GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

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

Snort Intrusion Detection System Audit: An Auditor's Perspective

GSNA Practical Version 2.1, March 2003
Jason Trudel

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Table of Contents

1 Assignment 1: Research in Audit, Measurement Practice, and Control 5
1.1 Introduction...5
1.2 Why Audit ACME?...5
1.3 Snort: What is it all about?..5
1.4 ACME's Defense: An In-Depth Explanation...5
1.5 System to be Audited..6
1.6 Risks to the System 9
1.7 Current state of practice...11
2 Assignment 2: Audit Checklist...13
2.1 Checklist Item 1 - IDS Policy:..13
2.2 Checklist Item 2 - IDS Procedure 14
2.3 Checklist Item 3 - Change Control...16
2.4 Checklist Item 4 - Physical Security 17
2.5 Checklist Item 5 - Hardware Redundancy 19
2.6 Checklist Item 6 - IDS Operating System Secured 20
2.7 Checklist Item 7 - Time Synchronization..21
Checklist Item 8 - Time Synchronization (NTP initialization)..23
2.8 Checklist Item 9 - Interfaces...24
2.9 Checklist Item 10 - Interfaces Initialization...25
2.10 Checklist Item 11 - SSH Daemon..26
2.11 Checklist Item 12 - SSH Initialization and Configuration 28
2.12 Checklist Item 13 - IDS Internal Interface...29
2.13 Checklist Item 14 - Snort Active 30
2.14 Checklist Item 15 - Snort Daemon Initialization and Configuration...........................31
2.15 Checklist Item 16 - Snort Backups 32
2.16 Checklist Item 17 - Snort Signatures 34
2.17 Checklist Item 18 - Snort Signature Update...35
2.18 Checklist Item 19 - Snort Performance...36
2.19 Checklist Item 20 - Snort Processing..37
2.20 Checklist Item 21 - Snort Attack Recognition...38
3 Assignment 3: Audit Evidence...46
3.1 Checklist Item 1 - IDS Policy – Pass (with comments)...46
3.2 Checklist Item 2 - IDS Procedure - Fail 47
3.3 Checklist Item 4 - IDS Physical Security – Pass 47
3.4 Checklist Item 7 - Time Synchronization - NTP – Pass 48
3.5 Checklist Item 9 - Interfaces – Pass 48
3.6 Checklist Item 11 - SSH Daemon – Fail 49
3.7 Checklist Item 15 - Snort - Initialization & Configuration - Pass................................49
3.8 Checklist Item 18 - Snort - Signature Update – Fail...49
3.9 Checklist Item 20 - Snort - Processing - Pass...50
3.10 Checklist Item 21 - Snort - Attack Recognition – Pass 51
3.11 Measure Residual Risk...53
3.12 Is the System Auditable 54
4 Assignment 4: Audit Report or Risk Assessment...55
4.1 Executive Summary..55
4.2 Audit Report..55
4.3 Summary...59
5 Appendices 60
5.1 Appendix 1 – Rule updater..60
6 References 62

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

List of Tables

Table 1 Risk Chart 9
Table 2 Results of Audit 46

List of Checklist Items

Checklist Item 1 IDS Policy 13
Checklist Item 2 IDS Procedure 14
Checklist Item 3 Change Control 16
Checklist Item 4 Physical Security 17
Checklist Item 5 IDS Hardware Redundancy 19
Checklist Item 6 IDS Operating System Secured 20
Checklist Item 7 Time Synchronization - NTP 21
Checklist Item 8 Time Synchronization – NTP initialization 23
Checklist Item 9 Interfaces 24
Checklist Item 10 Interfaces Initialization 25
Checklist Item 11 SSH Daemon 26
Checklist Item 12 SSH Initialization and Configuration 28
Checklist Item 13 IDS Administrative Interface 29
Checklist Item 14 Snort Active 30
Checklist Item 15 Snort Daemon Starting Configuration 31
Checklist Item 16 Snort Backups 33
Checklist Item 17 Snort Signatures 34
Checklist Item 18 Snort Signature Update 35
Checklist Item 19 Snort Performance 36
Checklist Item 20 Snort Processing 37
Checklist Item 21 Snort Attack Recognition 38

List of Audit Files and Results

Audit Result 1: Intrusion Detection System Policy 46
Audit Result 2: ps –ef | grep ntpd 48
Audit Result 3: ntpq –n –c rv 48
Audit Result 4 : ifconfig -a 48
Audit Result 5: ps –ef | sshd 49
Audit Result 6: ssh -V 49
Audit Result 7: cat /etc/rc.d/rc.inet2 49
Audit Result 8: ps -efl | grep snort 50
Audit Result 9: kill -HUP <pid> 50
Audit Result 10: cat /var/log/syslog 50

List of Simulated Attacks

Simulated Attack 1 IIS .HTR overflow Nessus Plugin ID: 11028 40
Simulated Attack 2 IIS Dangerous Sample files Nessus Plugin ID: 10370 41
Simulated Attack 3 IIS Directory Traversal Nessus Plugin ID: 10537 42
Simulated Attack 4 IIS 5.0 Malformed HTTP Printer Request Nessus Plugin ID: 10657 43
Simulated Attack 5 Socket80 44
Simulated Attack 6 Nmap 44

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

List of Figures

Figure 1: Visio Diagram on Layout of the Network System..8
Figure 2 : Nessus 51

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1 Assignment 1: Research in Audit, Measurement Practice, and Control

1.1 Introduction

This paper is to demonstrate the procedure for doing an independent audit on an Intrusion Detection System (IDS). It will be useful as a guide to anyone who is researching or conducting an IDS audit or System Administrators who need to prepare for an upcoming audit of their systems or to carry one out on their own.

1.2 Why Audit ACME?

The company ACME Inc. has hired me to audit their IDS running Snort[1], as they have not been happy about a recent compromise of a production system. This system is the first line of defense for monitoring in real time; therefore ACME's Time Based Security depends on it. Time Based Security is the time that it takes to recognize an attack, alert on it, and have it passed on to the Incident Handling team to the time it takes to actually carry out the attack and compromise a system or cause harm in the environment.

With their idea of Time Based Security, a compromise of this sort should have been detected and stopped before any damage was done.

1.3 Snort: What is it all about?

According to searchsecurity.com "Snort is an open source network intrusion detection system (NIDS) created by Martin Roesch. Snort is a packet sniffer that monitors network traffic in real time, scrutinizing each packet closely to detect a dangerous payload or suspicious anomalies.

Snort is based on *libpcap* (for library packet capture), a tool that is widely used in TCP/IP traffic sniffers and analyzers. Through protocol analysis and content searching and matching, Snort detects attack methods, including denial of service, buffer overflow, CGI attacks, stealth port scans, and SMB probes. When suspicious behavior is detected, Snort sends a real-time alert to *syslog*, a separate 'alerts' file, or to a pop-up window."

1.4 ACME's Defense: An In-Depth Explanation

ACME believes in defense in depth, their web servers sit on a Demilitarized Zone (DMZ) behind a firewall, which is connected to the Internet by a Supporting Router. The router is the first line of defense with Access Control Lists (ACLs) to limit any unwanted traffic (according to ACME internet policy) from ever hitting the firewall. The firewall further protects the servers behind it by limiting connections to certain servers on specific ports. Next we have a Network-based Intrusion Detection System and further each server has a

[1] *Snort Intrusion Detection System – http://www.snort.org*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Host based Integrity Checking System which only runs nightly. The part of this that ACME wants us to look at is the NIDS. Specifically it is a server running Slackware Linux[2], and the powerful IDS Snort.

