services running on the system:
	`/bin/ps –ef | grep lpd`
	6. There are no web servers running on the sytem:
	`/bin/ps –ef | grep httpd`

	7. NFS is not installed on the system:
	`/bin/rpm –qa | grep knfsd`
	8. Sendmail is listening only to localhost.
	`telnet proxy_server 25`		

Compliance	1. /etc/xinetd.d does not contain any of the following files: finger, ntalk, rexec, rlogin, rsh, rexec, telnet, talk 2. Only services necessary are enabled are required[9] 3. For any service which is returned, verify that it is critical to run the server 4. The output of netstat should reveal only services which are open by design and critical to the operation of the system. 5. Print services should not be running 6. Web services should not be running 7. NFS should not be running 8. sendmail on port 25 should not receive remote connections.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-04
Objective:	Remote access (authentication) on to system is limited to SSH.
Reference:	Securing linux
Risk:	The standard Unix remote access utilities of telnet and rlogin are not adequate in any environment today. Telnet and rlogin send passwords in clear-text over the network. All authentication for remote access should at least be encrypted, and strong, two-factor authentication should be considered for mission-critical systems

Test:	1. Is ssh installed on the system?
	`/bin/locate sshd`
	`/bin/locate ssh`
	2. Is ssh current version?
	`/usr/bin/ssh --V`
	3. Verify that telnet, rsh, rlogin and rexec are disabled.
	`/bin/grep telnetd /etc/xinetd.conf`
	`/bin/grep telnetd /etc/xinetd.d/*`
	`/bin/ps –ef | grep telnetd`

[9] Services known to be un-necessary according to industry best practices:
apmd, autofs, gpm, innd, IrDA, isdn, kdcrotate, lpd, lvs, mars-nwe, named, netfs, nfs, nfslock, oki4daemon, portmap, routed, rstatd, rusersd, rwalld, rwhod, sendmail, smb, snmpd, webmin, ypbind, ypserv, yppasswdd

	`/bin/grep rshd /etc/xinetd.conf`
	`/bin/grep rshd /etc/xinetd.d/*`
	`/bin/grep rlogin /etc/xinetd.conf`
	`/bin/grep rlogin /etc/xinetd.d/*`
	`/bin/grep rexec /etc/xinetd.conf`
	`/bin/grep rexec /etc/xinetd.d/*`
	4. Is tcp wrappers configured to deny all remote access except ssh?
	`/bin/cat /etc/hosts.deny,`
	`/bin/cat /etc/hosts.allowed`
	5. Verify there are no **.rhosts** files on the system
	`/usr/bin/find / -name .rhosts –print`
	6. Verify that remote telnet connections are prevented through configuration of /etc/securetty
	`/bin/grep –v tty /etc/securetty`
	7. Verify there is no hosts.equiv file
	`/bin/ls –l /etc/hosts.equiv`

Compliance	1. Locate should return the location of the ssh and sshd binaries 2. The current version should be supported by vendor/author 3. None of the searched for binaries should exist enabled in either xinetd.conf or xinetd.d 4. /etc/hosts.deny should have ALL:ALL and /etc/hosts.allow should designate authorized systems only: `sshd: allowed.domain trustedhost.allowed.domain` 5. Find should yield no files. 6. The grep should return nothing (empty) 7. There should be no hosts.equiv file on the system
Test Nature:	✓ Objective ☐ Subjective | ✓ Intrusive ☐ Passive | ✓ Setting ☐ Behavior
Evidence	
Findings:	
Results:	☐ Pass ☐ Fail

Item Number:	Hardening-05
Objective:	Close all un-necessary ports
Reference:	“Every port with a listening service is a potential doorway into the machine for the attacker…” (Skoudis, 200). Skoudis, Ed. Counter Hack. Upper Saddle River, NJ: Prentice Hall, 2002
Risk:	Open ports potentially enable enumeration attacks against the system as well as pose the same threat as un-necessary services with regard to unauthorized access and/or denial of service

	attacks.		
Test:	1. Use portscan tool to verify which ports are open and what services are running on them.[10] `nmap -p 1-65000 proxy_server_ip`		
Compliance	1. All open ports should be open-by-design and running services approved for use on them.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-06		
Objective:	An Intrustion Detection System must be installed which routinely (daily) verifies the integrity of binaries on the system.		
Reference:	Personal Experience.		
Risk:	Un-detected attacks against a system can provide the attacker and extended opportunity to compromise the services.		
Test:	1. Verify that tripwire is installed on the system: `/usr/bin/find / -name tripwire -print` 2. Ask to see where tripwire is automatically initiated /usr/bin/crontab –u root -l 3. Examine tripwire configuration files		
Compliance	1. Tripwire is installed on the system: 2. Tripwire should be in a daily (or more frequent) cron: 3. Tripwire should be configured to track modifications to key programs often modified by attackers:		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-07
Objective:	Disable all un-necessary suid programs.

[10] Latest version of NMAP will attempt to provide the service and version offered on an open port. Pretty cool, huh?

Reference:	Corporate policy		
Risrisk:	Exploitable suid binaries owned by root ensure attackers their exploit will run with escalated privileges.		

Test:	1. Find all suid binaries on the system:
	`/bin/find / -perm -4000 –print`
	`/bin/find / -perm -2000 -print`		

Compliance	Any binary returned must be required for operation of the proxy server.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-08
Objective:	System and application logs are established, reviewed (daily), and rotated with 30 days left on system
Reference:	Corporate policy SANS Security Essentials (Track 1)
Risk:	The absence of logging enhances the ability of a threat to compromise a system and aggravates any post-incident analysis.

Test:	1. Verify that logging is enabled.
	`/bin/ps –ef | grep syslog`
	2. Verify that syslog is not only stored locally, but also sent to a central log service.
	`/bin/grep "@" /etc/syslog.conf`
	3. Verify log rotation is executed and retained for 30 days
	`/bin/cat /etc/logrotate.conf`
	4. Verify logs are analyzed and admin aware
	5. Verify squid logs are analyzed in accordance with corporate acceptable use policy
	6. Verify that log files are analyzed periodically (daily)

Compliance	1. The /etc/syslog.conf file should contain entries for auth, authpriv, error and warning conditions. Any action that should be logged is written to the appropriate file in /var/log directory. 2. In the /etc/syslog.conf file there should exist an entry sending relevant (or all) syslog entries to: @you_central_log_server. 3. Logs are to be rotated daily, and 30 days should be retained online. Look in the logrotate.conf file and verify that for each file that relevant information is logged to, that the following parameters exist either for the individual files or as a global setting:

	4. Ask administrator to demonstrate how important log events are detected and addressed. 5. Ask administrator to demonstrate that squid access and error logs are analyzed through an automated script. Logs should be stored according to corporate policy (30 days online, daily rotate) 6. Ask to see log analysis scripts established to examine syslog and squid logs. Script should highlight issues and should alert administrators automatically.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Hardening-09		
Objective:	The system will report no High or Medium vulnerabilities when subjected to vulnerability assessment tools.		
Reference:	Personal Experience		
Risk:	The risk associated with running a machine increases over time if the machine is not routinely assessed for new vulnerabilities. Timely scans with Vulnerability Assessment tools diminishes this risk.		
Test:	Execute the following vulnerability scans against the system ✓ Nessus scan ✓ CIS Security Tool ✓ Retina[11]		
Compliance	No tool should return any vulnerabilities that are not documented as known risks which have been accepted (by management) and mitigated through other measures where possible.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

[11] Retina is a commercial product and as such may not be available. While Retina and Nessus usually yield very similar results (especially in the case of high vulnerabilities), the overlap is worth the potential reduction in risk if it is available

2.3.4 Network Checklists

Item Number:	Network-01		
Objective:	Enable network-related security settings on the system to minimize risks associated with attacks that rely upon packet manipulation.		
Reference:	Beauman,		
Risk:	Improper network configurations on the system can result compromise, denial of service attacks against the system, or in the utilization of the system in a denial of service against other systems or network services.		

Test:	1. Log illegal packets (spoofs, source routing, redirects)
	`/bin/grep "log_martians" /etc/sysctl.conf`
	2. Prevent SYN floods. Examine /etc/sysctl.conf
	`/bin/grep "syncookies" /etc/sysctl.conf`
	3. Prevent routing table alterations via ICMP redirects
	`/bin/grep "accept_redirects" /etc/sysctl.conf`
	4. Enforce fragmentation protection to prevent frag overlaps or exploits
	`/bin/grep "always defrag" /etc/sysctl.conf`		

Compliance	For each of the above grep statements, you should yield the following results: 1. net.ipv4.iconf.all.log_martians = 1 2. net.ipv4.tcp_syncookies = 1 3. net.ipv4.conf.all.accept_redirects = 0 4. net.ipv4.ip_always_defrag = 1		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Network-02
Objective:	All company controlled network devices associated with proxy server will be configured to only allow necessary traffic to/from the proxy server.
Reference:	Corporate Policy
Risk:	Network devices in the path of the proxy are also responsible for protecting the service to the extent that they are able.

Test:	1. Review copy of running configuration of router for ingress filters 2. Review copy of running configuration of router for egress filter 3. From external location, use nmap to scan internal IP address		
Evidence	1. The following inbound filters are required: access-list 100 deny ip 10.0.0.0 0.255.255.255 any log access-list 100 deny ip 127.0.0.0 0.255.255.255 any log access-list 100 deny ip 172.16.0.0 0.15.255.255 any log access-list 100 deny ip 192.168.0.0 0.0.255.255 any log access-list 100 deny ip <your public address block> any log access-list 100 deny ip any 10.0.0.0 0.255.255.255 log access-list 100 deny ip any 127.0.0.0 0.255.255.255 log access-list 100 deny ip any 172.16.0.0 0.15.255.255 log access-list 100 deny ip any 192.168.0.0 0.0.255.255 log access-list 100 permit ip any any 2. The following outbound filters are required: access-list 101 permit ip <your public address block> any access-list 101 deny ip any any log 3. Traffic should be rejected according to established ACLs		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Findings:			
Evidence:			
Results:	☐ Pass ☐ Fail		

2.3.5 SQUID Checklists

Item Number:	Squid-01
Objective:	Squid shall be installed under and run by a non-root, login-disabled, account.
Reference:	Galarneau, p. 3, also Industry Best Practice: “Principle of Least Privilege”
Risk:	In the event of a buffer overflow, the exploit is run with the uid of the account running the process.

Test:	1. Verify owner of squid binary:
	`/bin/ls –l /usr/sbin/squid`
	2. Verify owner of the running squid process:
	`/bin/ps –ef | grep squid`
	3. Examine the password entry for the squid account:

	/bin/grep squid /etc/passwd 4. Examine cache_effective_user in squid.conf 5. Examine cache_effective_group in squid.conf		
Compliance:	1. The squid executable will not be owned by root. 2. The squid process should not be run by root. 3. The resource account which runs squid should have no interactive login capability. 4. Should be set to dedicated account 5. Should be set to dedicated/safe group		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Findings:			
Evidence			
Results:	☐ Pass ☐ Fail		

Item Number:	Squid-02		
Objective:	Implement access control lists (ACL) to restrict access to server by trusted network ranges only.		
Reference:	Galarneau		
Risk:	Open network settings provide attackers unfettered access to system and allow unauthorized internal users to utilize proxy services.		
Test:	1. Only trusted addresses should be allowed to connect to the proxy server. 2. Configuration should default to deny that which is not explicitly allowed. 3. Only necessary ports should be opened by proxy services.		
Compliance	1. Networks allowed to connect through the proxy server should be explicitly enumerated (example): *Acl all src 0.0.0.0/0.0.0.0* *Acl offices src 10.7.0.0/255.255.0.0* *http_access allow offices* *Acl labs src 10.8.0.0/255.255.0.0* *http_access allow labs* *http_access deny all* 2. The last acl should be: *http_access deny all* 3. The acceptable ports should be enumerated and all others should be denied (example): *Acl trusted_ports port 21 80 443* *http_access deny !trusted_ports*		
Test	✓ Objective	✓ Intrusive	✓ Setting

Nature:	☐ Subjective	☐ Passive	☐ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Squid-03		
Objective:	Implement squid ftp configurations to reduce risk of attacks against confidentiality and enumeration attacks.		
Reference:	Squid Documentation		
Risk:	Absence of proper configurations can result in loss of notifications regarding potentially missed abuses by your users executed against others.		
Test:	1. Examine setting for `ftp_user`[12] in squid.conf 2. Examine setting for `ftp_passive`[13] in squid.conf 3. Examine setting for `ftp_sanitycheck`[14] in squid.conf		
Compliance	1. Make sure the value is a valid email address which is checked regularly. Send the address an email and see if it is responded to. 2. ftp_passive should be set to ON 3. ftp_sanitycheck should be set to ON		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Squid-04
Objective:	Implement squid configurations (squid.conf) to reduce risk of denial of service attacks against the service.
Reference:	Squid documentation

[12] ftp_user assigns the default password sent by Squid to anonymous ftp sites
[13] The passive mode is considered to be more secure because it uses 2 fixed ports; one for connection and one for data transfer while the active mode uses ephemeral ports. Firewalls generally need fixups in order to handle an active FTP session.
[14] This option uses an extensive mechanism to ensure the connection is established with the requested server and it must be left on.

Risk:	Several configurations in squid allow for protection against over-allocation of resources which could starve legitimate use.		
Test:	Examine the following settings in the squid.conf file: 1. `maximum_object_size` 2. `quick_abort_min` 3. `quick_abort_max` 4. `quick_abort_pct` 5. `dns_nameservers` 6. `ignore_unknown_nameservers`[15] 7. `client_lifetime`[16] 8. `pconn_timeout`[17] 9. `request_header_max_size`[18]		
Compliance	1. `maximum_object_size` <= 4096 2. `quick_abort_min` = 16KB 3. `quick_abort_max` = 16KB 4. `quick_abort_pct` = 95 5. Verify nameservers in /etc/resolv.conf are trusted – if so, then this value should not be set. If they are not trusted, this should be set to trusted servers. Name servers used by squid must come from trustable sources and configured safely. A compromised DNS server is often used by attackers to divert proxy servers and certain Squid versions can be crashed by sending malformed DNS answers 6. should be enabled or trusted 7. client lifetime <= 24 hours 8. timeout <= 120 seconds 9. max size <= 10KB		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

Item Number:	Squid-05
Objective:	The proxy server will protect against the introduction of malware

[15] This option verifies if a nameserver answering the lookup has the same IP address as the one the lookup was sent to.
[16] The client_lifetime sets the maximum time a client is allowed to be bound to a Squid process
[17] The pconn_timeout sets the maximum time an *idle* client is allowed to be bound to a squid process.
[18] The request_header_max_size option is used to limit the size of acceptable HTTP headers.

	into the environment.		
Reference:	http://www.squidguard.org		
Risk:	Users are prone to accepting hostile/dangerous code.		
Test:	Verify the installation of HTTP virus filters through use of squid guard.		
Compliance	The proxy server should do not let you download the EICAR file from an outsider location.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			
Findings:			
Results:	☐ Pass ☐ Fail		

3 Fieldwork: Conducting the Audit

3.1 Audit Strategy

Before proceeding with the audit, an understanding as to why the selected checklist items were chosen is in order. In section 1, the top risks were identified by a quantitative process of weight the threat, vulnerability, and consequence of potential risks against the proxy server. Generally speaking, these risks can be reduced to the following concept:

> *Relying upon improper configurations or maintenance of the operating system and/or proxy software, an employee can successfully compromise the proxy server's availability (to download), confidentiality (of web transactions) or integrity of transactions through the proxy server.*

Administrative (3) was selected because proper change control is crucial to verification that proper settings are established and maintained through the authorized changes that will occur during the lifespan of the server. In conjunction with this, Hardening (6) will allow for the potential discover of attacks or unauthorized modifications to the server. Hardening (8) focuses upon the use of effective log analysis to detect abnormalities not only in the system via syslog, but also in the transactions conducted through the server. Hardening (9) allows for some clemency with regard to the administrator's knowledge of hardening strategies and current known risks. Using current vulnerability assessment tools will not only catch potential vulnerabilities that the administrator may otherwise miss, but it also will capture many of the other checklist items which were omitted due to the parameters of this paper. The squid checklist items are included because this audit is of a squid proxy server and as such it is the underlying service we seek to audit. Squid (2-4) seek to verify numerous configurations which impedes abuse of the software. Given the greater threat posed by employees who are onsite, checklist items Physical (1) and (2) are included to highlight the risks associated with physical access.