Based on my SANS[3] training in Auditing Networks, Perimeters, and Systems, and some experience we will look at the steps needed do a complete and useful audit of this system.

1.5 System to be Audited

The scope of this audit will be conducted in two different stages:

- Review of Policies and Procedures (Time required: 2 days)
- Audit of the server system (Time required: 2 days)

1. Review the ACME DMZ IDS policy and ACME DMZ IDS procedure

   This includes an extensive review of the operation of Snort in the DMZ environment including proper configuration of the software, rule set and logs/alerts. Care will be taken to see if it is proper accordance to the ACME DMZ IDS policy. If any obvious problems are sighted with these documents, then the systems they are designed to be guidelines for is sure to have problems. The server and OS that Snort resides on are secured using the ACME Secure Server Build (SSB). This document has to be followed and signed off by an administrator that builds the server to ensure steps were followed that includes best practice according to ACME for secure Linux based systems.

2. Audit of the system

   a. Day 1: Interview system administrator to get basic server information.
   b. Day 2: Launch attacks and pull the IDS logs to analyze the information gathered. (This will assume that all upstream components are configured correctly and hardened to at least industry standards.)

Requirements for this include a dummy server on the "sniffing segment" to point our attacks, so we do not harm any production servers and a machine to carry out the attacks. This will be done with two laptops provided by the auditor.

ACME provided us this inventory of the server be audited. It is a physically secured machine running Slackware 8.1, Linux kernel 2.4.17 and Snort version 1.8.1 on PIII 800 MHz machine with 512MB of RAM, dual 9.1GB SCSI drives with hardware RAID 1[4] configuration and dual network interface cards. The first interface, eth0 is connected to the Production segment, listening only on the Secure Shell[5] (SSH) - Port 22, to act as the administration access portal and on the Network Time Protocol[6] (NTP) – Port 123, used for system time synchronization with the company's NTP infrastructure. The second

[2] *Linux Slackware Distribution - http://www.slackware.org*
[3] *Systems Administrator and Network Security - http://www.sans.org*
[4] *RAID - http://www.webopedia.com/TERM/R/RAID.html*
[5] *Secure Shell – http://www.openssh.org*
[6] *Network Time Protocol – http://www.ntp.org*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

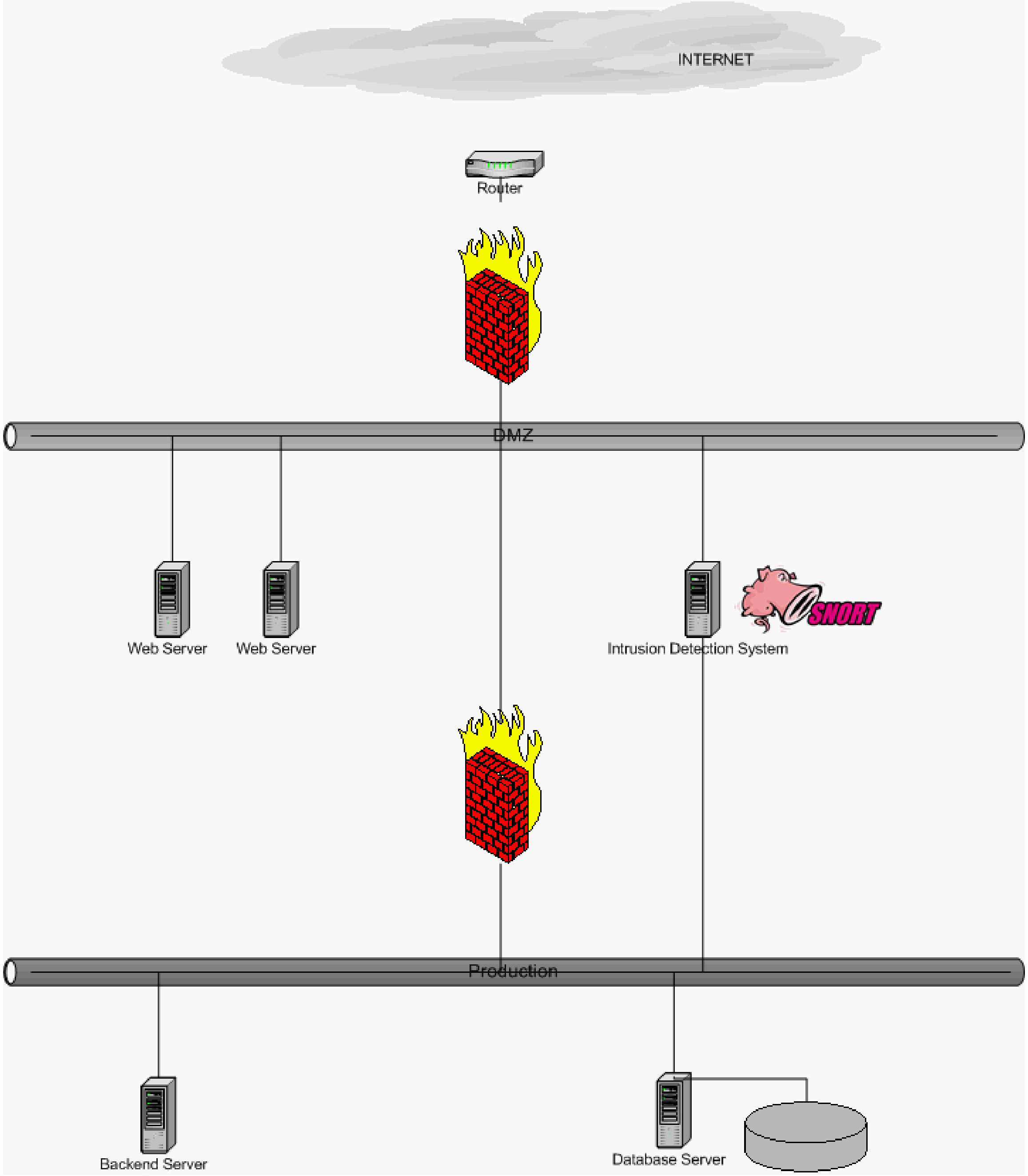
interface eth1 is the sniffing interface that is plugged into a mirror port on the DMZ switch, running promiscuous mode with no IP address to eliminate anyone connecting to, or detecting our "sniffer box". Snort will be analyzing both incoming and outgoing traffic, looking for patterns (signatures) that match known attack methods and malicious traffic.

The layout of the system is as follows:

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Figure 1: Visio Diagram on Layout of the Network System



© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.6 Risks to the System

The NIDS we will be looking at has many functions and is an integral part of ACME's layered security. There are many risks that go along with depending on any piece of hardware or software. These risks could potentially render it useless while other risks involved could mean that the system is not being used as efficiently as it could be. Some of the risks with this system are:

The following chart is divided into these columns: Risk, Chance, Consequences, and Severity.

- **Risk** – a risk that is considered relevant to this system
- **Chances** – the chance of the risk actually happening, 1 being least likely and 10 having a very high probability of happening
- **Consequences** – the results to the system/environment should one of these risks be carried out.
- **Severity** – the severity of the previous consequences to the system/environment, 1 being low priority and 10 being total system or environment compromise.

Table 1 Risk Chart

Risk	Chances	Consequences	Severity	Overview of Audit Strategy
Policy and procedure not adequate or non-existent	5	If policy and procedure are not done properly, tasks of the system might not be defined properly and procedures carried out on this system may be incorrect	6	Confirm existence of policy and procedure documentation and review to determine effectiveness of each
Hardware failure	7	All monitoring by NIDS would be halted, all functions of system would not be working	10	Confirm that critical systems/hardware are redundant
NIDS being compromised by hacker	1	Any data or alerts from the system could not be trusted, server could be used for further attacks	10	Is system in a physically secure area? Have sufficient actions been taken to secure server on

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

				network
Missing attacks or virus signatures (false negatives)	6	Active attacks on servers could go unnoticed or delayed making Time Based Security less effective	7	Random attacks on test server, confirm that IDS and Analyst report them
Alerting on valid traffic/network activity (false positives)	6	Too many false positives could make the IDS Analyst start to ignore	7	Confirm that false positives are at a minimum and Analyst can handle all alerts without disregarding valid alerts.
Packet loss (server not powerful enough)	2	Active attacks on servers could go unnoticed or delayed making Time Based Security less effective	6	Snort and Linux have statistics built in. We can check these for 0% packet loss.
ACME relies on NIDS too heavily (Only relies on IDS logs/alerting to find attacks)	2	Other anomalies on network could go unnoticed or new attacks could be used to evade IDSs, would give a false sense of security	5	Check overview of layered security. How does the NIDS fit into the corporate security?
Proper analyst being alerted, reviewing logs, and following procedure in the case of an incident	7	If policy & procedure are not followed there could be a security breakdown	5	Check policy & procedure, does anyone get called or paged if alert logged?
Important files not backed up	3	Important files (configuration / rule sets) could be lost if server is not backed up properly	9	Proof of important files backed up, either backup software or some scheduled backup job

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

				job
Unauthorized users able to log into machine or have more access than they need	6	Only users that need access to the system should have access. Also users that do need access should only have enough permission to do their job	7	Confirm that only legitimate users have access and permissions are set accordingly

1.7 Current state of practice

The current state of practice for an IDS audit is scarce. The Center for Internet Security (CIS)[7] does not have a listing for a standard IDS audit. Looking at Snort documentation we get a detailed view of what the system can be configured to do, and its strengths and weaknesses. For this audit research into the setup and configuration of Snort the modes of operation and touching on writing Snort rules will give us a good base to go on. Using the major search engines did not come up with any exact hits but a lot of information on auditing and IDSs but not too much with the two together. On the GIAC[8] site there was an excellent practical assignment posted on auditing a distributed IDS (www.giac.org/practial/Darrin_Wassom_GSNA.doc).

From these many gave great examples and layouts of IDS systems that we used as our ammunition to compare and rate ACME's IDS accordingly. With this knowledge we can apply it to a "standard" audit approach. By this I mean that we can get to the basics of auditing and get a thorough, useful audit of this system. By the end we should have a checklist specifically designed for an IDS system that will make future audits on these types of systems more efficient. There will still not be any checklist that will fit all systems, but a base can be established that an auditor can work from to get a complete and useful audit of their particular system.

Our plan of attack for this audit will be to use the 6 - Step process taught in SANS – Auditing Principles and concepts.

These steps include:

- Planning
- Strategy
- Entrance Conference
- Fieldwork
- Report
- Exit Conference

[7] *The Center for Internet Security (CIS)*[7] *http://www.cisecurity.org*
[8] *Global Information Assurance Certification http://www.giac.org/cert.php*

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

We will also incorporate some basic Linux auditing principles since this is running on the Linux platform and there is no need for us to rewrite this. There are many great papers written on Linux hardening and Linux auditing available on the GIAC website.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

2 Assignment 2: Audit Checklist

2.1 Checklist Item 1 - IDS Policy:

IDS Policy is the document that provides a guide to what the system is trying to accomplish and who is going to get it there.

Checklist Item 1 IDS Policy

Reference	General practice - Company should have IDS Policy documentation on hand so we have a reference on what is expected from this system The SANS Security Policy Project lists very useful guidelines that can be customized to most situations. http://www.sans.org/resources/policies/ IT security policies. http://www.network-and-it-security-policies.com/ http://www.metasecuritygroup.com/policy.html
Control Objective	Confirm that policy exists and is adequate so proper procedure can be used to satisfy company needs.
Risk	IDS might not be used as stated in company policy or the policy might not encompass everything needed to be complete. This could give a false sense of security if one thing is expected from the system and since policy wasn't followed it is doing another. Procedure might also be useless if policy in inadequate.
Compliance	Based on what is expected from the IDS, does policy cover all points? Some of the policy might be okay but others might not fit into what is expected or part might be missing altogether. We want to see a defined scope and who is responsible for certain tasks. An explanation of these is also very useful. Based on the range of responses for this item we might not make it an exception if there isn't total compliance. As long as the policy meets the current needs, it could pass but with comments on how to improve on it to fix items that don't

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	fit in properly or that are too vague or missing.
Testing	Obtain a copy of company's IDS Policy • Does it answer the questions who? And what? • Is it clear and firm? • Interview the Chief Security Officer (CSO) or who is responsible for policy, to determine whether it covers purposes intended for this system. • Some questions to ask o What is expected of this procedure document o Have the employees been informed of the existence and location of the policy document? o What exactly is the purpose and expected operation of the system covered by this policy?
Objective / Subjective	Subjective
Pass / Fail	

2.2 Checklist Item 2 - IDS Procedure

Procedure is derived from Policy. This is the document that provides the detailed steps to accomplish Policy. This is very useful to ensure all systems are documented and is easy to follow and not left open to interpretation.

Checklist Item 2 IDS Procedure

Reference	General practice Company should have Procedure to compliment Policy so a known and expected operation can be taken for a specific system
Control Objective	Confirm that policy exists, is accessible and is adequate so proper actions are being taken

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	with this system to satisfy company needs. We want to prove that the procedure document is easy to follow and not be left open to interpretation. Employees must know where this document is kept and is used when needed.
Risk	If we don't find a useful procedure, IDS might not be used to accomplish what is stated in policy. It is very difficult to know what is going on with system if there aren't any guidelines or rules people have followed for certain events. This could render the IDS useless or even worse open it up to attack. With documentation sometimes be neglected the chance of deficient procedures is a medium risk
Compliance	There are a few levels we could grade this. First, a valid procedure should exist. It must be sufficient to achieve what is needed. Employees must know where it is, and it must be used.
Testing	Obtain a copy of company's IDS Procedures Interview Analyst and/or Administrator and get information to determine whether Procedure is followed. Change Control documents for a recent task would be sufficient. We will review the Procedure to determine if it is written clearly and not left open for interpretation. Our interview will be a short, informal talk to determine if Procedure is accessible to the employees and followed by the employees. o Do they have general knowledge of how to access procedure documents? o Have they been shown how to use them? o Are the steps easy to follow and complete? o There should not be any ambiguous words like "should" or "may", these could leave steps open
Objective / Subjective	Objective - Procedure exists Subjective – Procedure is adequate and followed, unless proof can be shown e.g. signed off recent Change Control documents etc…

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Pass / Fail	

2.3 Checklist Item 3 - Change Control

Change control is simply the tracking and management of changes made to a system. This can include things from authorization forms/procedures to final sign-off and audit of systems.