3.2 Audit Checklist Results

Item Number:	Administrative-03
Objective:	Change management documentation must exist for all critical infrastructure systems.
Reference:	Industry Best Practice, Personal BKM

Risk:	With proper documentation detailing changes to the system the ability to discern legitimate changes from unauthorized changes is called into question.		
Test:	1. Ask to see administrative documentation for system (hard copy or online) 2. Question administrators with regard to change procedures for proxy server.		
Compliance Critieria	1. Documentation will cover: ➢ Original baseline configuration ➢ Any authorized changes to the system ➢ Any detected unauthorized changes to the system, and the steps that were taken to re-secure the system 2. Administrators will demonstrate knowledge of documentation and where to find documentation in case of need.		
Test Nature:	✓ **Objective** ☐ Subjective	☐ Intrusive ✓ **Passive**	☐ Setting ✓ **Behavior**
Evidence	The administrators had in their possession a bound notebook which was used to 'create' the proxy server. The book contained hand-written details on the process for establishing the original proxy server. This book is a very loosely organized document with no real formal change control implemented. There is no change control documentation. There is a book used for incident handling which adheres to many of the BKM's relayed in the SANS Incident Handling & Hacker Techniques course.[19]		
Findings:	While the book provided clear directions how to technically establish a proxy server, as well as the tools to be run against it before it is to land on the production network, the administrators have no sound method of differentiating between an authorized change to the system from an unauthorized modification.		
Results:	☐ Pass ✓ **Fail**		

Item Number:	Hardening-03
Objective:	Disable un-necessary services
Reference:	Securing Linux step by Step CIS Level-1 Benchmark and Scoring Tool for Linux". Laude
Risk:	Un-necessary services result in additional unjustifiable exposure to potential unauthorized access or denial-of-service.

[19] The class recommends a notebook for journaling incidents – a contiguous bound book with sequentially numbered pages. They also leave no whitespace, and initial their entries.

<table>
<tr><td>Test:</td><td>9. Verify none of the following services are configured through xinetd: (finger, ntalk, rexec, rlogin, rsh, telnet, talk)
<code>/bin/ls /etc/xinetd.d/*</code>
10. Review which services are started as daemons on system:
<code>/bin/grep -i "disable" /etc/xinetd.d/*</code>
11. Verify only necessary services are run at boot:
<code>/path/to/chkconfig -list | /bin/grep ":on"</code>
12. Verify there are no un-ncessary services listening
<code>/bin/netstat -anp</code>
13. There are no print services running on the system:
<code>/bin/ps -ef | grep lpd</code>
14. There are no web servers running on the sytem:
<code>/bin/ps -ef | grep httpd</code>
15. NFS is not installed on the system:
<code>/bin/rpm -qa | grep knfsd</code>
16. Sendmail is listening only to localhost.
<code>telnet proxy_server 25</code>
17. Port scan the system to verify which services are listening</td></tr>
<tr><td>Compliance</td><td>9. /etc/xinetd.d does not contain any of the following files: finger, ntalk, rexec, rlogin, rsh, rexec, telnet, talk
10. Only services necessary are enabled are required[20]
11. For any service which is returned, verify that it is critical to run the server
12. The output of an lsof should reveal only services which are expected and critical to the operation of the system
13. The output of netstat should reveal only services which are open by design and critical to the operation of the system.
14. Print services should not be running
15. Web services should not be running
16. NFS should not be running
17. sendmail on port 25 should not respond to remote connections.</td></tr>
<tr><td>Test Nature:</td><td>✓ Objective
☐ Subjective</td><td>✓ Intrusive
☐ Passive</td><td>✓ Setting
✓ Behavior</td></tr>
<tr><td>Evidence</td><td colspan="3">1. <code>/bin/ls /etc/xinetd.d/*</code>
 

2. <code>/bin/grep -i "disable" /etc/xinetd.d/*</code></td></tr>
</table>

[20] Services known to be un-necessary according to industry best practices:
apmd, autofs, gpm, innd, IrDA, isdn, kdcrotate, lpd, lvs, mars-nwe, named, netfs, nfs, nfslock, oki4daemon, portmap, routed, rstatd, rusersd, rwalld, rwhod, sendmail, smb, snmpd, webmin, ypbind, ypserv, yppasswdd

```
# /bin/grep -i "disable" /etc/xinetd.d/*
/etc/xinetd.d/apass:    disable = no
/etc/xinetd.d/chargen:  disable         = yes
/etc/xinetd.d/chargen-udp:      disable         = yes
/etc/xinetd.d/csdd:     disable = no
/etc/xinetd.d/daytime:  disable         = yes
/etc/xinetd.d/daytime-udp:      disable         = yes
/etc/xinetd.d/echo:     disable         = yes
/etc/xinetd.d/echo-udp: disable         = yes
/etc/xinetd.d/finger:   disable         = yes
/etc/xinetd.d/ntalk:    disable                 = yes
/etc/xinetd.d/rexec:    disable = no
/etc/xinetd.d/rlogin:   disable = no
/etc/xinetd.d/rsh:      disable = no
/etc/xinetd.d/rsync:    disable = yes
/etc/xinetd.d/talk:     disable                 = yes
/etc/xinetd.d/telnet:   disable = no
/etc/xinetd.d/tftp:     disable                 = yes
/etc/xinetd.d/time:     disable         = yes
/etc/xinetd.d/time-udp: disable         = yes
/etc/xinetd.d/wu-ftpd:  disable = no
#
```

3. `/sbin/chkconfig –list | /bin/grep ":on"`

```
# /sbin/chkconfig --list | /bin/grep ":on"
keytable        0:off   1:on    2:on    3:on    4:on    5:on    6:off
privoxy         0:off   1:off   2:off   3:on    4:on    5:on    6:off
microcode_ctl   0:off   1:off   2:on    3:on    4:on    5:on    6:off
xfs             0:off   1:off   2:off   3:off   4:on    5:off   6:off
gpm             0:off   1:off   2:off   3:off   4:on    5:off   6:off
nscd            0:off   1:off   2:off   3:off   4:on    5:off   6:off
netfs           0:off   1:off   2:off   3:off   4:on    5:off   6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
random          0:off   1:off   2:on    3:on    4:on    5:on    6:off
rawdevices      0:off   1:off   2:off   3:on    4:on    5:on    6:off
portmap         0:off   1:off   2:off   3:off   4:on    5:off   6:off
rhnsd           0:off   1:off   2:off   3:off   4:on    5:off   6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
sendmail        0:off   1:off   2:on    3:on    4:on    5:on    6:off
anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
apmd            0:off   1:off   2:off   3:off   4:on    5:off   6:off
atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
sshd2           0:off   1:off   2:off   3:on    4:on    5:on    6:off
pcmcia          0:off   1:off   2:off   3:off   4:on    5:off   6:off
nfslock         0:off   1:off   2:off   3:off   4:on    5:off   6:off
ntpd            0:off   1:off   2:off   3:on    4:on    5:on    6:off
rc.local        0:off   1:off   2:off   3:on    4:on    5:on    6:off
rc.once         0:off   1:off   2:off   3:on    4:on    5:on    6:off
acct            0:off   1:off   2:off   3:on    4:on    5:on    6:off
squid           0:off   1:off   2:off   3:on    4:off   5:on    6:off
idsmc           0:off   1:off   2:off   3:on    4:on    5:on    6:off
#
```

4. /bin/netstat –anp | grep –I "listen"

```
# /bin/netstat -anp | grep -i "listen"
tcp        0      0 0.0.0.0:9090            0.0.0.0:*               LISTEN      10951/boa
tcp        0      0 0.0.0.0:554             0.0.0.0:*               LISTEN      723/rtspd
tcp        0      0 0.0.0.0:587             0.0.0.0:*               LISTEN      658/sendmail: accep
tcp        0      0 0.0.0.0:911             0.0.0.0:*               LISTEN      10985/(squid)
tcp        0      0 0.0.0.0:8118            0.0.0.0:*               LISTEN      10999/privoxy
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      796/sshd2
tcp        0      0 0.0.0.0:1080            0.0.0.0:*               LISTEN      2511/socks5
tcp        0      0 0.0.0.0:1081            0.0.0.0:*               LISTEN      26108/socks5-1081
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      658/sendmail: accep
tcp        0      0 0.0.0.0:6010            0.0.0.0:*               LISTEN      1194/sshd2
tcp        0      0 0.0.0.0:6011            0.0.0.0:*               LISTEN      6942/sshd2
tcp        0      0 0.0.0.0:9595            0.0.0.0:*               LISTEN      834/idsmc
tcp        0      0 0.0.0.0:6013            0.0.0.0:*               LISTEN      17380/sshd2
tcp        0      0 0.0.0.0:6014            0.0.0.0:*               LISTEN      18464/sshd2
#
```

5. /bin/ps –ef | grep lpd

```
# ps -ef | grep lpd
root      1135 32566  0 13:36 pts/3    00:00:00 grep lpd
#
```

6. /bin/ps –ef | grep httpd

```
# ps -ef | grep httpd
root      1160 32566  0 13:37 pts/3    00:00:00 grep httpd
#
```

7. `/bin/rpm –qa | grep knfsd`

```
# /bin/rpm -qa | grep knfsd
#
```

8. `telnet proxy_server 25`

```
$ telnet [redacted] 25
Trying 10.7.210.139...


$
```

9. Portscan the system

```
$ strobe 10.7.210.139
strobe 1.03 (c) 1995 Julian Assange (proff@suburbia.net).
10.7.210.139                    ssh                  22/tcp # SSH
Remote Login Protocol
10.7.210.139                    rtsp                554/tcp # Real
Time Stream Control Protocol
10.7.210.139                    unknown              911/tcp
unassigned
10.7.210.139                    socks               1080/tcp #
socks proxy server
$
```

Findings:

- Several un-necessary services were configured through xinetd which should be removed.
- Examining the output from chkconfig, we are interested in runlevel 3 (Full multi-user mode no GUI) which reveals that only essential services are started during the final state of the system.
- Not only were service files established in xinetd.d/ for unnecessary services, some were not disabled – apass, csdd, wu-ftpd . Of greatest concern was the ability to enable the r-commands (rexec, rsh, rlogin) and telnet.
- The netstat demonstrated a limited number of open connections. All services which were listening were necessary services.
- The additional services of concern (lpd, http, nfs and sendmail listening for

	remote connections) were all disabled. • The portscan revealed no additional services listening beyond those require for remote administration and those services offered by the proxy server.
Results:	✓ **Pass** ☐ Fail

Item Number:	Hardening-06		
Objective:	An Intrustion Detection System must be installed which routinely (daily) verifies the integrity of binaries on the system.		
Reference:	Personal Experience.		
Risk:	Un-detected attacks against a system can provide the attacker and extended opportunity to compromise the services.		

Test:	4. Verify that tripwire is installed on the system:
	`/usr/bin/find / -name tripwire -print`
	5. Verify tripwire is automatically initiated
	`/usr/bin/crontab –u root -l`
	6. Examine tripwire configuration files		

Compliance	4. Tripwire is installed on the system. 5. Tripwire should be in a daily (or more frequent) cron 6. Tripwire should be configured to track modifications to key programs often modified by attackers:		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	☐ Setting ✓ Behavior

Evidence	`/usr/bin/find / -name tripwire –print`
	
	`Tripwire was located at- /usr/sbin/tripwire.`
	`Relying on the –version to discover which version we uncoverd the following:`		

```
# /usr/sbin/tripwire --version
Tripwire(R) 2.3.0.47 for Linux

The developer of the original code and/or files is Tripwire, Inc.  Portions
created by Tripwire, Inc. are copyright 2000 Tripwire, Inc. Tripwire is a
registered trademark of Tripwire, Inc.  All rights reserved.

This program is free software.  The contents of this file are subject to the
terms of the GNU General Public License as published by the Free Software
Foundation; either version 2 of the License, or (at your option) any later
version.  You may redistribute it and/or modify it only in compliance with
the GNU General Public License.

This program is distributed in the hope that it will be useful.  However,
this program is distributed "AS-IS" WITHOUT ANY WARRANTY; INCLUDING THE
IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
Please see the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with
this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Nothing in the GNU General Public License or any other license to use the
code or files shall permit you to use Tripwire's trademarks, service marks,
or other intellectual property without Tripwire's prior written consent.

If you have any questions, please contact Tripwire, Inc. at either
info@tripwire.org or www.tripwire.org.
#
```

`/usr/bin/crontab -u root -l`

```
# crontab -l | grep tripwire
#
```

Tripwire configuration – see appendices.

Findings:

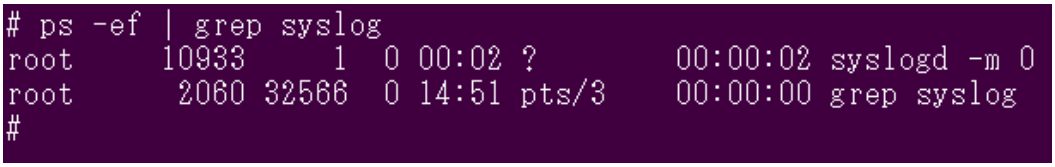
- Tripwire was found to be on the machine and configured. A tripwire database was found on the local machine indicating it had been run in the past. Uncertain of the version, we ran tripwire with the –version switch to discover this is an older, freeware version.
- There was no entry in the crontab file to suggest that this program was executed automatically at some specific interval. Lack of such a procedure was verbally verified by the system administrator.
- The tripwire configuration demonstrated that some care was taken in establishing the directories to watch. Specific attention was given to those directories which held binaries often tampered with, or replaced by, a rootkit.

Results:

☐ Pass
✓ **Fail**

Item Number:	Hardening-08		
Objective:	System logs are established, reviewed, and rotated 30 days of logs left on system		
Reference:	Corporate policy SANS Security Essentials (Track 1)		
Risk:	The absence of logging enhances the ability of a threat to compromise a system and aggravates any post-incident analysis.		

Test:	7. Verify that logging is enabled.
	`/bin/ps –ef | grep syslog`
	8. Verify that syslog is not only stored locally, but also sent to a central log service.
	`/bin/grep "@" /etc/syslog.conf`
	9. Verify log rotation is executed and retained for 30 days
	`/bin/cat /etc/logrotate.conf`
	10. Verify logs are analyzed and admin aware
	11. Verify that log files are analyzed periodically (daily)		

Compliance	7. Syslog should be running on the system 8. In the /etc/syslog.conf file there should exist an entry sending relevant (or all) syslog entries to: @you_central_log_server. 9. Logs are to be rotated daily, and 30 days should be retained online. Look in the logrotate.conf file and verify that for each file that relevant information is logged to, that the following parameters exist either for the individual files or as a global setting: 10. Ask administrator to demonstrate how important log events are detected and addressed. 11. Ask administrator to demonstrate that squid access and error logs are analyzed through an automated script. Logs should be stored according to corporate policy (30 days online, daily rotate)		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence	1. `/bin/ps –ef | grep syslog`  2. /bin/grep "@" /etc/syslog.conf		

```
# /bin/grep "@" /etc/syslog.conf
*.*                                                    @log01.xxx.yyy.com
#
```

3. `/bin/cat /etc/logrotate.conf`

```
# /bin/cat /etc/logrotate.conf
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# send errors to root
errors root

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own lastlog or wtmp -- we'll rotate them here
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}

# system-specific logs may be configured here
#
```

4. The administrator could demonstrate that log information was received in periodic emails received by them containing log entries which were raised to be examined via the freeware program *logcheck*

5. Verify squid logs are analyzed in accordance with corporate acceptable use policy

Findings:

- ✓ Syslog is running on the system.
- ✓ The system is configured to route all syslog information to a central log server (which was sterilized in the screen capture).
- ✓ The log rotation configuration demonstrates 28 days of logs which are rotated weekly. Further examination in the logrotate.d directory did not discover any further file level configurations which demonstrated compliance with corporate policy but the remote log server does retain

	information for 30+ days. ✓ Log analysis was conducted every 4 hours, with specific emails being received by key administrators with subject lines that distinguished days without events from those with. In the event of a "system attack", the script would email the administrator's pagers. ✓ Acceptable use filters were deployed which would highlight web surfing of sites with inappropriate content.
Results:	✓ **Pass** ☐ Fail

Item Number:	Hardening-09		
Objective:	The system will report no High or Medium vulnerabilities when subjected to vulnerability assessment tools.		
Reference:	Personal Experience		
Risk:	The risk associated with running a machine increases over time if the machine is not routinely assessed for new vulnerabilities. Timely scans with Vulnerability Assessment tools diminishes this risk.		
Test:	Execute the following vulnerability scans against the system ✓ Nessus scan ✓ CIS Security Tool		
Compliance	No tool should return any medium or high vulnerabilities that are not documented as known risks which have been accepted or mitigated through other measures.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence	Nessus Scan Results:  		

unknown (911/tcp)	High	The remote squid caching proxy, according to its version number, is vulnerable to various buffer overflows. An attacker may use these to gain a shell on this system.

ssh (22/tcp)	High	According to its banner, the remote SSH server is vulnerable to one or more of the following vulnerabilities: CAN-2002-1357 (incorrect length) CAN-2002-1358 (lists with empty elements/empty strings) CAN-2002-1359 (large packets and large fields) CAN-2002-1360 (string fields with zeros) Some of these vulnerabilities may allow remote attackers to execute arbitrary code with the privileges of the SSH process, usually root. Solution : Upgrade your SSH server to an unaffected version Risk factor : High CVE : CAN-2002-1357, CAN-2002-1358, CAN-2002-1359, CAN-2002-1360

*** CIS Ruler Run ***
Starting at time 20040823-11:19:47

<See appendix for details>

Preliminary rating given at time: Mon Aug 23 11:19:52 2004

Preliminary rating = 6.61 / 10.00

Ending run at time: Mon Aug 23 11:19:55 2004

Final rating = 6.61 / 10.00

Findings:

The two high findings uncovered by nessus were (in)valid.