Checklist Item 3 Change Control

Reference	COBIT http://www.isaca.org/standard/iscontrl.htm Example of Standards and Procedures http://www.uky.edu/~change/sp.html Experience
Control Objective	We want to determine that the company has clear and concise documentation of the steps taken for software and hardware upgrades done on this server. The changes should be done in organized and consistent method. Also we want to validate that audit trails exist.
Risk	Upgrades to software or hardware that are not done in a controlled environment could render the system useless. Also someone not familiar with the system must be able to find out what has been done to it. If there is a problem and the person that set it up is not present, troubleshooting can be very difficult if there is no record to what has been done to the system. There is a very good chance that this could actually happen. In the heat of a crisis, system changes could go undocumented or someone with the attitude of "this one little change won't affect anything" and skip the sometimes-timely change control steps.
Compliance	This will depend greatly on the company's change control methods. It could range from greatly documented and complete signoff on changes and upgrades to no documentation or audit and changes made in an ad-hoc fashion. We must see proof of one of these being used. This will include a completed form of the last change done to the system or an audit trail that

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	outlines that the steps in the procedure were being followed. Log entries, signed checklists, etc…
Testing	Review and gauge change control documentation and sign off forms (if any exist). Clear and concise steps must be outlined to audit or document procedural changes to software and hardware upgrades and changes on this system. Ask to see a completed document or audit trail of a recent change to the system.
Objective / Subjective	Subjective
Pass / Fail	

2.4 Checklist Item 4 - Physical Security

Physical security can mean many things. From location of your building to a server in a locked rack, physical security is just as important as other security measures in place.

Checklist Item 4 Physical Security

Reference	http://www.sans.org/rr/physical/protect.php Personal experience
Control Objective	Determine how physical access to this server is controlled. From entering the building to sitting down at the console, security measures in place to protect this critical system will be documented. We want to determine if sufficient guards are being taken.
Risk	If an attacker/unauthorized user had access to the console of the machine it would be compromised and/or rendered useless. Physical damage could also be done to the system making it unusable until it could be replaced. This would make our time based security

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	model even more difficult to sustain. This risk is quite high, maybe not particular systems being targeted, but easy money for a would-be thief if security is lax.
Compliance	The checklist item can have a range of results. The machine could in fact be secure, but there is always room for improvement. Even the most secure area could be vulnerable to Social Engineers or in unusual cases, extreme force. We will determine if sufficient steps have been taken to secure the system. It could be argued on how secure is secure, the auditor will use common sense and experience in rating the access to this machine.
Testing	Auditor will physically go to console of machine, taking detailed notes of what security measures are in place to protect the server. Things to look for: • Outside the building o Note location of building o Surroundings o Signs on building o Security cameras o Response times from emergency services o Open dumpsters o Propped open doors o Location of windows • Inside the building o Security guard o Security cameras o Card readers/locks/biometrics o Logs of visitors o Unlocked Doors o Secured elevators • Walking to the server room o Security cameras o Door access

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	○ Can access be "piggy backed" e.g. can you follow someone in an area without authenticating? (swiping card, signing in bio-metrics, etc…) • In the server room ○ Security cameras ○ Operators present ○ There shouldn't be any windows ○ Walls should go to solid floor if in raised-floor room and should go to roof or next floor above if there is a drop ceiling ○ Server in locked rack
Objective / Subjective	Subjective
Pass / Fail	

2.5 Checklist Item 5 - Hardware Redundancy

Hardware redundancy is just that, having two instead of one is always better. If a system is critical and it's power supply fails….well hope you have two :)

Checklist Item 5 IDS Hardware Redundancy

Reference	http://linux-ha.org/ http://www.zdnet.com.au/newstech/enterprise/story/0,2000025001,20267015-3,00.htm Experience
Control Objective	We want to prove that all hardware that can, be redundant or all preventive measures have been taken on this system.
Risk	This is a mission critical system; if we have a hardware failure in a component that is not redundant then the system will be inactive until that part is replaced. Hardware failure is

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	common and this is a high risk.
Compliance	There is a range of values for this item the server could have many different redundant parts. With the risk being high on this we don't want to see any single points of failure. A second machine that could replace this one in an outage is crucial.
Testing	Get a hardware inventory from the server to see what components are redundant. Check that standby server is ready if needed.
Objective / Subjective	Subjective
Pass / Fail	

2.6 Checklist Item 6 - IDS Operating System Secured

The Operating System is your foundation to build on. Start with a solid base to ensure proper security.

Checklist Item 6 IDS Operating System Secured

Reference	ACME Secure Server Build document. The Center for Internet Security has some great benchmarks for popular OSs. The Linux Benchmark document contains detailed instructions for implementing the steps necessary for CIS Level-I security. http://www.cisecurity.com/bench_linux.html Security Quick-Start HOWTO for Linux from www.linux.org http://www.linux.org/docs/ldp/howto/Security-Quickstart-HOWTO/index.html http://www.sans.org/rr/linux/sec_install.php General practice – any systems connected to a network should have followed secure server / hardening procedures. http://thaicert.nectec.or.th/event/itsec2002-material/Implementation.pdf

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Control Objective	Confirm that system OS matches up to a secure installation / configuration according to ACME Corporate Server Policy.
Risk	Server could be access or compromised by an unauthorized user. This could render the system useless.
Compliance	We are looking for this server to be signed off on a SSB form. This will be adequate for this audit to ensure the underlying operation system is secured to ACME's standards.
Testing	We will not be doing a full out OS Audit. To accomplish this step we have agreed with management that proof of signoff on a Secure Server Build (SSB) is adequate for us. ACME keeps a record of SSB that would be signed off by administrator that built the server. The signed SSB must be produced for this particular server.
Objective / Subjective	Objective
Pass / Fail	

2.7 Checklist Item 7 - Time Synchronization

Time synchronization is achieved by running clients that use the NTP protocol to provide accurate times compared to trusted sources such as radio or atomic clocks on the internet.

Checklist Item 7 Time Synchronization - NTP

Reference	http://www.eecis.udel.edu/~ntp/ntp_spool/html/ntpq.html comp.protocols.time.ntp newsgroup http://geodsoft.com/howto/timesync/ experience

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Control Objective	Verify accurate and consistent timing is available for logging. Also verify that it is configured to the correct time source as stated in ACME Policy.
Risk	If time on server were out, logs would have inconsistent entries with actual time logged. This could make it difficult to correlate with other system logs for incident response and forensics. Also like any service NTP has been a victim of exploits, current versions contain patches to protect the service.
Compliance	To comply we want to see that the NTP daemon is actually running at the time of the audit and version of NTP is current checked with the production version located on the NTP website. Also stratum must be < 6.
Testing	Run the following commands: **ps –ef | grep ntpd** to confirm if a NTP daemon is running. **ntpq –n –c rv** verify NTP is latest stable version "version=x.x" compared to the "production version" located http://www.eecis.udel.edu/~ntp/download.html "stratum=x" must be less than 6
Objective / Subjective	Objective

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Checklist Item 8 - Time Synchronization (NTP initialization)

It is important to first have the NTP daemon running and even more important to have it start properly to ensure accurate times on the server.

Checklist Item 8 Time Synchronization – NTP initialization

Reference	http://www.eecis.udel.edu/~ntp/ntpfaq/NTP-s-config.htm#AEN2516 Experience
Control Objective	Validate that NTP is started correctly in initialization script.
Risk	If NTP is not start correctly in an initialization script, the daemon could possibly not be started after a system restart or if configured wrong, not keep the server clock the right time.
Compliance	This can only be a pass or fail. We must see the ntp1.ACME.com as our time source server and it must be running and configured this way in an initialization script. The time should also be set at boot time with the ntpdate command.
Testing	View the /etc/rc.d/rc.init2 script using **less /etc/rc.d/rc.init2** and look for the line to start the ntp daemon. It should look like this: **/usr/sbin/ntpd** Confirm that it is configured to synchronize with **ntp1.ACME.com**. To do this check the file /etc/ntp.conf with the command **less /etc/ntp.conf** and verify that the time server line is correctly set to "server=ntp1.acme.com" Confirm that server time is set at boot time. Look at the /etc/rc.d/rc.init2 script using **less /etc/rc.d/rc.init2** and look for this: **/usr/sbin/ntpdate ntp1.ACME.com**
Objective / Subjective	Objective

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, Author retains full rights.

2.8 Checklist Item 9 - Interfaces

Having the interfaces configured properly is essential in the operation of any system.

Checklist Item 9 Interfaces

Reference	Snort documentation - http://www.snort.org/docs/ http://www.linux.org Experience
Control Objective	Determine if interfaces are configured properly on this server to be used for and IDS.
Risk	If interfaces have been configured incorrectly, Snort might not run as expected, and machine could be vulnerable to attack.
Compliance	The results will be compared to how the interfaces should be configured. There is no in between with this control, they must match to comply.

Testing	We will take the output from the command:
	ifconfig –a
	Are the interfaces configured as expected? We should see the management interface configured with an internal IP address (specified by ACME) on eth0. The "sniffing interface" should show up without an IP address and be in promiscuous mode.
	Example:
	`eth0      Link encap:Ethernet  HWaddr 00:02:A5:34:9E:9E`
	`          inet addr:10.1.30.25  Bcast:10.1.255.255  Mask:255.255.0.0`
	`          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1`
	`          RX packets:280667967 errors:0 dropped:0 overruns:296 frame:3`

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	TX packets:13578702 errors:0 dropped:0 overruns:76 carrier:0 collisions:0 txqueuelen:100 RX bytes:2800690976 (2670.9 Mb) TX bytes:530866821 (506.2 Mb) Interrupt:10 Base address:0x2000 eth1 Link encap:Ethernet HWaddr 00:00:D1:ED:27:C5 UP BROADCAST RUNNING PROMISCMULTICAST MTU:1500 Metric:1 RX packets:1473302476 errors:0 dropped:0 overruns:0 frame:0 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:100 RX bytes:1452656646 (1385.3 Mb) TX bytes:0 (0.0 b) Interrupt:11 Base address:0x4000
Objective / Subjective	Objective

2.9 Checklist Item 10 - Interfaces Initialization

It is important that the systems interfaces are configured correctly, even after the system restarts.

Checklist Item 10 Interfaces Initialization

Reference	Snort documentation - http://www.snort.org/docs/ http://www.slackware.com/config/init.php Experience
Control Objective	Determine if interfaces are configured properly in the initialization script to start on boot.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Risk	If interfaces have been configured to start properly, Snort might not run as expected, or interfaces might not come up at all. If networking interfaces are not running on this machine the IDS will be useless. The machine could also be vulnerable to attack if the interface is inaccurately configured.
Compliance	The results will be compared to how the interfaces should be configured. There is no in between with this control, they must match to comply.
Testing	We will view the file: /etc/rc.d/rc.inet1 using **less /etc/rc.d/rc.inet1**. The interfaces should be configured to start as follows; the management interface configured with an internal IP address (specified by ACME) on eth0; the "sniffing interface" should show up without an IP address and be in promiscuous mode.
Objective / Subjective	Objective

2.10 Checklist Item 11 - SSH Daemon

SSH is used to provide a secure alternative to telnet[9] and rlogin[10].

Checklist Item 11 SSH Daemon

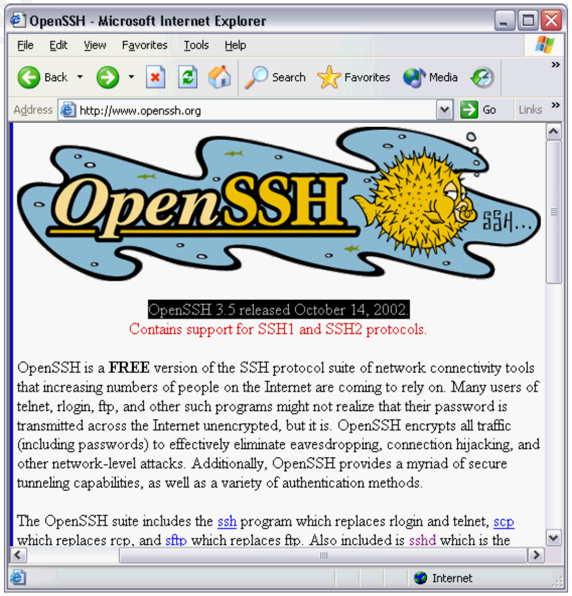
Reference	http://www.openssh.org experience
Control Objective	Confirm that there is a means of secure communication with the server to provide access for administration.
Risk	There have been many exploits[11] against the SSH protocol; an older version could be vulnerable to certain attacks. This could lead to system compromise. If the SSH daemon is

[9] Remote terminal emulation http://www.computeruser.com/resources/dictionary/definition.html?lookup=7021

[10] Remote login to Unix systems http://www.unidata.ucar.edu/cgi-bin/man-cgi?rlogin+1

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	vulnerable to certain attacks. This could lead to system compromise. If the SSH daemon is not running the administrator would not be able to connect to the server. Since there have recent exploits I would consider this a high risk, which would be quite likely to happen.
Compliance	We must see the /etc/sbin/sshd daemon running and be the latest stable/patched build compared to http://www.openssh.org
Testing	Run the commands: **ps –ef | grep sshd**; to confirm daemon is running. It should look like this: **/usr/sbin/sshd** **ssh –V** ; to view version of Secure Shell running  OpenSSH - Microsoft Internet Explorer File Edit View Favorites Tools Help Back Search Favorites Media Address http://www.openssh.org Go Links OpenSSH ssh... Internet OpenSSH 3.5 released October 14, 2002. Contains support for SSH1 and SSH2 protocols. OpenSSH is a **FREE** version of the SSH protocol suite of network connectivity tools that increasing numbers of people on the Internet are coming to rely on. Many users of telnet, rlogin, ftp, and other such programs might not realize that their password is transmitted across the Internet unencrypted, but it is. OpenSSH encrypts all traffic (including passwords) to effectively eliminate eavesdropping, connection hijacking, and other network-level attacks. Additionally, OpenSSH provides a myriad of secure tunneling capabilities, as well as a variety of authentication methods. The OpenSSH suite includes the ssh program which replaces rlogin and telnet, scp which replaces rcp, and sftp which replaces ftp. Also included is sshd which is the The latest version and date released can be found on the main page of

[11] *OpenSSH Security Advisory http://www.openssh.org/txt/trojan.adv*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	http://www.openssh.org (highlighted in picture)
Objective / Subjective	Objective

2.11 Checklist Item 12 - SSH Initialization and Configuration

The proper initialization of SSH is vital in a secure system.

Checklist Item 12 SSH Initialization and Configuration

Reference	http://www.openbsd.org/cgi-bin/man.cgi?query=sshd Experience
Control Objective	Verify that our means of secure communication with the server is configured to start whenever the server reboots.
Risk	If SSH is not started on reboot, the server won't be able to be securely, remotely accessed by the system administrator and Intrusion Analyst.
Compliance	We must see the /usr/sbin/sshd daemon configured to start on reboot in the /etc/rc.d/rc.inet1 initialization script.
Testing	We will view the file: /etc/rc.d/rc.inet1 the by using **less /etc/rc.d/rc.inet1** and review the script for the lines(s) that start the SSH daemon. It should look like this: **echo "Starting OpenSSH SSH daemon:"** **/etc/rc.d/rc.sshd**

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Objective / Subjective	Objective

2.12 Checklist Item 13 - IDS Internal Interface

The IDS internal interface provides administrative access to the system.