Referring to the squid advisories (http://www.squid-cache.org/Advisories/) we discover the following:

SQUID-2004:2, **June 7, 2004**
Buffer overflow bug in 'ntlm_auth' authentication helper. Squid-2.5.STABLE6 addresses this bug.

SQUID-2004:1, **February 29, 2004**
Fixes and features for URL encoding tricks.
Squid-2.5.STABLE5 addresses these issues.

SQUID-2002:3, **July 3, 2002**
Security advisory several issues in Squid-2.4.STABLE6 and earlier.
Squid-2.4.STABLE7 released to address these issues.

Recovering the version from the system:

```
$ ./squid -v
Squid Cache: Version 2.4.STABLE6
$
```

We now know that this system is vulnerable (by version) to all of these

	advisories. None of these vulnerabilities risk the proxy server itself; they can instead impact the web/ftp services directed through the service. Since this is an area of high risk (according to your initial risk assessment), these vulnerabilities are significant to this audit.
	
	Examinig the ssh binaries on the proxy server, we discover:
	`[root@proxy bin]$ ssh -v`
	`ssh: F-Secure-SSH-2.3.1 (build 7.afs) on i686-pc-linux-gnu`
	
	Referring to the above advisories (CAN 2002-1357 – 1360) and the Fsecure website, we discover that none of these risks applied to this client software, so the HIGH reported through Nessus is a false positive.
	
	The low score (6.6) on the CIS test indicated several issues with the current configuration of the system which were important to note:
	1. xinetd should be configured with an “only-from” statement thereby restricting access to the services it is providing.
	2. the machine’s sendmail is set to receive email – this is not the purpose of the system.
	3. /proc/sys/net/ipv4/tcp_max_syn_backlog should be at least 4096 to handle SYN floods.
	4. /proc/sys/net/ipv4/conf/eth0/send_redirects should be 0 to disable outgoing redirect messages.
	5. /proc/sys/net/ipv4/conf/lo/send_redirects should be 0 to disable outgoing redirect messages.
	6. /proc/sys/net/ipv4/conf/default/send_redirects should be 0 to disable outgoing redirect messages.
	7. rhosts authentication not deactivated in /etc/pam.d/rexec.
	8. rhosts authentication not deactivated in /etc/pam.d/rlogin.
	9. rhosts authentication not deactivated in /etc/pam.d/rsh.
	10. rhosts authentication not deactivated in /etc/pam.d/rexec.dist.
	11. rhosts authentication not deactivated in /etc/pam.d/rlogin.dist.
	12. rhosts authentication not deactivated in /etc/pam.d/rsh.dist
	13. Numerous system accounts have no shell in /etc/passwd, which defaults to /bin/sh
	14. Shadow passwords are not enabled.

Results:	☐ Pass ✓ **Fail**

Item Number:	Network-01
Objective:	Enable security settings on the system to minimize risks associated with

	attacks that rely upon packet manipulation.
Reference:	Beauman, Sean "Auditing a Linux FTP and DNS Server: And Administrators Perspective". GSNA Practical. Sept 20 2003 Mourani, Gerhard. "Securing and Optimizing Linux: RedHat Edition." OpenDocs, LLC. 2000. URL: http://www.tldp.org/LDP/solrhe/Securing-Optimizing-Linux-RH-Edition-v1.3 (September 19, 2003)
Risk:	Improper network configurations on the system can result compromise, denial of service attacks against the system, or in the utilization of the system in a denial of service against other systems or network services.

Test:	5. Log illegal packets (spoofs, source routing, redirects)
	`/bin/grep "log_martians" /etc/sysctl.conf`
	6. Prevent SYN floods. Examine /etc/sysctl.conf
	`/bin/grep "syncookies" /etc/sysctl.conf`
	7. Prevent routing table alterations via ICMP redirects
	`/bin/grep "accept_redirects" /etc/sysctl.conf`
	8. Enforce fragmentation protection to prevent frag overlaps or exploits
	`/bin/grep "always_defrag" /etc/sysctl.conf`

Compliance	For each of the above grep statements, you should yield the following results: 5. net.ipv4.iconf.all.log_martians = 1 6. net.ipv4.tcp_syncookies = 1 7. net.ipv4.conf.all.accept_redirects = 0 8. net.ipv4.ip_always_defrag = 1
Test Nature:	✓ **Objective** ☐ Subjective | ✓ **Intrusive** ☐ Passive | ✓ **Setting** ☐ Behavior

Evidence	`# /bin/grep "log_martians" /etc/sysctl.conf`
	`# /bin/grep "syncookies" /etc/sysctl.conf`
	`# /bin/grep "accept_redirects" /etc/sysctl.conf`
	`# /bin/grep "always_defrag" /etc/sysctl.conf`
	`# /bin/cat /etc/sysctl.conf`
	`# Disables packet forwarding`
	`net.ipv4.ip_forward = 0`
	`# Enables source route verification`
	`net.ipv4.conf.all.rp_filter = 1`
	`# Disables the magic-sysrq key`
	`kernel.sysrq =1`
	`# We need more aggressive keepalive intervals due to the DMZ firewall's`
	`# default timeout of under 1 hour.  Set it for 30 minutes.`
	`net.ipv4.tcp_keepalive_time = 1800`
	`#`

Findings:	None of the recommended network configurations were present.
Results:	☐ Pass ✓ **Fail**

Checklist Item:	Physical-01
Objective:	The proxy server will reside in a location designed to reduce risks associated with physical access to the system.

References:	"If a bad guy has unrestricted physical access to your computer, it's not your computer anymore" - 10 Immutable Laws of Security Corporate policy regarding minimum security standards for data centers.		
Risk:	The major threats from physical access are denial-of-service and inappropriate access to the equipment. Physical access means the system can (potentially) be physically disabled, data on the permanent media storage devices stolen, or the box can be rebooted and administrative access obtained.		
Test:	Is system located in a restricted access data center? Building access is monitored Room access is monitored Room access granted through ACL ACL is reviewed periodically All entrances to room are locked		
Compliance Criteria	9. All entrances to building are locked 10. All entrances to building are monitored 11. Entrances to room are authenticated 12. Entrances to room are monitored 13. Cameras view all entrances monitored by security services 24x7 14. Doors left open for extended period of time are responded to by site security. 15. Room access granted through ACL 16. ACL is reviewed periodically		
Test Nature:	✓ **Objective** ☐ Subjective	✓ **Intrusive** ☐ Passive	✓ **Setting** ☐ Behavior
Evidence	Photographing this area to demonstrate evidence is prohibited by corporate policy. A walk-through was arranged as well as a conversation with site security.		
Findings:	The system was sufficiently protected by a defense in depth approach to security. Chokepoints allowing entry into the building and into the room where the server resides are either monitored by personnel or have badge-authenticated turnstiles to prevent tailgating. The doors require authentication by possession (badge) and are monitored from a central monitoring center. Access is reviewed quarterly by the room owner.		
Results:	✓ **Pass** ☐ Fail		

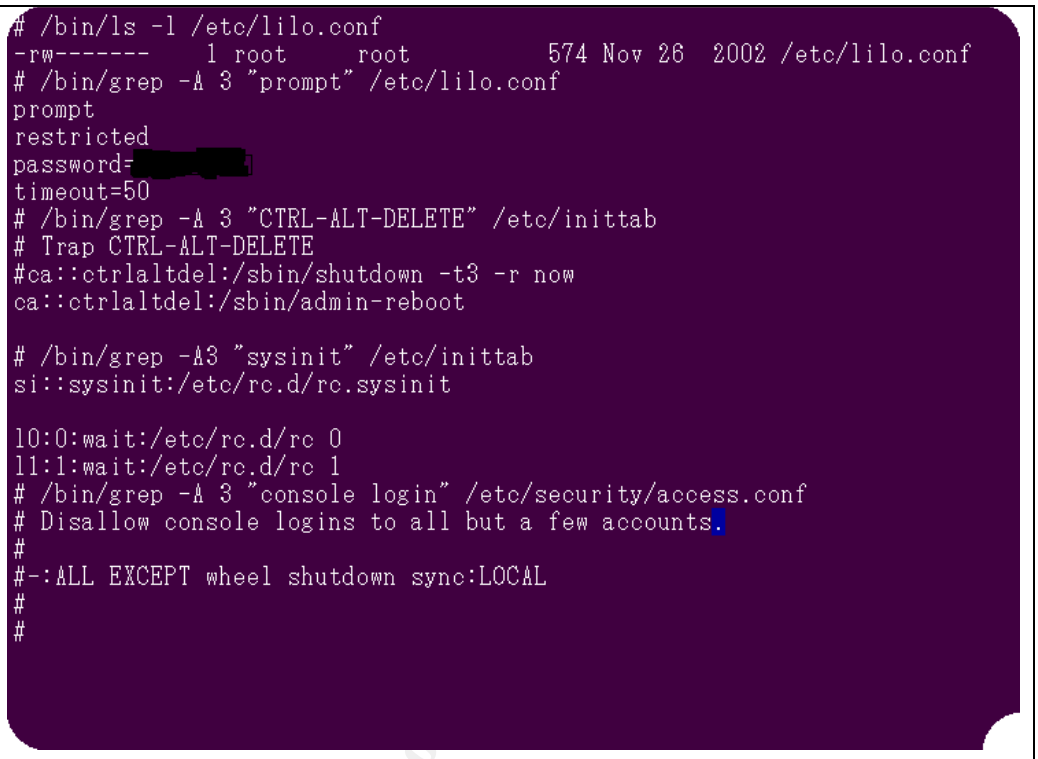
Checklist Item:	Physical-04
Objective:	The operating system will be configured so as to reduce risks associated with physical access to the system.
References:	"If a bad guy has unrestricted physical access to your computer, it's not your computer anymore" - 10 Immutable Laws of Security

Risk:	Physical access means the system can (potentially) be physically disabled, data on the permanent media storage devices stolen, or the box can be rebooted and administrative access obtained.		

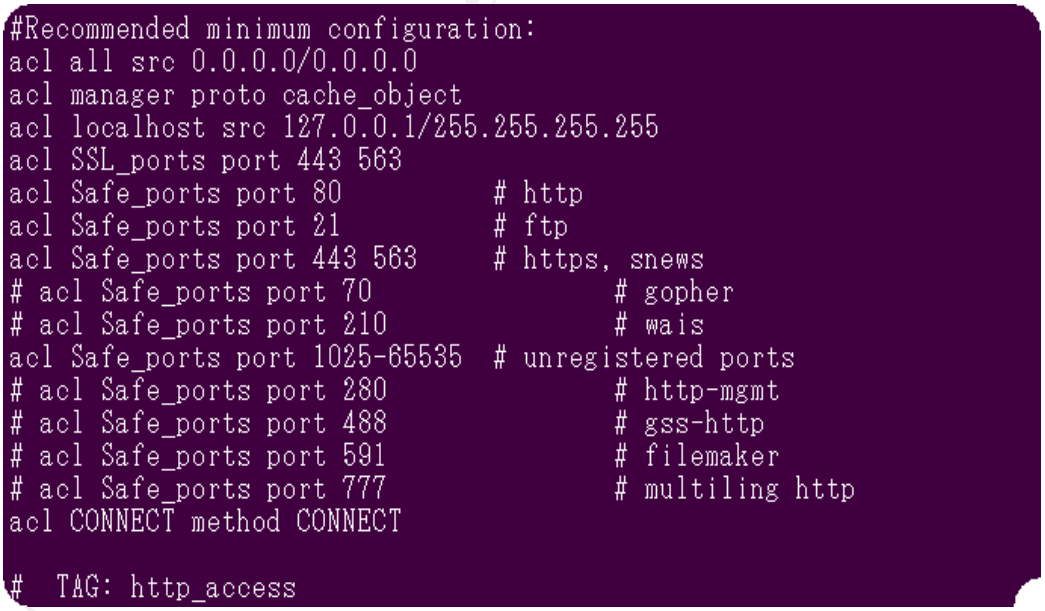
Test:	8. Access to lilo.conf file is restricted to root.
	`/bin/ls –l /etc/lilo.conf`
	9. The LILO prompt password protected.
	`/bin/grep –A 3 "prompt" /etc/lilo.conf`
	10. Reboot from console w/ Ctrl+Alt+Del is disabled.
	`/bin/grep –A 3 "CTRL-ALT-DELETE" /etc/inittab`
	11. Root password is required to enter single user mode
	`/bin/grep –A 2 "sysinit" /etc/inittab`
	12. Console logins restricted to root and authorized users
	`/bin/grep –A 3 "console login" \`
	`/etc/security/access.conf`
	Attempt to login at console w/ non-authorized account.
	13. Screensaver in enabled and autolocks after 15 minutes idle
	Allow system to remain idle – verify screen saver engages
	14. Screensaver requires password of console logged in user to unlock.
	Attempt to unlock screensaver w/o password. Then with password.		

Compliance Criteria	8. /etc/lilo.conf will be owned by root with permissions set to 600
	9. The 2 directives following the prompt directive will be:
	`password = <Your_LILO_Password>`
	`restricted`
	10. The directive allowing this has been commented out.
	`# Trap CTRL-ALT-DELETE`
	`#ca::ctrlaltdel:/sbin/shutdown -t3 -r now`
	11. The *wait* directive has been added below the *sysinit* directive.
	`# System initialization.`
	`si::sysinit:/etc/rc.d/rc.sysinit`
	`~~:S:wait:/sbin/sulogin`
	12. The line restricting access will be uncommented:
	# Disallow console logins to all but a few accounts.
	#
	-:ALL EXCEPT wheel shutdown sync:LOCAL
	13. Screensaver should enable in 15 minutes.
	14. Only password that should unlock screen is that of the user logged in on the console.		

Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			

	 System was examined physically. Administrator logged in at console and we waited – as expected the screen lock engaged and could not be opened except by the administrator using his password.
Findings:	While many of the potential controls were in place, the system did allow for a reboot of the system through the keyboard (CTRL-ALT-DELETE) and also permitted any valid account to login from keyboard.
Results:	☐ Pass ✓ Fail

Item Number:	Squid-02
Objective:	Implement access control lists (ACL) to restrict access to server by trusted network ranges only.
Reference:	Galarneau
Risk:	Open network settings provide attackers unfettered access to system and allow unauthorized internal users to utilize proxy services.
Test:	4. Only trusted addresses should be allowed to connect to the proxy server. 5. Configuration should default to deny that which is not explicitly allowed.

	6. Only necessary ports should be opened by proxy services.		
Compliance	4. Networks allowed to connect through the proxy server should be explicitly enumerated (example): *Acl all src 0.0.0.0/0.0.0.0* *Acl offices src 10.7.0.0/255.255.0.0* *http_access allow offices* *Acl labs src 10.8.0.0/255.255.0.0* *http_access allow labs* *http_access deny all* 5. The last acl should be: *http_access deny all* 6. The acceptable ports should be enumerated and all others should be denied (example): *Acl trusted_ports port 21 80 443* *http_access deny !trusted_ports*		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ☐ Behavior
Evidence			
Findings:	There are insufficient ACL controls in place on the server with regard to which IP addresses may access the system. The configuration file lacks the deny all default. Ports are properly controlled through acls allowing only essential services and high end ports used for socks connections.		
Results:	☐ Pass ✓ **Fail**		

Item Number:	Squid-04		
Objective:	Implement squid configurations (squid.conf) to reduce risk of denial of service attacks against the service.		
Reference:	Squid web site http://www.xatrix.org/print1312.html		
Risk:	Several configurations in squid allow for protection against over-allocation of resources which could starve legitimate use.		
Test:	Examine the following settings in the squid.conf file: 10. `maximum_object_size` 11. `quick_abort_min` 12. `quick_abort_max` 13. `quick_abort_pct` 14. `dns_nameservers` 15. `ignore_unknown_nameservers`[21] 16. `client_lifetime`[22] 17. `pconn_timeout`[23] 18. `request_header_max_size`[24]		
Compliance	10. `maximum_object_size` <= 4096 11. `quick_abort_min` = 16KB 12. `quick_abort_max` = 16KB 13. `quick_abort_pct` = 95 14. Verify nameservers in /etc/resolv.conf are trusted – if so, then this value should not be set. If they are not trusted, this should eb set to trusted servers. Name servers used by squid must come from trustable sources and configured safely. A compromised DNS server is often used by attackers to divert proxy servers and certain Squid versions can be crashed by sending malformed DNS answers 15. should be enabled or trusted. 16. client lifetime should be <= 1 day 17. pconn_timeout <= 120 18. value should be <= 10 kb.		
Test Nature:	✓ Objective ☐ Subjective	✓ Intrusive ☐ Passive	✓ Setting ✓ Behavior
Evidence			

[21] This option verifies if a nameserver answering the lookup has the same IP address as the one the lookup was sent to.
[22] The client_lifetime sets the maximum time a client is allowed to be bound to a Squid process
[23] The pconn_timeout sets the maximum time an *idle* client is allowed to be bound to a squid process.
[24] The request_header_max_size option is used to limit the size of acceptable HTTP headers.

```
# grep maximum_object_size squid.conf
#  TAG: maximum_object_size     (bytes)
maximum_object_size 4096 KB
#  TAG: maximum_object_size_in_memory   (bytes)
maximum_object_size_in_memory 8 KB
#       the value of maximum_object_size above its default of 4096 KB to
# grep quick_abort squid.conf
#  TAG: quick_abort_min (KB)
#  TAG: quick_abort_max (KB)
#  TAG: quick_abort_pct (percent)
#       quick_abort values to the amount of data transfered until
#       If the transfer has less than 'quick_abort_min' KB remaining,
#       it will finish the retrieval.  Setting 'quick_abort_min' to -1
#       will disable the quick_abort feature.
#       If the transfer has more than 'quick_abort_max' KB remaining,
#       If more than 'quick_abort_pct' of the transfer has completed,
quick_abort_min 16 KB
quick_abort_max 16 KB
quick_abort_pct 95
#       request_timeout, pconn_timeout and quick_abort values.
# grep dns_nameservers squid.conf
#  TAG: dns_nameservers
#       Example: dns_nameservers 10.0.0.1 192.172.0.4
#
```

```
#  TAG: ignore_unknown_nameservers
#       By default Squid checks that DNS responses are received
#       from the same IP addresses that they are sent to.  If they
#       don't match, Squid ignores the response and writes a warning
#       message to cache.log.  You can allow responses from unknown
#       nameservers by setting this option to 'off'.
#
#Default:
# ignore_unknown_nameservers on

#  TAG: digest_generation
```

```
#  TAG: client_lifetime time-units
#       The maximum amount of time that a client (browser) is allowed to
#       remain connected to the cache process.  This protects the Cache
#       from having a lot of sockets (and hence file descriptors) tied up
#       in a CLOSE_WAIT state from remote clients that go away without
#       properly shutting down (either because of a network failure or
#       because of a poor client implementation).  The default is one
#       day, 1440 minutes.
#
#       NOTE:  The default value is intended to be much larger than any
#       client would ever need to be connected to your cache.  You
#       should probably change client_lifetime only as a last resort.
#       If you seem to have many client connections tying up
#       filedescriptors, we recommend first tuning the read_timeout,
#       request_timeout, pconn_timeout and quick_abort values.
#
#Default:
# client_lifetime 1 day

#  TAG: half_closed_clients
```

```
#  TAG: pconn_timeout
#       Timeout for idle persistent connections to servers and other
#       proxies.
#
#Default:
# pconn_timeout 120 seconds
```

```
#  TAG: request_header_max_size (KB)
#       This specifies the maximum size for HTTP headers in a request.
#       Request headers are usually relatively small (about 512 bytes).
#       Placing a limit on the request header size will catch certain
#       bugs (for example with persistent connections) and possibly
#       buffer-overflow or denial-of-service attacks.
#
#Default:
# request_header_max_size 10 KB

#  TAG: request_body_max_size   (KB)
```

```
#  TAG: delay_pools
#       This represents the number of delay pools to be used.  For example,
#       if you have one class 2 delay pool and one class 3 delays pool, you
#       have a total of 2 delay pools.
#
#       To enable this option, you must use --enable-delay-pools with the
#       configure script.
#
#Default:
# delay_pools 0
```

Findings:

- The maximum_object_size is configured in such a manner as to control the ability of denial of service attacks to overload the proxy server through requests for extremely large content. Similarly, the request_header_max_size is also set to reduce the risk of large content

	denial of service strategies. • Additional configurations of the quick abort features have been enabled to contribute to the overall squid configuration strategy to limit denial of service attacks against the system. • Client lifetime is configured in accordance with the business goal of 1 day. • DNS is served to this system through trusted corporate services and as such there is no need for the ignore_unknown_nameservers setting
Results:	✓ **Pass** ☐ Fail

4 Audit Report

4.1 Executive Summary

The results of this audit are a mixed assortment of positive and negative findings. Reasonably competent technical work was put into landing this server in a secure state in June of 2002. Many industry best known methods were employed and the machine was assessed with vulnerability scanners to close most of the gaps known in 2002. It is evident to this auditor that a sound defense in depth strategy was employed when landing this system in the form of several network appliance configurations as well as monitoring devices that were installed on the squid serve. Administrative procedures, on the other hand, were completely absent from all their processes. As is often the case, the administrators for this server are also responsible for an extensive number of systems and have growing demands on their time which allowed the system's security posture to degrade over time. The squid software itself is at least a year out of date and the machine does not appear to have been patched since it was established. Through a combination of strong initial settings and network acls, the machine has remained intact to date. (Or as far as this auditor can discern)

4.2 Audit Findings

4.2.1 Introduction

As detailed in earlier sections, the greatest potential risks assigned to this system would come from employees who would rely upon poor maintenance and/or configuration settings to compromise the integrity of the server and/or the availability, confidentiality, or integrity of the web services the proxy provides. Given this defined high risk, we selected checklist items which would best test the controls in place for this risk. Several areas of examination were selected to provide a cross section of auditable items. The following sections detail the results of those areas.

This audit was driven by a desire to reduce risk for the squid proxy server which is used by a business unit within a larger parent corporation. The principle use for this proxy is to provide web access for both web browsing and the download of files from the internet that are relevant to the work done by the business unit. As such, while casual web surfing is not considered a business goal, the ability to conduct certain types of transaction as well as download relevant files is.

4.2.2 Physical Audit

The physical audit involved an on-site examination of the controls which were put into place to reduce risk associated with physical access to the system. Corporate policy prohibits the capturing of such security measures on any media such as photographs. Equipped with the checklist designed in part 2, the on site location was checked for the 8 criteria established. Many of these criteria were found to exist in the corporate physical security specifications for data centers and are incorporated into all new physical sites. As a result, it was not surprising to discover that sufficient controls were in place to prevent unauthorized physical access to the system. Numerous checkpoints were established between the exterior of the building and the physical location of the system which required a successful response to an authentication challenge. Access controls tied to these authentication systems are reviewed periodically. The entrances are also monitored for unauthorized accesses. Everyone with access to these rooms are also required to take specific training courses designed to help reduce the risk of social engineering or a breach of these controls via "tail gaiting" on an authorized person's credentials.

Security controls on the system itself were implemented to reduce risk from those who obtain physical access to the system. We examined configurations to the restricted access lilo.conf file which verified that the system was protected by a lilo boot password. Given this system is in production, rebooting the system to verify these settings in action was prohibited due to the potential impact it could have on the business goals this system supports.

Table 4.1: Screen shot of lilo.conf settings

```
# /bin/ls -l /etc/lilo.conf
-rw-------    1 root     root          574 Nov 26  2002 /etc/lilo.conf
# /bin/grep -A 3 "prompt" /etc/lilo.conf
prompt
restricted
password=[redacted]
timeout=50
# /bin/grep -A 3 "CTRL-ALT-DELETE" /etc/inittab
# Trap CTRL-ALT-DELETE
#ca::ctrlaltdel:/sbin/shutdown -t3 -r now
ca::ctrlaltdel:/sbin/admin-reboot

# /bin/grep -A3 "sysinit" /etc/inittab
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
# /bin/grep -A 3 "console login" /etc/security/access.conf
# Disallow console logins to all but a few accounts.
#
#-:ALL EXCEPT wheel shutdown sync:LOCAL
#
#
```

We could however validate that the screen saver did engage after 15 minutes of idle time and would only unlock for the user logged into the system currently who was challenged for their password.

The areas where this system failed to achieve sufficient control was in the potential use of CTRL-ALT-DELETE to force the system to reboot. By examining the /etc/inittab setting we determined that this hotkey would initiate a shutdown of the system. This gap in their controls could allow for a potential attack upon the availability of the system. Further examination is called for to verify that the cd drive was not a bootable device and that a hard drive password was established. Both of these controls would significantly help mitigate the risk involved with the ability to reboot the system from the keyboard from an attack on integrity perspective.

4.2.3 Administrative Procedures and Behaviors

Upon arrival, the system administrators were asked to deliver all relevant information that they had regarding the establishment and change control for the system. All administrators who were responsible for this system were aware of

the location of the creation documentation so in the case of an emergency those responsible for the system all knew where to access the relevant information. This request was answered with a single notebook which contained all notes relevant to the squid proxy server. The book contained the hardware and software required, as well as initial build and configuration settings for the system. As an aside, this book would provide reasonable instructions in case of a catastrophic loss – it had sufficient information with regard to corporate contacts and requirements that this auditor feels a system could be established without a great deal of trouble assuming the availability of hardware and current versions of the software.

Table 4.2: Example of Notebook Used

What the book lacked was any procedures for change control. There was no indication any changes had been made to the system or the configurations after it was established in 2002. As explained by one administrator, all configuration files were held under RCS control[25] and that all changes were thus documented on the system itself. This auditor finds this to be an unacceptable level of change control. Beyond the risk that an intruder could tamper with these files, they also would be lost in the case of a catastrophic loss. Copies of these files must be off the system as proper controls for the integrity of the content not to be in question.

The administrators were also questioned regarding the actions taken during a routine day to maintain awareness of the system and its settings. This inquiry focused initially on the use of an IDS to detect modifications to the system. The administrators could demonstrate that the freeware version of tripwire was present on the system and that a copy of the database was stored offline. the database and policies included all the relevant directories which store many of the critical binaries[26] which are often tampered with by a rootkit or traditional compromise. (A copy of the policy that was inspected is included in the

[25] This information was not validated as it was disclosed in the exit meeting.
[26] While there is an extensive list, commands such as ls, ps, finger, passwd, netstat, and other system-awareness binaries are choice targets for modification.

appendices). This initial good effort however was ineffectual since no periodic (daily) automated execution of this IDS was enabled on the system in such a manner that the administrators would be notified in case of problems.

Table 4.3; Evidence of IDS

```
# /usr/sbin/tripwire --version
Tripwire(R) 2.3.0.47 for Linux

The developer of the original code and/or files is Tripwire, Inc.  Portions
created by Tripwire, Inc. are copyright 2000 Tripwire, Inc. Tripwire is a
registered trademark of Tripwire, Inc.  All rights reserved.

This program is free software.  The contents of this file are subject to the
terms of the GNU General Public License as published by the Free Software
Foundation; either version 2 of the License, or (at your option) any later
version.  You may redistribute it and/or modify it only in compliance with
the GNU General Public License.

This program is distributed in the hope that it will be useful.  However,
this program is distributed "AS-IS" WITHOUT ANY WARRANTY; INCLUDING THE
IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
Please see the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with
this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Nothing in the GNU General Public License or any other license to use the
code or files shall permit you to use Tripwire's trademarks, service marks,
or other intellectual property without Tripwire's prior written consent.

If you have any questions, please contact Tripwire, Inc. at either
info@tripwire.org or www.tripwire.org.
#
```

While the administrators were lacking in good security practices with regard to the monitoring of the IDS software installed on the squid proxy server, this auditor did confirm that they were receiving the results of their log analysis software (into which the proxy server was sending its syslog information). Periodic (4 hour interval) emails are sent to the administrators – those that detect system attacks are forwarded immediately to their pagers via a telepage service. This suggests that they have the proper procedures in place and need to only re-enable automated tripwire executions to close the gap in this matter.

4.2.4 System Configurations/Behaviors

4.2.4.1 Unnecessary Services

The system was audited for the existence of un-necessary services through several checklist tests. Arguably the most secure option is to remove those services you do not require from xinetd and the accompany xinetd.d directory. This strategy was not employed on this system. Start up files for numerous well known[27] and often exploitable services could be found when examining the xinetd directory.

Table 4.4: Evidence of Unnecessary Xinetd Service Files

```
# /bin/ls /etc/xinetd.d
apass     chargen-udp  daytime      echo      finger  rexec   rsh    talk    tftp  time-udp
chargen   csdd         daytime-udp  echo-udp  ntalk   rlogin  rsync  telnet  time  wu-ftpd
#
```

While service files were found for things such as telnet and the r-commands, further examination (via the additional tests in the ckeclklist) was in order to determine their usage. If these files had to exist, the next best thing would be for them to be configured to be disabled[28]. Using the unix *grep* command we examined the disable setting in the services files established for each of these services in the xinetd.d directory.

[27] chargen, echo, rexec, rsh, telnet, and tftp all have proven in the past to be instrumental in the compromise of unix systems.
[28] The service file would contain an entry "disable = yes" signifying the service is disabled.

Table 4.5: Evidence of run disable settings in xinetd.d/

```
# /bin/grep -i "disable" /etc/xinetd.d/*
/etc/xinetd.d/apass:    disable = no
/etc/xinetd.d/chargen:  disable         = yes
/etc/xinetd.d/chargen-udp:      disable         = yes
/etc/xinetd.d/csdd:     disable = no
/etc/xinetd.d/daytime:  disable         = yes
/etc/xinetd.d/daytime-udp:      disable         = yes
/etc/xinetd.d/echo:     disable         = yes
/etc/xinetd.d/echo-udp: disable         = yes
/etc/xinetd.d/finger:   disable         = yes
/etc/xinetd.d/ntalk:    disable                 = yes
/etc/xinetd.d/rexec:    disable = no
/etc/xinetd.d/rlogin:   disable = no
/etc/xinetd.d/rsh:      disable = no
/etc/xinetd.d/rsync:    disable = yes
/etc/xinetd.d/talk:     disable                 = yes
/etc/xinetd.d/telnet:   disable = no
/etc/xinetd.d/tftp:     disable                 = yes
/etc/xinetd.d/time:     disable         = yes
/etc/xinetd.d/time-udp: disable         = yes
/etc/xinetd.d/wu-ftpd:  disable = no
#
```

While several unnecessary services (chargen, echo, daytime, finger, rsync, talk, tftp, time) were disabled in this fashion, not all of them were. This required further investigation. As per the checklist on the matter, the run level configurations were examined. Given the business goals of this system, it is important to learn what the final run state is for the system since that will determine how the output from the chkconfig would be interpreted. In this case, the server is run without a graphical user interface (GUI) and as such our focus turns to Run Level 3[29]. Using the *chkconfig* command, it is possible to determine some of the services and configuration files executed at run level 3. We found that most of the unnecessary services (xfs, netfs, portmap) were OFF during this run level.