Checklist Item 13 IDS Administrative Interface

Reference	General practice – any systems connected to a network should have all unused services turned off. ACME Policy - Snort server should only be listening on SSH port, for secure administration and NTP port for time synchronization. NMAP port mapper http://www.insecure.org/nmap/index.html#intro An Introduction to NMAP http://www.sans.org/rr/audit/nmap2.php
Control Objective	Confirm that the server is only listening on necessary ports.
Risk	Server could be accessed or compromised by an unauthorized user
Compliance	In order for this system to comply we must only see a response from SSH on port 22 and NTP on port 123.
Testing	Using the network scanning tool Nmap, we will conduct a port scan against this server to find all open port the server is listening on. NMap scan with the following parameters: **nmap –vv –sS –O –P0 –oN <logfilenameTCP> -p 1-65535 –T Polite ids.ACME.com**

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	nmap –vv –sU –O –P0 –oN <logfilenameUDP> -p 1-65535 –T Polite ids.ACME.com Options in the scan include: **–vv** (very verbose) to get more information (auditors like lots of info) **-sS** TCP SYN scan **-sU** UDP scan **-O** provides us with remote host identification via TCP/IP fingerprinting **-P0** do not try and ping hosts at all before scanning them **-oN <logfilename>** the file we will save the scan to in **n**ormal **o**utput that we can later print out and add to our audit results **-p 1-65535** scan of all ports **-T Polite** canned timing policy in nmap; Polite is easy on the network and has less chance of crashing the machine being scanned **Ids.ACME.com** IP address or DNS name of system being scanned
Objective / Subjective	Objective

2.13 Checklist Item 14 - Snort Active

Snort must be running for it to be any use. This is the heart of our Intrusion Detection System.

Checklist Item 14 Snort Active

Reference	http://www.nevis.columbia.edu/cgi-bin/man.sh?man=ps Experience

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

Control Objective	Verify that the snort daemon is running on the system at the time of audit.
Risk	If the Snort daemon is not running then the IDS will not process any packets, log and your server is pretty much just sitting there doing nothing.
Compliance	Snort daemon must be running for compliance
Testing	Run command: **ps –efl | grep snort** Verify that Snort daemon is running, we should see: "…/usr/local/sbin/snort -c /usr/local/etc/snort_eth1.conf -d -D -i eth1 -I -l /var…"
Objective / Subjective	Objective

2.14 Checklist Item 15 - Snort Daemon Initialization and Configuration

Snort initialization is very important. This is how it is started and configured to run on our system.

Checklist Item 15 Snort Daemon Starting Configuration

Reference	http://www.slackware.org/config/init.php http://msbnetworks.net/snort/snortd.txt http://www.snort.org/docs/faq.html#2.1
Control Objective	Verify that snort is configured to start when the server is rebooted. In this step we will also check that snort is being started with the correct switches. Since Snort can be used so many ways the command line to start Snort is very important.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

Risk	If it was not setup for the Snort daemon to be started in initialization scripts, a server reboot or outage could cause Snort not start unless done manually.
Compliance	We want to see proof of the Snort daemon being started correctly during system startup. To be configured correctly and comply is must have the following included: **<full path to snort>**; must have absolute path to Snort binary. **-c <path to rules file>**; tells Snort what rules file to use. **-d**; tells Snort to dump packet payloads. **-D**; run in daemon mode. **-i eth1**; use interface eth1 configured for Snort. **-l <path to log file>**; log to a certain location.
Testing	We will view the file: /etc/rc.d/rc.inet2 by using **less /etc/rc.d/rc.inet2** and review the script for the lines(s) that start the Snort daemon. It should look like this: **echo "Starting snort..."** **/usr/local/sbin/snort -c /usr/local/etc/snort_eth1.conf -d -D -i eth1 -I -l /var/log/alert_eth1/**
Objective / Subjective	Objective

2.15 Checklist Item 16 - Snort Backups

Backing up any system is fundamental. They are the safeguards to fall back on if something really bad happens. From system and hardware crashes causing data corruption to quick fingered users who use the line "I think I needed that file I just deleted".

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

Checklist Item 16 Snort Backups

Reference	Experience http://www.pcguide.com/care/bu/exer-c.html http://www.iol.ie/~ecarroll/backuptale.html (kind of funny)
Control Objective	To prove the existence of a backup method to ensure that critical files will not be lost in the event of a failure.
Risk	If these files are not backed up and moved off the server, in the event of a server crash or data loss the time to recover would be much longer and with custom rule sets it might be impossible to recreate.
Compliance	These files are either getting backed up or not. Backups should then be securely moved off the server to another server or tape. The backups should include the snort.conf file and the snort.rules file.
Testing	Talk to the system administrator to find out what method is used for backup. Locate job either in crontab or backup software. Have administrator explain how script achieves the backups we require. Proof of backups and secure file transfer of the **/usr/local/etc/snort.rules** and **/usr/local/etc/snort.conf** files. As root get administrator to run **crontab -l** to check if there are any root cron jobs setup to copy files and move them. Obtain a copy of job that backs up and moves files. and/or Proof of backup software backing up the files. Obtain a screenshot or output of software listing the Snort configuration and rules files. The exact location of these files are: **/usr/local/etc/snort.rules** and **/usr/local/etc/snort.conf**

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Objective / Subjective	Objective

2.16 Checklist Item 17 - Snort Signatures

Our rules or signatures define what Snort should watch for including specific attacks and other suspicious traffic.

Checklist Item 17 Snort Signatures

Reference	SANS – Intrusion Detection Snort Style 3.3 pg 1-168 http://www.snort.org/snort-db/ ACME Policy
Control Objective	ACME Policy states that only attacks on port 80 and 443 will be considered, as these are the only two allowed through by the protecting firewall. Rules file should only contain valid attack signatures for these types of attacks. Up to date signatures must exist.
Risk	If Snort is overloaded with superfluous rules it could not only affect the performance of the server making it run slow, drop or not process packets but maintenance of the rules file is much more difficult. Missing attack signatures could lead to an unnoticed attack, meaning deteriorated Time Based Security and possible system compromises.
Compliance	Snort rules file will be viewed and verify that only web based signatures (port 80 and 443)
Testing	Check for unnecessary rules. run **cat /usr/local/etc/snort.rules** to view the configuration file. Compare what preprocessors and output plug-ins are found with what is need to be achieved in company policy. Can all requirements be met with the configuration found or could it be

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

	done with less.
Objective / Subjective	Subjective

2.17 Checklist Item 18 - Snort Signature Update

Since Snort relies on its rules file to match patterns, signatures of new exploits and attacks must be up to date.

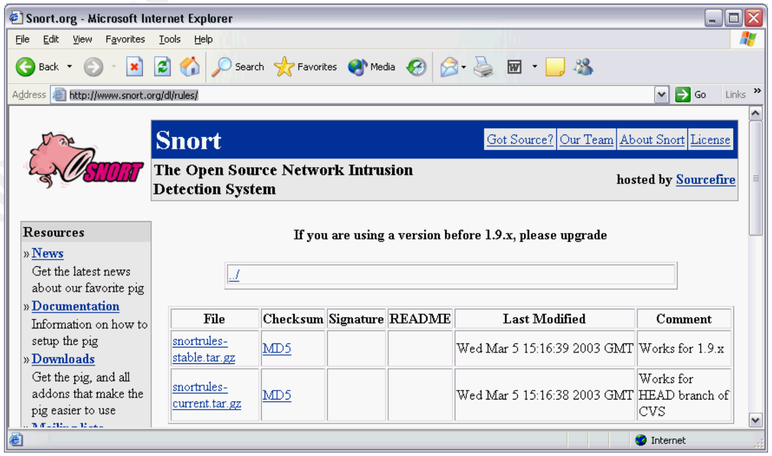
Checklist Item 18 Snort Signature Update

Reference	http://www.snort.org/dl/signatures/ http://www.snort.org/docs/faq.html#3.15 SANS Track 3 – Intrusion Detection In Depth[12] Experience
Control Objective	See proof of rules updated according to policy and procedure. There needs to be more than one accessible source to be reliable.
Risk	Missing updated rules could allow attacks go undetected. Patterns/rules must be kept updated to reflect changing security concerns.
Compliance	This step will comply with policy and procedure or it will be an exception. It must include more than one live source where rules can be downloaded.