[29] More information about the various run levels can be found in both the man pages for chkconfig and in the linux user manuals and websites.

Table 4.6: Output from chkconfig option

```
# /sbin/chkconfig --list | /bin/grep ":on"
keytable        0:off   1:on    2:on    3:on    4:on    5:on    6:off
privoxy         0:off   1:off   2:off   3:on    4:on    5:on    6:off
microcode_ctl   0:off   1:off   2:on    3:on    4:on    5:on    6:off
xfs             0:off   1:off   2:off   3:off   4:on    5:off   6:off
gpm             0:off   1:off   2:off   3:off   4:on    5:off   6:off
nscd            0:off   1:off   2:off   3:off   4:on    5:off   6:off
netfs           0:off   1:off   2:off   3:off   4:on    5:off   6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
random          0:off   1:off   2:on    3:on    4:on    5:on    6:off
rawdevices      0:off   1:off   2:off   3:on    4:on    5:on    6:off
portmap         0:off   1:off   2:off   3:off   4:on    5:off   6:off
rhnsd           0:off   1:off   2:off   3:off   4:on    5:off   6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
sendmail        0:off   1:off   2:on    3:on    4:on    5:on    6:off
anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
apmd            0:off   1:off   2:off   3:off   4:on    5:off   6:off
atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
sshd2           0:off   1:off   2:off   3:on    4:on    5:on    6:off
pcmcia          0:off   1:off   2:off   3:off   4:on    5:off   6:off
nfslock         0:off   1:off   2:off   3:off   4:on    5:off   6:off
ntpd            0:off   1:off   2:off   3:on    4:on    5:on    6:off
rc.local        0:off   1:off   2:off   3:on    4:on    5:on    6:off
rc.once         0:off   1:off   2:off   3:on    4:on    5:on    6:off
acct            0:off   1:off   2:off   3:on    4:on    5:on    6:off
squid           0:off   1:off   2:off   3:on    4:off   5:on    6:off
idsmc           0:off   1:off   2:off   3:on    4:on    5:on    6:off
#
```

At this point, it became necessary to test the behavior of the system. This was initiated with several on-system tests that verified which ports were listening and which services were running in the process table. Use of *netstat* demonstrated that only essential services were listening for connections from remote hosts. At first the sendmail listening was of concern but subsequent testing confirmed the administrator's claim that it was only listening for local connections[30]. This condition was confirmed by trying to connect to port 25 on the system and by trying to send email to a user account that existed on the system.

[30] This is established so that mail could be send off the system, but would neither receive nor relay email to the company mail servers.

Table 4.7: netstat evidence of listening services

```
# /bin/netstat -anp | grep -i "listen"
tcp        0      0 0.0.0.0:9090            0.0.0.0:*               LISTEN      10951/boa
tcp        0      0 0.0.0.0:554             0.0.0.0:*               LISTEN      723/rtspd
tcp        0      0 0.0.0.0:587             0.0.0.0:*               LISTEN      658/sendmail: accep
tcp        0      0 0.0.0.0:911             0.0.0.0:*               LISTEN      10985/(squid)
tcp        0      0 0.0.0.0:8118            0.0.0.0:*               LISTEN      10999/privoxy
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      796/sshd2
tcp        0      0 0.0.0.0:1080            0.0.0.0:*               LISTEN      2511/socks5
tcp        0      0 0.0.0.0:1081            0.0.0.0:*               LISTEN      26108/socks5-1081
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      658/sendmail: accep
tcp        0      0 0.0.0.0:6010            0.0.0.0:*               LISTEN      1194/sshd2
tcp        0      0 0.0.0.0:6011            0.0.0.0:*               LISTEN      6942/sshd2
tcp        0      0 0.0.0.0:9595            0.0.0.0:*               LISTEN      834/idsmc
tcp        0      0 0.0.0.0:6013            0.0.0.0:*               LISTEN      17380/sshd2
tcp        0      0 0.0.0.0:6014            0.0.0.0:*               LISTEN      18464/sshd2
#
```

Our final behavioral test was to use a quick port scanner called *strobe* to determine which ports were open to a remote host. Using another authorized system on the network, the squid proxy server was scanned for both tcp and udp connections. The only ports that were discovered where those assigned for services provided by the business goals of the system.

```
$ strobe ***.***.***.***
strobe 1.03 (c) 1995 Julian Assange (proff@suburbia.net).
***.***.***.***      ssh                 22/tcp # SSH Remote Login
Protocol
***.***.***.***      rtsp               554/tcp # Real Time Stream
Control Protocol
***.***.***.***      unknown            911/tcp unassigned
***.***.***.***      socks             1080/tcp # socks proxy server
$
```

Having determined which services were accessible remotely, the examination continued with verifying several other unnecessary services were not in place. By examining the process table and rpm modules installed, it was possible to determine that print services, http, and nfs were not being offered by this system.

Overall, while all the services were disabled, the presence of configuration files does raise concern over the potential for the services to be re-enabled. This could be done as part of an attack against the system and thereby allow the attacker access to services which would facilitate attacks against the system.

4.2.4.2 Network Configurations

While the administrators established specific controls on the network equipment in-line with the proxy service, a proper defense in depth strategy would not rely upon a single control to reduce risk. Given that Improper network configurations on the system can result compromise, denial of service attacks against the

system, or in the utilization of the system in a denial of service against other systems or network services, network control on the system itself are as important as those established on the equipment in the path of the server.

Our audit on the system involved 4 specific configurations we hoped to find in the /etc/sysctl.conf file. As part of the strategy to prevent syn floods >> insert definition of syn flood in footnote here << we would expect the syn cookie setting (net.ipv4.tcp_syncookies) to be set to 1. We found no such setting in the configuration file. To prevent routing table alterations, we would expect that the accept_redirects setting would be disabled. This setting was also absent. Given the recent use of fragmentation overlaps in exploitation attacks, the always_defrag option is to be set so that fragmentation is controlled. Again, the expected setting was not in place. Finally, the rather amusing setting, log_martians, which logs illegal packets such as one would find in spoof attacks, source routing attempts, and redirects, was also not established. As a result, all tests for the network checklist were failed.

Table 4.8: Network settings in /etc/sysctl.conf

```
# /bin/grep "log_martians" /etc/sysctl.conf
# /bin/grep "syncookies" /etc/sysctl.conf
# /bin/grep "accept_redirects" /etc/sysctl.conf
# /bin/grep "always_defrag" /etc/sysctl.conf
# /bin/cat /etc/sysctl.conf
# Disables packet forwarding
net.ipv4.ip_forward = 0
# Enables source route verification
net.ipv4.conf.all.rp_filter = 1
# Disables the magic-sysrq key
kernel.sysrq =1
# We need more aggressive keepalive intervals due to the DMZ firewall's
# default timeout of under 1 hour.  Set it for 30 minutes.
net.ipv4.tcp_keepalive_time = 1800
#
```

4.2.4.3 Vulnerability Assessment

Given the limited number of checklist tests we could execute against the system, the use of vulnerability assessment tools was selected in hopes that their results, if negative, could provide relevant gaps that could be addressed.

For this audit, we have selected two freeware tools to include in the checklist items, Nessus and the CIS scoring tool. For the system to pass this test, it had to be free of any medium or higher vulnerability. This rank was selected because the administrators revealed that in prior scans, low vulnerabilities were often accepted risks that were left unaddressed or mitigated through alternate controls, such as the settings on the network equipment in the path of the proxy server.

Nessus was configured to attempt all relevant plug-ins from a database current at the time of the audit.[31] We performed a full tcp/udp scan of the system as well. As revealed in the Nessus report, 2 high severity vulnerabilities were uncovered by the scan.

Table 4.9; Nessus Scan Summary

The two high issues in question were both related to out of date copies of software that have known vulnerabilities published to CERT. The ssh vulnerability that was reported turned out to be a false positive. While their software was not the current version, it was also not OpenSSH. Checking at the vendor (Fsecure) website it was confirmed that the vulnerabilities reported were all not applicable to the Fsecure ssh software. The other application reported to be out of date was the squid software. Unfortunately this was not a false positive. Referring to the squid advisory page, 3 different advisories had been released for vulnerabilities that applied to the 2.4STABLE6 version that is being used on the proxy server. >> insert discussion on the advisories and their applicability to squid server here <<

In conjunction with the nessus scan, the CIS security scoring tool was run against the system. This software required the administrator's to install the linux rpm and then to run the scan as root. While an initial score of 6.61 is not terrible, it does provide for a sufficient concern that the negatives detailed should be examined. The low score (6.6) on the CIS test indicated several issues with the current configuration of the system which were important to note. (The complete output is provided in the appendices.) Referring to our previous examination of xinetd, the CIS tool recommends that it should be configured with an "only-from" statement thereby restricting access to the services it is providing. This auditor agrees that such a configuration for certain services, such as the ssh service used to connect remotely to the machine, would be an excellent control to add to the system configurations. Several network configurations were called out (tcp_max_syn_backlog, send_redirects) that should be disabled to prevent syn floods and redirected messages. While the administrators claim such events are otherwise controlled through the network configurations, implementing these settings would not impact the services the system provides and would be a good defense in depth strategy. The scoring tool uncovered that several system accounts had no shell in /etc/passwd[32] which should instead employ a noshell[33]

[31] Relevant means that we disabled the windows family of plug-ins.
[32] No shell in /etc/passwd defaults to using /bin/sh.
[33] http://www.fish.com/titan/sr

shell. The final issue of concern was that rhost authentication was not deactivated in the /etc/pam.d/ so as to prevent users from setting up a .rhost file that would allow for them to access the system without an authentication challenge.

4.2.4.4 Logging and Log Analysis

Given the extensive throughput of a proxy server, it is important to develop automated processes which can monitor that which the fleshware cannot comprehend effectively in real time. Log analysis has the capability to close this gap in the process and as such is an important control to audit. The absence of logging enhances the ability of a threat to compromise a system and aggravates any post-incident analysis. This applies not only to threats against the system and the services it provides, but also is critical in the enforcement of corporate acceptable use policies as they pertain to downloads and web surfing in general.

We initiated our audit of logging procedures on the system by confirming that syslog, the unix logging service, was enabled on the system. This was confirmed by examining the process table via the *ps* command.

Table 4.10: Evidence of Syslog running on system

```
# ps -ef | grep syslog
root     10933     1  0 00:02 ?        00:00:02 syslogd -m 0
root      2060 32566  0 14:51 pts/3    00:00:00 grep syslog
#
```

The above syslog daemon (syslogd) confirms that the process is running with a constant interval. Given that logging is enabled, we next wished to confirm that the logs for this system were sent to a central log server (CLS). This strategy is part of a defense in depth solution that assumes if the proxy server is compromised that the logs which may contain information about the attack are safely stored on another system. By examining the syslog configuration file (syslog.conf), we confirmed that all syslog information was routed to another system via the last entry in the file which indicates:

```
# Route all to central log server
 *.*                @logserver_hostname.company.com
```

Now that it is established that the system has logging enabled and routes copies of the logs to a central log server, the retention policy must be examined to verify that the logs are kept long enough in case an event or incident is not immediately recognized as such. Company policy requires that 30 days of logs be retained on the system. By examining the log rotation configuration file (logrotate.conf), we can verify that the logs are rotate weekly, and that 4 weeks (28 days) are kept. While at the outset this may seem like a failed test, the administrators pointed out

that while only 28 days are kept on the system, the central log server retains more than 30 days online. Thus the corporate requirement is met and this auditor is convinced that 28 days on the system is sufficient to meet the controls necessary with regard to log retention; if those 2 days are significant, they can be extracted from the central log server.

The administrator then walked this auditor through the procedures for how the logs on the central log server are handled. Several times a day (6), the logs on the CLS are processes by the *logcheck*[34] program. This program is designed to highlight both known events that merit further investigation and to highlight abnormalities that are not otherwise discounted as known good traffic. This information is then routed to a distribution list of administrators. In the event that the *logcheck* script detects an event that it would qualify as a known system attack, a *procmail*[35] filter routes the message to the administrator's pager as well as their email folder.

While *logcheck* is established to look for potential security events or incidents, the administrators have added to the search terms to include terms and words that are often found on inappropriate web sites. (We will leave this list up to your imagination. Needless to say, it far surpasses George Carlin's 7 dirty words). This information is gathered so that it can be considered for inclusion in a blacklist that is routinely updated.

4.2.5 Application (Squid) Settings

The system failed to achieve a passing result with regard to checklist item Squid-02. While the proxy server was properly configured to control which ports were available to users, the server lacked any configuration of the IP address ACL's to reduce access to the system only to those authorized. Examining the squid configuration file (/etc/squid/squid.conf) the server was missing any definition of acceptable subnets and the default deny to all others was omitted. All ports present in the squid.conf can be accounted for by the business goals of this system. There is no control in place to limit who could use this system. This runs contrary to the business goal that this machine was established for a particular internal group to use. As such, anyone with a system on the internal network could access the services this machine offers. This universal access across a company of this size, combined with the use of dhcp[36], detrimentally impacts the ability to investigation misuse or abuse of the system.

[34] Logcheck can be found at: http://freshmeat.net/projects/logcheck/
[35] Procmail can be found at: http://www.procmail.org/
[36] While dhcp greatly alleviates the management of ip addresses for a large company, it also provides a poor track to systems on the network and can, depending on the degree to which dhcp is configure to log connections, hinder forensic examinations.

Table 4.11: Suid ACL settings

```
#Recommended minimum configuration:
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl SSL_ports port 443 563
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443 563     # https, snews
# acl Safe_ports port 70                # gopher
# acl Safe_ports port 210               # wais
acl Safe_ports port 1025-65535  # unregistered ports
# acl Safe_ports port 280               # http-mgmt
# acl Safe_ports port 488               # gss-http
# acl Safe_ports port 591               # filemaker
# acl Safe_ports port 777               # multiling http
acl CONNECT method CONNECT

#  TAG: http_access
```

Given the ability an attacker would have to overload the squid service, several configurations were validated which could together help reduce the risk from such attacks.

Several configurations in squid allow for protection against over-allocation of resources which could starve legitimate use. These settings were captured in the Squid-04 checklist. The first several audit items involve examining the size of web traffic being passed through the squid service. These configurations are established to prevent an attacker from requesting exceptionally large content or numerous queries summing up to sufficient content that it would choke the resources of the proxy server. In addition, the setting for the request_header_max_size was also verified to be such that it would prevent an attacker from overloading the squid services with extremely large html headers. An examination of the squid.conf file revealed that these setting were established according to recommended values that would protect against denial of service strategies that would rely on overloading the system.

Table 12: Squid.conf settings for traffice size control

```
# grep maximum_object_size squid.conf
#  TAG: maximum_object_size     (bytes)
maximum_object_size 4096 KB
#  TAG: maximum_object_size_in_memory   (bytes)
maximum_object_size_in_memory 8 KB
#       the value of maximum_object_size above its default of 4096 KB to
# grep quick_abort squid.conf
#  TAG: quick_abort_min (KB)
#  TAG: quick_abort_max (KB)
#  TAG: quick_abort_pct (percent)
#       quick_abort values to the amount of data transfered until
#       If the transfer has less than 'quick_abort_min' KB remaining,
#       it will finish the retrieval.  Setting 'quick_abort_min' to -1
#       will disable the quick_abort feature.
#       If the transfer has more than 'quick_abort_max' KB remaining,
#       If more than 'quick_abort_pct' of the transfer has completed,
quick_abort_min 16 KB
quick_abort_max 16 KB
quick_abort_pct 95
#       request_timeout, pconn_timeout and quick_abort values.
# grep dns_nameservers squid.conf
#  TAG: dns_nameservers
#       Example: dns_nameservers 10.0.0.1 192.172.0.4
#
```

The next configuration options which were of concern were those dealing with Domain Name Services (DNS)[37]. By examining the settings for dns_nameservers and ignore_unknown_nameservers, it was possible to conclude that the squid service is configured to only access resolutions from a trusted DNS servers found in the /etc/resolv.conf.