[12] *Roesch, Martin. Track 3 – Intrusion Detection In Depth, Volume 3.3 Intrusion Detection Snort Style. SANS Institute 2001*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

<table>
<tr><td><b>Testing</b></td><td>Proof of custom script or manual procedure to get new rules from more than one source. Have administrator show and explain this. Browse to, or ftp to the sites where the rules are being updated. These sites must be available with the latest rules.
e.g. This shows the rules page on http://www.snort.org/dl/rules/ notice the “Last Modified” column. We would want to make sure we have these are up to date.

Snort.org - Microsoft Internet Explorer
Address http://www.snort.org/dl/rules/
Snort
Got Source? Our Team About Snort License
The Open Source Network Intrusion Detection System
hosted by Sourcefire
Resources
» News
Get the latest news about our favorite pig
» Documentation
Information on how to setup the pig
» Downloads
Get the pig, and all addons that make the pig easier to use
If you are using a version before 1.9.x, please upgrade
../
<table>
<tr><th>File</th><th>Checksum</th><th>Signature</th><th>README</th><th>Last Modified</th><th>Comment</th></tr>
<tr><td>snortrules-stable.tar.gz</td><td>MD5</td><td></td><td></td><td>Wed Mar 5 15:16:39 2003 GMT</td><td>Works for 1.9.x</td></tr>
<tr><td>snortrules-current.tar.gz</td><td>MD5</td><td></td><td></td><td>Wed Mar 5 15:16:38 2003 GMT</td><td>Works for HEAD branch of CVS</td></tr>
</table>
</td></tr>
<tr><td><b>Objective / Subjective</b></td><td>Objective</td></tr>
</table>

2.18 Checklist Item 19 - Snort Performance

Performance of the system is also very important. We don’t want to overload our system with redundant rules or unnecessary preprocessors. An optimized system will have the least chance of packets not getting processed.

Checklist Item 19 Snort Performance

Reference	SANS – Intrusion Detection Snort Style 3.3 pg 1-163 – 1-170

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

	http://www.snort.org/
Control Objective	Snort optimized. Verify that Snort is configured to run as fast as possible and still collect appropriate data to comply with company policy.
Risk	If Snort is overloaded with unnecessary rules and preprocessors the overhead could affect the performance of the server making it run slow, drop or not process packets.
Compliance	Snort configuration script will be viewed and verify that only necessary preprocessors and output plug-ins are being used.
Testing	Check for unnecessary preprocessors, output plug-ins. Check **/usr/local/etc/snort.conf** configuration file using **less /usr/local/etc/snort.conf** Compare what preprocessors and output plug-ins are found with what is needed to achieve company policy. Can all requirements be met with the configuration found or could it be done with less.
Objective / Subjective	Subjective

2.19 Checklist Item 20 - Snort Processing

Dropped or unprocessed packets are unacceptable in IDS. Snort keeps statistics that can be viewed to provide useful system information.

Checklist Item 20 Snort Processing

Reference	http://www.snort.org Personal experience

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Control Objective	The Snort system should not experience any unprocessed packets. We will confirm that the system is not overworked.
Risk	If traffic levels are too great for the current Snort server, packets could go unprocessed. This is very bad. For the IDS to be effective it must be able to process all packets so it can log any problems.
Compliance	Packets dropped must = 0
Testing	**! Note**: we will be getting the administrator to restart the snort daemon in this step, make sure you have proper authorization to have this done. Check snort statistics with following commands: run **ps –efl | grep snort** - find the pid of Snort **kill -HUP <pid>** - restart the process to dump statistics to syslog Confirm that packets dropped = 0 using **cat /var/log/syslog**
Objective / Subjective	Objective

2.20 Checklist Item 21 - Snort Attack Recognition

Attacks and potential harmful traffic must be reported. Even a passive portscan that could mean reconnaissance, we want to see it.

Checklist Item 21 Snort Attack Recognition

Reference	Listing of Snort rules with documentation http://www.snort.org/cgi-bin/done.cgi

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2003, Author retains full rights.

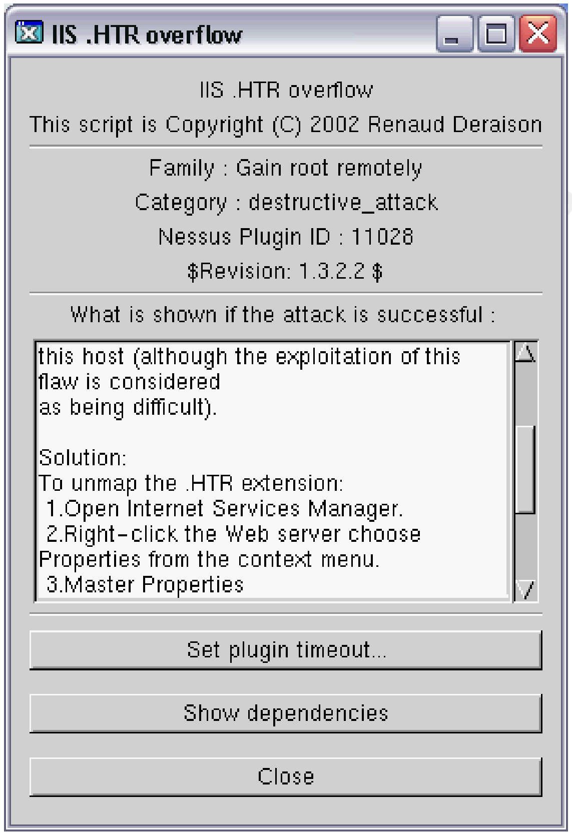
	SANS Track 3 – Intrusion Detection In Depth Using Nessus's NIDS evasion features: http://www.nessus.org/doc/nids.html
Control Objective	To determine if snort is detecting and alerting on the various exploits and attack types that it should be looking for.
Risk	It is very important to choose the rules added wisely. Since we are monitoring a DMZ that has only Windows IIS servers running, so rules relevant to these should be in our rules file. It is also a waste of Snort resources to test traffic against attacks we don't wish to look for.
Compliance	All attacks carried out, including portscan, should be found in snort logs.
Objective / Subjective	Objective
Testing	Using the tools Nessus[13], Socket80[14] and NMAP against the dummy server we can simulate many attacks that Snort should alert on. Check the Snort log to confirm that is has recognized attacks. **Sample Files 1 – 4 on the following 4 pages show the Nessus plugins, procedures and the output expected in the Snort logs.**

[13] Nessus Security Scanner –http:// www.nessus.org/intro.html

*[14] **Socket80** is a program for checking the UniCode Exploit on Microsoft IIS Web Servers.*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Simulated Attack 1 IIS .HTR overflow Nessus Plugin ID: 11028