The final settings which this checklist is concerned with pertain to the user's connection to the squid service. The client_lifetime sets the maximum amount of time a client is allowed to connect to a squid process. This setting will prevent a an attacker from initiating and hold enough sessions to starve other users from access to the squid service. This risk is sufficiently mitigated by the allowing clients a 1 day lifetime. The administrator's explained that some processes run through the server could take most of a business day and as such, this lifetime was an agreed upon since it was the default. The default pconn_timeout was retained and verified in the squid.conf as 120 seconds before an idle persistent connection is dropped.

[37] Those unfamiliar with DNS are advised to look on the web for further information. One good site on this matter is: http://computer.howstuffworks.com/dns1.htm

4.2.6 Summary of Findings

Checklist Item	Objective	Pass or Fail
Administrative-03	Change management documentation must exist for all critical infrastructure systems	Fail
Hardening-03	Disable un-necessary services	Pass
Hardening-06	An Intrustion Detection System must be installed which routinely (daily) verifies the integrity of binaries on the system.	Fail
Hardening-08	System and application logs are established, reviewed (daily), and rotated with 30 days left on system	Pass
Hardening-09	The system will report no High or Medium vulnerabilities when subjected to vulnerability assessment tools.	Fail
Network-01	Enable network-related security settings on the system to minimize risks associated with attacks that rely upon packet manipulation.	Fail
Physical-01	The proxy server will reside in a location designed to reduce risks associated with physical access to the system.	Pass
Physical-04	The operating system will be configured so as to reduce risks associated with physical access to the system.	Fail
Squid-02	Implement access control lists (ACL) to restrict access to server by trusted network ranges only.	Fail
Squid-04	Implement squid configurations (squid.conf) to reduce risk of denial of service attacks against the service.	Pass

4.3 Audit Recommendations

4.3.1 Rebuild/Upgrade squid proxy server

4.3.1.1 Recommendation

While this may seem like an extreme recommendation to begin with, there are numerous controls which were not in place that call into question the integrity of the system. Given the significant configuration changes, software upgrades, and behavioral changes that this audit recommends, it appears it would be easier to

build a replacement proxy server with all the proper control than to try to implement them on the production system.

4.3.1.2 Costs

The costs involved in this matter would include the hardware and engineering resources. However, given the current HW is at least 2 years old, the upgrade is already overdue.

4.3.1.3 Compensating controls

If this cost is insurmountable at this time, the administrators need to at least correct all the current outstanding security gaps presented in this document, those disclosed by the vulnerability assessment tools, and adopt a program to schedule periodic upgrades to the system at least every 6 months.

4.3.2 Implement Change Control Procedures

4.3.2.1 Recommendation

Anyone who can affect an authorized change to the system must be taught to incorporate those changes into a change control system. From creation to current state, every setting should be accounted for, either as out of the box or explicitly chosen. The use of RCS by the admins is commendable and an excellent idea, but it must be accompanied by an off-system change control that can be used to chart the evolution of the system.

4.3.2.2 Cost

The cost for the adoption of this process is time. The degree of discipline required to implement this correctly is not that extensive, but its adoption will require the administrators to spend more time in the non-technical aspects of their jobs.

4.3.2.3 Compensating Controls

If a solid change control process cannot be adopted, then a mitigating control would be to place greater focus upon the use of the tripwire database as a method of change control. Each time an authorized change is made the updated tripwire database can serve as a last known good image for the system. Saved previous databases could reveal the iterations and evolution of the system, to some degree.

4.3.3 Update Defense in Depth Paradigm

4.3.3.1 Recommendation

There needs to be a change in philosophy for the administrators of the system. Their otherwise sound defense in depth strategy needs to incorporate the idea of

removing services and rpm modules, not just disabling them. These extra steps will help raise the difficulty for someone trying to compromise the system. Resource further efforts toward using the current vulnerability assessment tools to critically evaluate the replacement server. The CIS tool and the Nessus scan both drew attention to system configurations which could be modified to reduce risk. More of these changes need to be validated and implemented.

4.3.3.2 Cost

The adoption of a revitalized defense in depth strategy will require engineering resources to revisit and relearn this strategy again. It will take time away from other projects and responsibilities as greater testing will be required to land new systems with minimal risk. This is a learning process and there is a risk involved that something essential will be shut off and a service will stop running. Such risks should be controlled by performing thorough testing before applying new controls to the production environment.

4.3.3.3 Compensating Controls

If the administrators cannot be retrained in this material, then they should then select well known industry standards to implement. The use of industry BKM's to reduce risk will help mitigate the growing gaps that arise from degraded security postures which result from neglected responsibilities.

4.3.4 Improve intrusion detection/prevention capabilities

4.3.4.1 Recommendation

Greater detection capabilities need to be enabled on the system. The disabled IDS is a demonstration of a good idea with less than effective implementation. IDS and log analysis need to be revisited and implemented in a manner that results in greater awareness of change by the administrators. This detection capability would be dramatically improved by the introduction of a Network Intrusion Detection System in the path of the server for additional coverage of the network traffic passing through the proxy server.

4.3.4.2 Cost

Depending on if freeware is used or not, the cost could include software purchases and/or engineering resources to install and to react to, the detection/prevention technologies. In the NIDS space, there could be additional costs for SW and HW depending on if a vendor product, a vendor appliance, or an open source solution is selected.

4.3.4.3 Compensating Controls

Without comprehensive intrusion detection/prevention, the administrators should refine their ability to react to an incident. Given the 3 aspects of security, prevention – detection – response, if the company selects not to endorse prevention and detection, then they should suitably prepare to response. This

would include a formal business continuity procedure that would provide sufficient guidance and information so that if/when the system is compromised; they can restore a (hopefully patched and improved) replacement.

4.3.5 Increased Security Awareness

4.3.5.1 Recommendation

There needs to be a shift in the mindset of the administrators to demonstrate greater awareness of potential vulnerabilities and the proactive remediation of those risks through the adoption of a diligent patching procedure. To facilitate this paradigm shift, management needs to recognize the importance of this diligence and endorse it regardless of the impact it may have on the other responsibilities the administrators are assigned, unless those responsibilities are assigned a greater importance in the company's business goals and strategies.[38]

4.3.5.2 Cost

The greatest cost for implementing this recommendation is the reallocation of time spent by the administrators on being aware of alerts and disclosed vulnerabilities that pertain to the software and operating system of the proxy server. The time to monitor either key security discussion lists or parse through emails from alert lists will still detract from the time the administrators can spend on other responsibilities.

4.3.5.3 Compensating Controls

If the mindset of the administrators cannot be adjusted to proactively seek out this information, then at the very least they should be subscribed to the key advisories email lists for the operating system and software so that they receive timely notifications about known, verified, published vulnerabilities.

[38] While we all would like to hope that security is the top priority, clearly business goals must be weighed in the equation.

5 For the Auditor: Post Mortem Thoughts on Audit

In the aftermath of the audit, there are certain steps which were not taken that upon reflection, would have potentially provided further validation of the findings. As such, I have elected to capture these as a post-mortem of sorts to the audit process. These findings were not submitted to the audit target and are included as part of this paper in affirmation that there is always an opportunity to refine our processes.

- *Automate Configuration Examinations where possible*. As I worked through the various aspects of examining the system and application configurations, it would have been easier to generate a shell script to be executed to gather all the relevant information for me rather than executing each validation individually.
- *Syslog Program*. It would be of benefit to write an open ended syslog program that would allow for a free-style message to be sent to any daemon and facility. This way a specifically crafted message could be sent, and tracked through their system. Ex. "When you get this please let the auditor know."
- *External Web Site for Testing*. It would be extremely helpful to have an external website prepared in advance to test certain squid conditions, such as the maximum header size, under controlled conditions.
- *Software to Stress Test DOS Resistance*. Several settings within squid are designed to reduce the ability of Denial of Service attacks. Software which could emulate 1000's of connections, connection idle times, etc would be of tremendous help in verifying the behavior of squid. One such tool could be the apache benchmark software. It may also be possible to design this in perl given the right web-interface modules.
- *Packet Crafting Software*. This software would be extremely valuable in verifying network ACL's which are in place on either side of the proxy server. While not among the 10 selected checklists, this could be an emerging concern down the line and is worthy of inclusion in a complete audit of such systems.
- *Checklist modification*. It would be worthwhile to add content to the checklists to itemize which tests in the checklist need to be run as root versus a standard user, which are local to the system, which are remote, and which could require the system to be rebooted. While these factors should have no bearing the checklists selected, including them could assist when covering the content in the entrance meeting so that the administrators of the auditable items are better informed with regard to the audit procedures.

References

Beauman, Sean “Auditing a Linux FTP and DNS Server: And Administrators Perspective”. Sept 20 2003

Common Vulnerabilities and Exposures. “CAN-2002-1357 - 1360”, http://cve.mitre.org/

Red Hat, Inc. “Red Hat Linux 9, Red Hat Linux Security Guide”, June 12, 2003. http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/security-guide/

The SANS Institute. Securing Linux Step-by-Step, Version 1.0. The SANS Institute, 2000.

The SANS Institute. Auditing Networks, Perimeters, And Systems. The SANS Institute, 2003.

The SANS Institute. Incident Handling and Hacker Techniques. The SANS Institute, 2003.

The SANS Institute. Security Essentials. The SANS Institute, 2003.
Laude, Mary. “Auditing Red Hat Linux 7.0” 23 July 2001. URL: http://www.giac.org/practical/Mary_Laude_GSNA.zip

Spitzner, Lance. “Armoring Linux”. 19 September 2000. URL: http://www.enteract.com/~lspitz/linux.html

Bayne, James, “An Overview of Threat and Risk Assessment“. Jan 22, 2002.

Galarneau, Eric. “Security considerations with Squid Proxy Server”. April 2, 2003

Cole, Eric, Jason Fossen, Stephen Northcutt, Hal Pomeranz. SANS Security Essentials with CISSP CBK Version 2.1, USA: The SANS Institute, 2003

Weber, Don C. “Sourcefire Intrusion Detection System Deployment
An Auditor’s Perspective”. September 24, 2003

Aubry, Carmen. “Auditing a print and scan server protected by the VisNetic for Workstation firewall” (12 Feb 2004)

Skoudis, Ed. Counter Hack. Upper Saddle River, NJ: Prentice Hall, 2002

Mourani, Gerhard. “Securing and Optimizing Linux: RedHat Edition.” OpenDocs, LLC. 2000. September 19, 2003
URL: http://www.tldp.org/LDP/solrhe/Securing-Optimizing-Linux-RH-Edition-v1.3

6 Appendice

6.1 Output from CIS security tool

*** CIS Ruler Run ***
Starting at time 20040823-11:19:47

Negative: 1.1 System appears not to have been patched within the last month.
Negative: 2.2 No Authorized Only banner for telnet in file /etc/xinetd.d/telnet.
Negative: 2.2 No Authorized Only banner for ftp in file /etc/xinetd.d/wu-ftpd.
Negative: 2.2 No Authorized Only banner for login in file /etc/xinetd.d/rlogin.
Positive: 2.3 telnet is deactivated.
Positive: 2.4 ftp is deactivated.
Positive: 2.5 rsh, rcp and rlogin are deactivated.
Positive: 2.6 tftp is deactivated.
Negative: 2.7 xinetd either requires global 'only-from' statement or one for each service.
Positive: 3.1 Miscellaneous scripts are all turned off.
Positive: 3.2 NFS Server script nfs is deactivated.
Positive: 3.3 This machine isn't being used as an NFS client.
Positive: 3.4 NIS Client processes are deactivated.
Positive: 3.5 NIS Server processes are deactivated.
Positive: 3.6 portmapper has been deactivated.
Positive: 3.7 samba windows filesharing daemons are deactivated.
Positive: 3.8 netfs rc script is deactivated.
Positive: 3.9 printing daemon is deactivated.
Positive: 3.10 Graphical login is deactivated.
Negative: 3.11 Mail daemon is on and collecting mail from the network.
Positive: 3.12 Web server is deactivated.
Positive: 3.13 snmp daemon is deactivated.
Positive: 3.14 DNS server is deactivated.
Positive: 3.15 postgresql (SQL) database server is deactivated.
Positive: 3.16 routing daemons are deactivated.
Positive: 3.17 Webmin GUI-based system administration daemon deactivated.
Negative: 3.18 Squid web cache daemon not deactivated.
Positive: 3.19 inetd/xinetd not activated.
Positive: 3.20 Found a good daemon umask.
Negative: 4.1 Coredumps aren't deactivated.
Positive: 4.2 /etc/exports is empty or doesn't exist, so it doesn't need to be tuned for privports.
Negative: 4.3 /proc/sys/net/ipv4/tcp_max_syn_backlog should be at least 4096 to handle SYN floods.
Negative: 4.4 /proc/sys/net/ipv4/conf/eth0/send_redirects should be 0 to disable outgoing redirect messages.

Negative: 4.4 /proc/sys/net/ipv4/conf/lo/send_redirects should be 0 to disable outgoing redirect messages.
Negative: 4.4 /proc/sys/net/ipv4/conf/default/send_redirects should be 0 to disable outgoing redirect messages.
Positive: 5.1 syslog captures auth and authpriv messages.
Negative: 6.1 Removable filesystem /mnt/cdrom is not mounted nosuid.
Negative: 6.2 PAM allows users to mount CD-ROMS. (/etc/security/console.perms)
Negative: 6.2 PAM allows users to mount floppies. (/etc/security/console.perms)
Negative: 6.3 /etc/shadow has wrong permissions.
Positive: 6.4 all temporary directories have sticky bits set.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rexec.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rlogin.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rsh.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rexec.dist.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rlogin.dist.
Negative: 7.1 rhosts authentication not deactivated in /etc/pam.d/rsh.dist.
Positive: 7.2 /etc/hosts.equiv file not present or has size zero.
Negative: 7.3 User ident is not present in /etc/ftpusers
Negative: 7.3 User gdm is not present in /etc/ftpusers
Negative: 7.3 User squid is not present in /etc/ftpusers
Negative: 7.3 User rpcuser is not present in /etc/ftpusers
Negative: 7.3 User root is not present in /etc/ftpusers
Negative: 7.3 User mailnull is not present in /etc/ftpusers
Negative: 7.4 Couldn't open cron.allow
Negative: 7.4 Couldn't open at.allow
Negative: 7.5 The permissions on /etc/crontab are not sufficiently restrictive.
Negative: 7.6 No Authorized Only message in /etc/motd.
Positive: 7.6 All authorized-use-only warning banners are in place.
Negative: 7.7 /etc/securetty has a non tty1-12 line: tty10.
Positive: 7.8 lilo is password-protected.
Negative: 8.1 uucp has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 operator has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 adm has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 bin has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 daemon has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 ftp has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 gdm has a valid shell of /bin/bash.
Negative: 8.1 ident has a valid shell of /bin/false.

Negative: 8.1 lp has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 mail has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 news has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 nobody has a valid shell of /bin/sh. Remember, an empty shell field in /etc/passwd signifies /bin/sh.
Negative: 8.1 rpcuser has a valid shell of /bin/false.
Positive: 8.2 There were no +: entries in passwd, shadow or group maps.
Positive: 8.4 Only one UID 0 account AND it is named root.
Positive: 8.5 root's PATH is clean of group/world writable directories or the current-directory link.
Positive: 8.6 root account has no dangerous rhosts, shosts, or netrc files.
Positive: 8.7 No user's home directory is world or group writable.
Positive: 8.8 No group or world-writable dotfiles!
Positive: 8.9 No user has a .netrc or .rhosts file.
Negative: 8.10 Default umask may not block group-writable. Check /etc/csh.login.
Negative: 8.10 Default umask may not block world-writable. Check /etc/bashrc.
Negative: 8.10 Default umask may not block group-writable. Check /etc/bashrc.
Negative: 8.10 Default umask may not block world-writable. Check /etc/csh.cshrc.
Negative: 8.10 Default umask may not block group-writable. Check /etc/csh.cshrc.
Positive: 9.1 System is running sshd.
Positive: 9.2 This machine is synced with ntp.
Preliminary rating given at time: Mon Aug 23 11:19:52 2004

Preliminary rating = 6.61 / 10.00

Negative: 6.5 Non-standard SUID program /etc/X11/Xconf/ModXF86Helper
Negative: 6.5 Non-standard SUID program /usr/local/bin/sudo
Negative: 6.5 Non-standard SUID program /usr/sbin/sendmail.dist
Negative: 6.5 Non-standard SGID program /usr/sbin/sendmail
Ending run at time: Mon Aug 23 11:19:55 2004

Final rating = 6.61 / 10.00

6.2 Tripwire Output

```
###############################################################################
######

@@section GLOBAL
TWROOT="/usr/sbin";
TWBIN="/usr/sbin";
TWPOL="/etc/tripwire";
TWDB="/var/lib/tripwire";
TWSKEY="/etc/tripwire";
TWLKEY="/etc/tripwire";
TWREPORT="/var/lib/tripwire/report";
HOSTNAME=HOSTNAME;

@@section FS
SEC_CRIT      = $(IgnoreNone)-SHa ;  # Critical files that cannot change
SEC_SUID      = $(IgnoreNone)-SHa ;  # Binaries with the SUID or SGID flags set
SEC_BIN       = $(ReadOnly) ;        # Binaries that should not change
SEC_CONFIG    = $(Dynamic) ;         # Config files that are changed infrequently but
accessed often
SEC_LOG       = $(Growing) ;         # Files that grow, but that should never change
ownership
SEC_INVARIANT = +tpug ;              # Directories that should never change permission
or ownership
SIG_LOW       = 33 ;                 # Non-critical files that are of minimal security impact
SIG_MED       = 66 ;                 # Non-critical files that are of significant security impact
SIG_HI        = 100 ;                # Critical files that are significant points of vulnerability


# Tripwire Binaries
(
  rulename = "Tripwire Binaries",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  $(TWBIN)/siggen                 -> $(SEC_BIN) ;
  $(TWBIN)/tripwire               -> $(SEC_BIN) ;
  $(TWBIN)/twadmin                  -> $(SEC_BIN) ;
  $(TWBIN)/twprint                -> $(SEC_BIN) ;
}

# Tripwire Data Files - Configuration Files, Policy Files, Keys, Reports, Databases
(
```

```
  rulename = "Tripwire Data Files",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  # NOTE: We remove the inode attribute because when Tripwire creates a backup,
  # it does so by renaming the old file and creating a new one (which will
  # have a new inode number).  Inode is left turned on for keys, which shouldn't
  # ever change.

  # NOTE: The first integrity check triggers this rule and each integrity check
  # afterward triggers this rule until a database update is run, since the
  # database file does not exist before that point.

  $(TWDB)                       -> $(SEC_CONFIG) -i ;
  $(TWPOL)/tw.pol                  -> $(SEC_BIN) -i ;
  $(TWPOL)/tw.cfg                  -> $(SEC_BIN) -i ;
  $(TWLKEY)/$(HOSTNAME)-local.key      -> $(SEC_BIN) ;
  $(TWSKEY)/site.key                -> $(SEC_BIN) ;

  #don't scan the individual reports
  $(TWREPORT)                   -> $(SEC_CONFIG) (recurse=0) ;
}


# Tripwire HQ Connector Binaries
#(
#  rulename = "Tripwire HQ Connector Binaries",
#  severity = $(SIG_HI)
#)
#{
#  $(TWBIN)/hqagent                -> $(SEC_BIN) ;
#}
#
# Tripwire HQ Connector - Configuration Files, Keys, and Logs

##########################################################################
###### #
#                                                       # #
# Note: File locations here are different than in a stock HQ Connector      # #
# installation.  This is because Tripwire 2.3 uses a different path         # #
# structure than Tripwire 2.2.1.                                  # #
#                                                       # #
# You may need to update your HQ Agent configuation file (or this policy     # #
# file) to correct the paths.  We have attempted to support the FHS standard # #
# here by placing the HQ Agent files similarly to the way Tripwire 2.3      # #
```

```
# places them.                                          # #
#                                                    ##
#########################################################################
######

#(
#  rulename = "Tripwire HQ Connector Data Files",
#  severity = $(SIG_HI)
#)
#{
#
#########################################################################
#####
#
#########################################################################
######
#  # NOTE: Removing the inode attribute because when Tripwire creates a backup ##
#  # it does so by renaming the old file and creating a new one (which will    ##
#  # have a new inode number).  Leaving inode turned on for keys, which        ##
#  # shouldn't ever change.                                                    ##
#
#########################################################################
#####
#
#  $(TWBIN)/agent.cfg                -> $(SEC_BIN) -i ;
#  $(TWLKEY)/authentication.key      -> $(SEC_BIN) ;
#  $(TWDB)/tasks.dat                 -> $(SEC_CONFIG) ;
#  $(TWDB)/schedule.dat              -> $(SEC_CONFIG) ;
#
#  # Uncomment if you have agent logging enabled.
#  #/var/log/tripwire/agent.log      -> $(SEC_LOG) ;
#}



# Commonly accessed directories that should remain static with regards to owner and
group
(
  rulename = "Invariant Directories",
  severity = $(SIG_MED),
  emailto  = squid.admin@company.com
)
{
  /                         -> $(SEC_INVARIANT) (recurse = 0) ;
  /home                        -> $(SEC_INVARIANT) (recurse = 0) ;
  /etc                       -> $(SEC_INVARIANT) (recurse = 0) ;
```

```
 /bin                        -> $(SEC_INVARIANT) (recurse = 0) ;
}
 ################################################
 #                              ##
################################################ #
#                              # #
# File System and Disk Administration Programs # #
#                              ##
################################################

(
 rulename = "File System and Disk Administraton Programs",
 severity = $(SIG_HI),
 emailto  = squid.admin@company.com
)
{
 /sbin/accton                -> $(SEC_CRIT) ;
 /sbin/badblocks              -> $(SEC_CRIT) ;
 /sbin/dosfsck               -> $(SEC_CRIT) ;
 /sbin/e2fsck                -> $(SEC_CRIT) ;
 /sbin/debugfs               -> $(SEC_CRIT) ;
 /sbin/dumpe2fs               -> $(SEC_CRIT) ;
 /sbin/dump                  -> $(SEC_CRIT) ;
 /sbin/dump.static            -> $(SEC_CRIT) ;
 /sbin/e2label               -> $(SEC_CRIT) ;
 /sbin/fdisk                 -> $(SEC_CRIT) ;
 /sbin/fsck                  -> $(SEC_CRIT) ;
 /sbin/fsck.ext2              -> $(SEC_CRIT) ;
 /sbin/fsck.minix             -> $(SEC_CRIT) ;
 /sbin/fsck.msdos            -> $(SEC_CRIT) ;
 /sbin/ftl_check             -> $(SEC_CRIT) ;
 /sbin/ftl_format            -> $(SEC_CRIT) ;
 /sbin/hdparm                -> $(SEC_CRIT) ;
 /sbin/mkbootdisk             -> $(SEC_CRIT) ;
 /sbin/mkdosfs               -> $(SEC_CRIT) ;
 /sbin/mke2fs                -> $(SEC_CRIT) ;
 /sbin/mkfs                  -> $(SEC_CRIT) ;
 /sbin/mkfs.ext2              -> $(SEC_CRIT) ;
 /sbin/mkfs.minix             -> $(SEC_CRIT) ;
 /sbin/mkfs.msdos             -> $(SEC_CRIT) ;
 /sbin/mkinitrd              -> $(SEC_CRIT) ;
 /sbin/mkpv                  -> $(SEC_CRIT) ;
 /sbin/mkraid                -> $(SEC_CRIT) ;
 /sbin/mkswap                 -> $(SEC_CRIT) ;
 /sbin/pcinitrd              -> $(SEC_CRIT) ;
 /sbin/quotacheck             -> $(SEC_CRIT) ;
```

```
  /sbin/quotaon                  -> $(SEC_CRIT) ;
  /sbin/raidstart                -> $(SEC_CRIT) ;
  /sbin/resize2fs                -> $(SEC_CRIT) ;
  /sbin/restore                  -> $(SEC_CRIT) ;
  /sbin/restore.static           -> $(SEC_CRIT) ;
  /sbin/scsi_info                -> $(SEC_CRIT) ;
  /sbin/sfdisk                   -> $(SEC_CRIT) ;
  /sbin/tune2fs                  -> $(SEC_CRIT) ;
  /sbin/update                   -> $(SEC_CRIT) ;
  /bin/mount                     -> $(SEC_CRIT) ;
  /bin/umount                    -> $(SEC_CRIT) ;
  /bin/touch                     -> $(SEC_CRIT) ;
  /bin/mkdir                     -> $(SEC_CRIT) ;
  /bin/mknod                     -> $(SEC_CRIT) ;
  /bin/mktemp                    -> $(SEC_CRIT) ;
  /bin/rm                        -> $(SEC_CRIT) ;
  /bin/rmdir                     -> $(SEC_CRIT) ;
  /bin/chgrp                     -> $(SEC_CRIT) ;
  /bin/chmod                     -> $(SEC_CRIT) ;
  /bin/chown                     -> $(SEC_CRIT) ;
  /bin/cp                        -> $(SEC_CRIT) ;
  /bin/cpio                      -> $(SEC_CRIT) ;
}

################################### #
#                       # #
# Kernel Administration Programs # #
#                       ##
###################################

(
  rulename = "Kernel Administration Programs",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/depmod                   -> $(SEC_CRIT) ;
  /sbin/adjtimex                 -> $(SEC_CRIT) ;
  /sbin/ctrlaltdel               -> $(SEC_CRIT) ;
  /sbin/insmod                   -> $(SEC_CRIT) ;
  /sbin/insmod.static            -> $(SEC_CRIT) ;
  /sbin/insmod_ksymoops_clean    -> $(SEC_CRIT) ;
  /sbin/klogd                    -> $(SEC_CRIT) ;
  /sbin/ldconfig                 -> $(SEC_CRIT) ;
  /sbin/minilogd                 -> $(SEC_CRIT) ;
  /sbin/modinfo                  -> $(SEC_CRIT) ;
```

```
  /sbin/sysctl                    -> $(SEC_CRIT) ;
}

######################## #
#                 # #
# Networking Programs # #
#                 ##
########################

(
  rulename = "Networking Programs",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/arp                      -> $(SEC_CRIT) ;
  /sbin/dhcpcd                     -> $(SEC_CRIT) ;
  /sbin/ifcfg                    -> $(SEC_CRIT) ;
  /sbin/ifconfig                   -> $(SEC_CRIT) ;
  /sbin/ifdown                     -> $(SEC_CRIT) ;
  /sbin/ifenslave                  -> $(SEC_CRIT) ;
  /sbin/ifport                   -> $(SEC_CRIT) ;
  /sbin/ifup                     -> $(SEC_CRIT) ;
  /sbin/ifuser                   -> $(SEC_CRIT) ;
  /sbin/ip                      -> $(SEC_CRIT) ;
  /sbin/ipchains                   -> $(SEC_CRIT) ;
  /sbin/ipchains-restore             -> $(SEC_CRIT) ;
  /sbin/ipchains-save              -> $(SEC_CRIT) ;
  /sbin/ipfwadm                    -> $(SEC_CRIT) ;
  /sbin/ipmaddr                  -> $(SEC_CRIT) ;
  /sbin/iptables                 -> $(SEC_CRIT) ;
  /sbin/iptunnel                 -> $(SEC_CRIT) ;
  /sbin/iwconfig                   -> $(SEC_CRIT) ;
  /sbin/iwpriv                   -> $(SEC_CRIT) ;
  /sbin/iwspy                    -> $(SEC_CRIT) ;
  /sbin/netreport                  -> $(SEC_CRIT) ;
  /sbin/plipconfig                   -> $(SEC_CRIT) ;
  /sbin/portmap                    -> $(SEC_CRIT) ;
  /sbin/ppp-watch                    -> $(SEC_CRIT) ;
  /sbin/route                    -> $(SEC_CRIT) ;
  /sbin/slattach                 -> $(SEC_CRIT) ;
  /sbin/ypbind                     -> $(SEC_CRIT) ;
  /bin/ping                      -> $(SEC_CRIT) ;
}

################################### #
```

```
#                       # #
# System Administration Programs # #
#                       ##
####################################

(
 rulename = "System Administration Programs",
 severity = $(SIG_HI),
 emailto  = squid.admin@company.com
)
{
 /sbin/chkconfig              -> $(SEC_CRIT) ;
 /sbin/fuser                  -> $(SEC_CRIT) ;
 /sbin/halt                   -> $(SEC_CRIT) ;
 /sbin/init                   -> $(SEC_CRIT) ;
 /sbin/initlog                -> $(SEC_CRIT) ;
 /sbin/killall5               -> $(SEC_CRIT) ;
 /sbin/pwdb_chkpwd            -> $(SEC_CRIT) ;
 /sbin/rescuept               -> $(SEC_CRIT) ;
 /sbin/rmt                    -> $(SEC_CRIT) ;
 /sbin/rpc.lockd              -> $(SEC_CRIT) ;
 /sbin/rpc.statd              -> $(SEC_CRIT) ;
 /sbin/rpcdebug               -> $(SEC_CRIT) ;
 /sbin/service                -> $(SEC_CRIT) ;
 /sbin/setsysfont             -> $(SEC_CRIT) ;
 /sbin/shutdown               -> $(SEC_CRIT) ;
 /sbin/sulogin                -> $(SEC_CRIT) ;
 /sbin/swapon                 -> $(SEC_CRIT) ;
 /sbin/syslogd                -> $(SEC_CRIT) ;
 /sbin/unix_chkpwd            -> $(SEC_CRIT) ;
 /bin/pwd                     -> $(SEC_CRIT) ;
 /bin/uname                   -> $(SEC_CRIT) ;
}

######################################## #
#                       # #
# Hardware and Device Control Programs # #
#                       ##
########################################
(
 rulename = "Hardware and Device Control Programs",
 severity = $(SIG_HI),
 emailto  = squid.admin@company.com
)
{
 /sbin/cardctl                -> $(SEC_CRIT) ;
```

```
  /sbin/cardmgr                  -> $(SEC_CRIT) ;
  /sbin/hwclock                  -> $(SEC_CRIT) ;
  /sbin/isapnp                   -> $(SEC_CRIT) ;
  /sbin/kbdrate                  -> $(SEC_CRIT) ;
  /sbin/losetup                  -> $(SEC_CRIT) ;
  /sbin/lspci                    -> $(SEC_CRIT) ;
  /sbin/pnpdump                  -> $(SEC_CRIT) ;
  /sbin/probe                    -> $(SEC_CRIT) ;
  /sbin/pump                     -> $(SEC_CRIT) ;
  /sbin/setpci                   -> $(SEC_CRIT) ;
  /sbin/shapecfg                 -> $(SEC_CRIT) ;
}

################################# #
#                       # #
# System Information Programs # #
#                       ##
#################################
(
  rulename = "System Information Programs",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/consoletype              -> $(SEC_CRIT) ;
  /sbin/kernelversion            -> $(SEC_CRIT) ;
  /sbin/runlevel                 -> $(SEC_CRIT) ;
}

#################################### #
#                         # #
# Application Information Programs # #
#                         ##
####################################

(
  rulename = "Application Information Programs",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/genksyms                 -> $(SEC_CRIT) ;
  /sbin/rtmon                    -> $(SEC_CRIT) ;
  /sbin/sln                      -> $(SEC_CRIT) ;
}
```

```
########################## #
#                  # #
# Shell Related Programs # #
#                  ##
##########################
(
  rulename = "Shell Releated Programs",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/getkey                    -> $(SEC_CRIT) ;
}


################ #
#           # #
# OS Utilities # #
#           ##
################
(
  rulename = "Operating System Utilities",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /bin/cat                      -> $(SEC_CRIT) ;
  /bin/date                      -> $(SEC_CRIT) ;
  /bin/dd                       -> $(SEC_CRIT) ;
  /bin/df                      -> $(SEC_CRIT) ;
  /bin/echo                      -> $(SEC_CRIT) ;
  /bin/egrep                     -> $(SEC_CRIT) ;
  /bin/false                    -> $(SEC_CRIT) ;
  /bin/fgrep                     -> $(SEC_CRIT) ;
  /bin/gawk                       -> $(SEC_CRIT) ;
  /bin/grep                      -> $(SEC_CRIT) ;
  /bin/true                     -> $(SEC_CRIT) ;
  /bin/arch                      -> $(SEC_CRIT) ;
  /bin/ash                      -> $(SEC_CRIT) ;
  /bin/ash.static                 -> $(SEC_CRIT) ;
  /bin/aumix-minimal                -> $(SEC_CRIT) ;
  /bin/basename                   -> $(SEC_CRIT) ;
  /bin/consolechars                -> $(SEC_CRIT) ;
  /bin/dmesg                     -> $(SEC_CRIT) ;
  /bin/doexec                     -> $(SEC_CRIT) ;
  /bin/ed                       -> $(SEC_CRIT) ;
```

```
  /bin/gunzip                   -> $(SEC_CRIT) ;
  /bin/gzip                    -> $(SEC_CRIT) ;
  /bin/hostname                  -> $(SEC_CRIT) ;
  /bin/igawk                    -> $(SEC_CRIT) ;
  /bin/ipcalc                   -> $(SEC_CRIT) ;
  /bin/kill                    -> $(SEC_CRIT) ;
  /bin/ln                      -> $(SEC_CRIT) ;
  /bin/loadkeys                  -> $(SEC_CRIT) ;
  /bin/login                    -> $(SEC_CRIT) ;
  /bin/ls                      -> $(SEC_CRIT) ;
  /bin/mail                     -> $(SEC_CRIT) ;
  /bin/more                     -> $(SEC_CRIT) ;
  /bin/mt                      -> $(SEC_CRIT) ;
  /bin/mv                       -> $(SEC_CRIT) ;
  /bin/netstat                  -> $(SEC_CRIT) ;
  /bin/nice                     -> $(SEC_CRIT) ;
  /bin/ps                      -> $(SEC_CRIT) ;
  /bin/rpm                      -> $(SEC_CRIT) ;
  /bin/sed                     -> $(SEC_CRIT) ;
  /bin/setserial                -> $(SEC_CRIT) ;
  /bin/sfxload                  -> $(SEC_CRIT) ;
  /bin/sleep                    -> $(SEC_CRIT) ;
  /bin/sort                    -> $(SEC_CRIT) ;
  /bin/stty                    -> $(SEC_CRIT) ;
  /bin/su                      -> $(SEC_CRIT) ;
  /bin/sync                     -> $(SEC_CRIT) ;
  /bin/tar                     -> $(SEC_CRIT) ;
  /bin/usleep                   -> $(SEC_CRIT) ;
  /bin/vi                      -> $(SEC_CRIT) ;
  /bin/vimtutor                 -> $(SEC_CRIT) ;
  /bin/zcat                     -> $(SEC_CRIT) ;
}

############################### #
# Critical Utility Sym-Links # #
###############################
(
  rulename = "Critical Utility Sym-Links",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /sbin/clock                   -> $(SEC_CRIT) ;
  /sbin/ipfwadm-wrapper              -> $(SEC_CRIT) ;
  /sbin/kallsyms                -> $(SEC_CRIT) ;
  /sbin/ksyms                   -> $(SEC_CRIT) ;
```

```
  /sbin/lsmod                -> $(SEC_CRIT) ;
  /sbin/modprobe             -> $(SEC_CRIT) ;
  /sbin/mount.smb            -> $(SEC_CRIT) ;
  /sbin/mount.smbfs          -> $(SEC_CRIT) ;
  /sbin/pidof                -> $(SEC_CRIT) ;
  /sbin/poweroff             -> $(SEC_CRIT) ;
  /sbin/quotaoff             -> $(SEC_CRIT) ;
  /sbin/raid0run             -> $(SEC_CRIT) ;
  /sbin/raidhotadd           -> $(SEC_CRIT) ;
  /sbin/raidhotremove        -> $(SEC_CRIT) ;
  /sbin/raidstop             -> $(SEC_CRIT) ;
  /sbin/rdump.static         -> $(SEC_CRIT) ;
  /sbin/rrestore             -> $(SEC_CRIT) ;
  /sbin/rrestore.static      -> $(SEC_CRIT) ;
  /sbin/swapoff              -> $(SEC_CRIT) ;
  /sbin/rdump                -> $(SEC_CRIT) ;
  /sbin/reboot               -> $(SEC_CRIT) ;
  /sbin/rmmod                -> $(SEC_CRIT) ;
  /sbin/telinit              -> $(SEC_CRIT) ;
  /bin/awk                   -> $(SEC_CRIT) ;
  /bin/dnsdomainname         -> $(SEC_CRIT) ;
  /bin/domainname            -> $(SEC_CRIT) ;
  /bin/ex                    -> $(SEC_CRIT) ;
  /bin/gtar                  -> $(SEC_CRIT) ;
  /bin/nisdomainname         -> $(SEC_CRIT) ;
  /bin/red                   -> $(SEC_CRIT) ;
  /bin/rvi                   -> $(SEC_CRIT) ;
  /bin/rview                 -> $(SEC_CRIT) ;
  /bin/view                  -> $(SEC_CRIT) ;
  /bin/ypdomainname          -> $(SEC_CRIT) ;
}


########################## #
# Temporary directories # #
##########################
(
  rulename = "Temporary directories",
  recurse = false,
  severity = $(SIG_LOW),
  emailto  = squid.admin@company.com
)
{
#  /usr/tmp                  -> $(SEC_INVARIANT) ;
#  /var/tmp                  -> $(SEC_INVARIANT) ;
  /tmp                       -> $(SEC_INVARIANT) ;
```

```
}

################ #
# Local files # #
################
(
  rulename = "User binaries",
  severity = $(SIG_MED),
  emailto  = squid.admin@company.com
)
{
  /sbin                         -> $(SEC_BIN) (recurse = 1) ;
  /usr/local/bin                   -> $(SEC_BIN) (recurse = 1) ;
  /usr/sbin                     -> $(SEC_BIN) (recurse = 1) ;
  /usr/bin                      -> $(SEC_BIN) (recurse = 1) ;
}

(
  rulename = "Shell Binaries",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /bin/bash                      -> $(SEC_BIN) ;
  /bin/ksh                      -> $(SEC_BIN) ;
  /bin/tcsh                     -> $(SEC_BIN) ;
}

(
  rulename = "Security Control",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /etc/group                    -> $(SEC_CRIT) ;
  /etc/security/                -> $(SEC_CRIT) ;
  /var/spool/cron/root             -> $(SEC_CRIT) ;
}

# (
  # rulename = "Boot Scripts",
  # severity = $(SIG_HI)
# )
# {
  # /etc/rc                     -> $(SEC_CONFIG) ;
  # /etc/rc.local                -> $(SEC_CONFIG) ;
```

```
#  /etc/rc.tcpip                    -> $(SEC_CONFIG) ;
#  /etc/trcfmt.Z                    -> $(SEC_CONFIG) ;
#}

( rulename = "Proxy Cfg File",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /etc/squid/squid.conf            -> $(SEC_CONFIG) ;
  /etc/squid/mime.conf             -> $(SEC_CONFIG) ;
  /etc/privoxy/config              -> $(SEC_CONFIG) ;
  /etc/privoxy/default.action      -> $(SEC_CONFIG) ;
  /etc/privoxy/default.filter      -> $(SEC_CONFIG) ;
  /etc/privoxy/standard.action     -> $(SEC_CONFIG) ;
  /etc/privoxy/trust               -> $(SEC_CONFIG) ;
  /etc/privoxy/user.action         -> $(SEC_CONFIG) ;
}


(
  rulename = "Login Scripts",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
  /etc/csh.cshrc                   -> $(SEC_CONFIG) ;
  /etc/csh.login                   -> $(SEC_CONFIG) ;
  /etc/profile                     -> $(SEC_CONFIG) ;
}

# Libraries
(
  rulename = "Libraries",
  severity = $(SIG_MED),
  emailto  = squid.admin@company.com
)
{
  /usr/lib                         -> $(SEC_BIN) ;
  /usr/local/lib                   -> $(SEC_BIN) ;
}

########################################################## #
#                                          # #
# Critical System Boot Files                    # #
# These files are critical to a correct system boot. # #
```

```
#                                     ##
#######################################################

(
  rulename = "Critical system boot files",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
    /boot                       -> $(SEC_CRIT) ;
    /sbin/lilo                   -> $(SEC_CRIT) ;
    !/boot/System.map ;
    !/boot/module-info ;

    # other boot files may exist.  Look for:
    #/ufsboot                    -> $(SEC_CRIT) ;
}
  ####################################################
  ####################################################
  # These files change every time the system boots ##
  ####################################################
(
  rulename = "System boot changes",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
    !/var/run/ftp.pids-all ; # Comes and goes on reboot.
    !/root/.enlightenment ;
    /dev/log                    -> $(SEC_CONFIG) ;
    /dev/cua0                   -> $(SEC_CONFIG) ;
    # /dev/printer              -> $(SEC_CONFIG) ; # Uncomment if you have a printer
device
    /dev/console                -> $(SEC_CONFIG) -u ; # User ID may change on console
login/logout.
    #/dev/tty2                  -> $(SEC_CONFIG) ; # tty devices
    /dev/tty3                   -> $(SEC_CONFIG) ; # are extremely
    /dev/tty4                   -> $(SEC_CONFIG) ; # variable
    /dev/tty5                   -> $(SEC_CONFIG) ;
    /dev/tty6                   -> $(SEC_CONFIG) ;
    /dev/urandom                -> $(SEC_CONFIG) ;
    /dev/initctl                -> $(SEC_CONFIG) ;
    /var/lock/subsys            -> $(SEC_CONFIG) ;
    /var/lock/subsys/random     -> $(SEC_CONFIG) ;
    /var/lock/subsys/network    -> $(SEC_CONFIG) ;
    /var/lock/subsys/syslog     -> $(SEC_CONFIG) ;
```

```
    /var/lock/subsys/atd            -> $(SEC_CONFIG) ;
    /var/lock/subsys/crond          -> $(SEC_CONFIG) ;
    /var/lock/subsys/sendmail       -> $(SEC_CONFIG) ;
    /var/lock/subsys/anacron        -> $(SEC_CONFIG) ;
    /var/lock/subsys/keytable       -> $(SEC_CONFIG) ;
    /var/run                    -> $(SEC_CONFIG) ; # daemon PIDs
    /var/log                    -> $(SEC_CONFIG) ;
    /etc/issue.net              -> $(SEC_CONFIG) -i ; # Inode number changes
    /etc/ioctl.save             -> $(SEC_CONFIG) ;
    /etc/issue                  -> $(SEC_CONFIG) ;
    /etc/.pwd.lock              -> $(SEC_CONFIG) ;
    /etc/mtab                   -> $(SEC_CONFIG) -i ; # Inode number changes on any
mount/unmount
    /lib/modules                -> $(SEC_CONFIG) ;
}

# These files change the behavior of the root account
(
 rulename = "Root config files",
 severity = 100,
 emailto  = squid.admin@company.com
)
{
    /root                       -> $(SEC_CRIT) ;
    /root/.tcshrc               -> $(SEC_CONFIG) ;
    /root/.mcoprc               -> $(SEC_CONFIG) ;
    /root/.cshrc                -> $(SEC_CONFIG) ;
    /root/.bashrc               -> $(SEC_CONFIG) ;
    /root/.bash_profile         -> $(SEC_CONFIG) ;
    /root/.bash_logout          -> $(SEC_CONFIG) ;
    /root/.bash_history         -> $(SEC_CONFIG) ;
    /root/.Xresources           -> $(SEC_CONFIG) ;
    /root/.Xauthority           -> $(SEC_CONFIG) -i ;
}

 ################################
 #                              #
################################ #
#                              # #
# Critical configuration files # #
#                              # #
################################
(
 rulename = "Critical configuration files",
 severity = $(SIG_HI),
 emailto  = squid.admin@company.com
```

```
)
{
    /etc/crontab                -> $(SEC_BIN) ;
    /etc/cron.hourly            -> $(SEC_BIN) ;
    /etc/cron.daily             -> $(SEC_BIN) ;
    /etc/cron.weekly            -> $(SEC_BIN) ;
    /etc/cron.monthly           -> $(SEC_BIN) ;
    /etc/default                -> $(SEC_BIN) ;
    /etc/fstab                  -> $(SEC_BIN) ;
    /etc/exports                -> $(SEC_BIN) ;
    /etc/group-                 -> $(SEC_BIN) ;  # changes should be infrequent
    /etc/host.conf              -> $(SEC_BIN) ;
    /etc/hosts.allow            -> $(SEC_BIN) ;
    /etc/hosts.deny             -> $(SEC_BIN) ;
    /etc/protocols              -> $(SEC_BIN) ;
    /etc/services               -> $(SEC_BIN) ;
    /etc/rc.d/init.d            -> $(SEC_BIN) ;
    /etc/rc.d                   -> $(SEC_BIN) ;
    /etc/mail.rc                -> $(SEC_BIN) ;
    /etc/motd                   -> $(SEC_BIN) ;
    /etc/passwd                 -> $(SEC_CONFIG) ;
    /etc/passwd-                -> $(SEC_CONFIG) ;
    /etc/profile.d              -> $(SEC_BIN) ;
    /var/lib/nfs/rmtab          -> $(SEC_BIN) ;
    /usr/sbin/fixrmtab          -> $(SEC_BIN) ;
    /etc/rpc                    -> $(SEC_BIN) ;
    /etc/sysconfig              -> $(SEC_BIN) ;
    /etc/nsswitch.conf          -> $(SEC_BIN) ;
    /etc/yp.conf                -> $(SEC_BIN) ;
    /etc/hosts                  -> $(SEC_CONFIG) ;
    /etc/xinetd.conf            -> $(SEC_CONFIG) ;
    /etc/inittab                -> $(SEC_CONFIG) ;
    /etc/resolv.conf            -> $(SEC_CONFIG) ;
    /etc/syslog.conf            -> $(SEC_CONFIG) ;

}

##################### #
# Critical devices # #
#####################
(
  rulename = "Critical devices",
  severity = $(SIG_HI),
  recurse = false,
  emailto  = squid.admin@company.com
)
```

```
{
    /dev/kmem                   -> $(Device) ;
    /dev/mem                    -> $(Device) ;
    /dev/null                   -> $(Device) ;
    /dev/zero                   -> $(Device) ;
    /proc/devices               -> $(Device) ;
    /proc/net                   -> $(Device) ;
    /proc/sys                   -> $(Device) ;
    /proc/cpuinfo               -> $(Device) ;
    /proc/modules               -> $(Device) ;
    /proc/mounts                -> $(Device) ;
    /proc/dma                   -> $(Device) ;
    /proc/filesystems           -> $(Device) ;
    /proc/pci                   -> $(Device) ;
    /proc/interrupts            -> $(Device) ;
    /proc/ioports               -> $(Device) ;
    /proc/scsi                  -> $(Device) ;
    /proc/kcore                 -> $(Device) ;
    /proc/self                  -> $(Device) ;
    /proc/kmsg                  -> $(Device) ;
    /proc/stat                  -> $(Device) ;
    /proc/ksyms                 -> $(Device) ;
    /proc/loadavg               -> $(Device) ;
    /proc/uptime                -> $(Device) ;
    /proc/locks                 -> $(Device) ;
    /proc/version               -> $(Device) ;
    /proc/mdstat                -> $(Device) ;
    /proc/meminfo               -> $(Device) ;
    /proc/cmdline               -> $(Device) ;
    /proc/misc                  -> $(Device) ;
}

# Rest of critical system binaries
(
  rulename = "OS executables and libraries",
  severity = $(SIG_HI),
  emailto  = squid.admin@company.com
)
{
    /lib                        -> $(SEC_BIN) ;
}
```