IIS .HTR overflow

IIS .HTR overflow
This script is Copyright (C) 2002 Renaud Deraison

Family : Gain root remotely
Category : destructive_attack
Nessus Plugin ID : 11028
$Revision: 1.3.2.2 $

What is shown if the attack is successful :

```
this host (although the exploitation of this
flaw is considered
as being difficult).

Solution:
To unmap the .HTR extension:
 1.Open Internet Services Manager.
 2.Right-click the Web server choose
Properties from the context menu.
 3.Master Properties
```

With Nessus client open and logged in:

3. Go to the **Plug-in tab**, press the **Filter** button and check "**ID Number**"
4. Enter the **Nessus Plug-in ID**: 11028 (this will Enable the Plug-in)
5. Go to the **target selection** tab, add in "**dummy web server**" laptop IP address
6. Start the scan
7. On the IDS, using **cat /var/log/syslog | grep <IP address of attacking laptop>** have Intrusion Analyst show the attack in log, if it was detected.
8. Go to **Plug-ins** tab and select "**Disable all**".
9. Repeat for next plugin

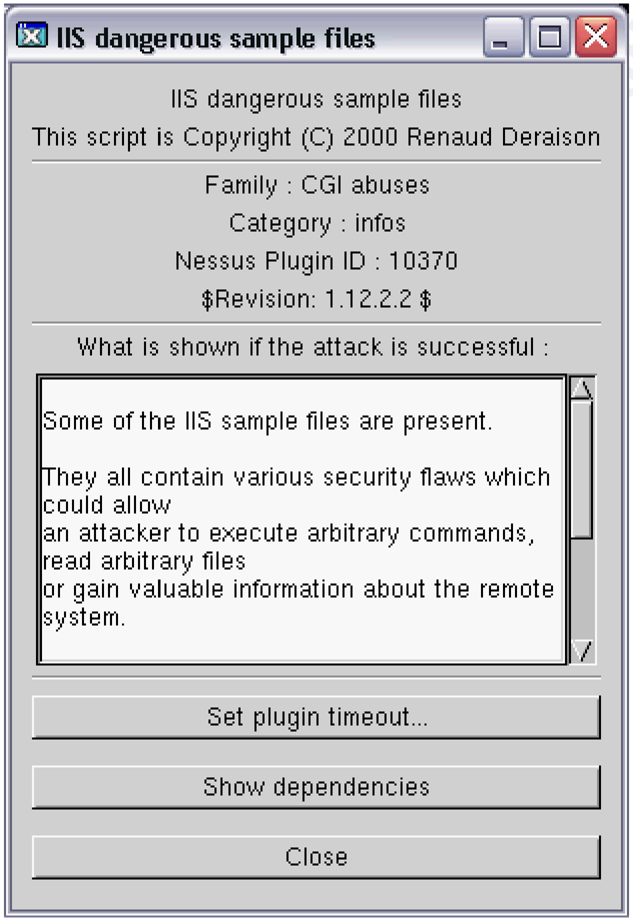
An Alert Example in Snort logs:

```
[**] WEB-IIS .htr access [**]
03/10-14:28:54.651395 10.1.1.100:58827 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:13958 IpLen:20 DgmLen:173 DF
***AP*** Seq: 0x48639169  Ack: 0x86FE4F8C  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199272429 0
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Author retains full rights.

Simulated Attack 2 IIS Dangerous Sample files Nessus Plugin ID: 10370



IIS dangerous sample files

IIS dangerous sample files

This script is Copyright (C) 2000 Renaud Deraison

Family : CGI abuses

Category : infos

Nessus Plugin ID : 10370

$Revision: 1.12.2.2 $

What is shown if the attack is successful :

```
Some of the IIS sample files are present.

They all contain various security flaws which
could allow
an attacker to execute arbitrary commands,
read arbitrary files
or gain valuable information about the remote
system.
```

With Nessus client open and logged in:

1. Go to the **Plug-in tab**, press the **Filter** button and check "**ID Number**"
2. Enter the **Nessus Plug-in ID**: 10370 (this will Enable the Plug-in)
3. Go to the **target selection** tab, add in "**dummy web server**" laptop IP address
4. Start the scan
5. On the IDS, using **cat /var/log/syslog | grep <IP address of attacking laptop>** have Intrusion Analyst show the attack in log, if it was detected.
6. Go to **Plug-ins** tab and select "**Disable all**".
7. Repeat for next plugin

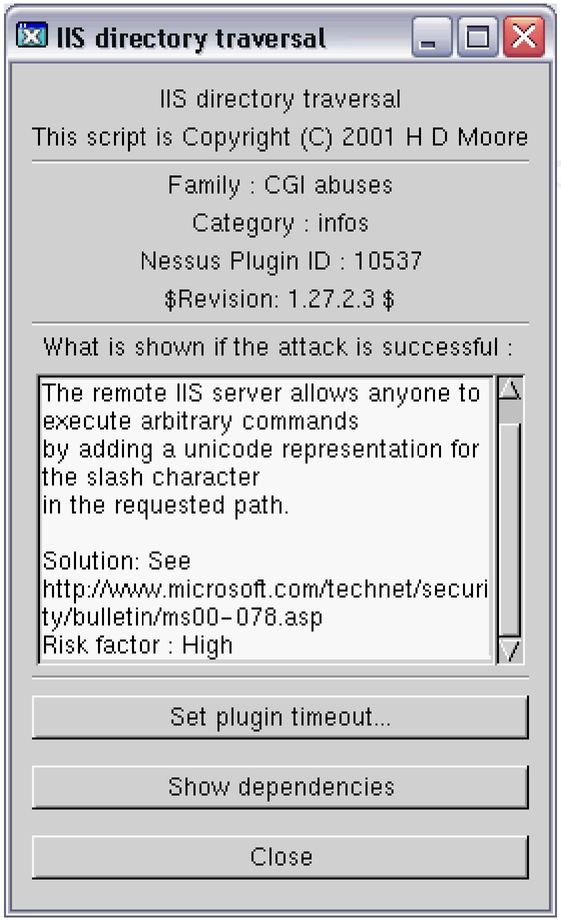
An Alert Example in Snort logs:

```
[**] WEB-IIS iissamples access [**]
03/10-14:09:33.851405 10.1.1.100:55247 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:34772 IpLen:20 DgmLen:110 DF
***AP*** Seq: 0xFECC6556  Ack: 0x770E02A5  Win: 0x16D0  TcpLen:
    32
TCP Options (3) => NOP NOP TS: 199156359 0
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Author retains full rights.

Simulated Attack 3 IIS Directory Traversal Nessus Plugin ID: 10537



IIS directory traversal
This script is Copyright (C) 2001 H D Moore

Family : CGI abuses
Category : infos
Nessus Plugin ID : 10537
$Revision: 1.27.2.3 $

What is shown if the attack is successful :

```
The remote IIS server allows anyone to
execute arbitrary commands
by adding a unicode representation for
the slash character
in the requested path.

Solution: See
http://www.microsoft.com/technet/securi
ty/bulletin/ms00-078.asp
Risk factor : High
```

With Nessus client open and logged in:

1. Go to the **Plug-in tab**, press the **Filter** button and check "**ID Number**"
2. Enter the **Nessus Plug-in ID**: 10537 (this will Enable the Plug-in)
3. Go to the **target selection** tab, add in "**dummy web server**" laptop IP address
4. Start the scan
5. On the IDS, using **cat /var/log/syslog | grep <IP address of attacking laptop>** have Intrusion Analyst show the attack in log, if it was detected.
6. Go to **Plug-ins** tab and select "**Disable all**".
7. Repeat for next plugin

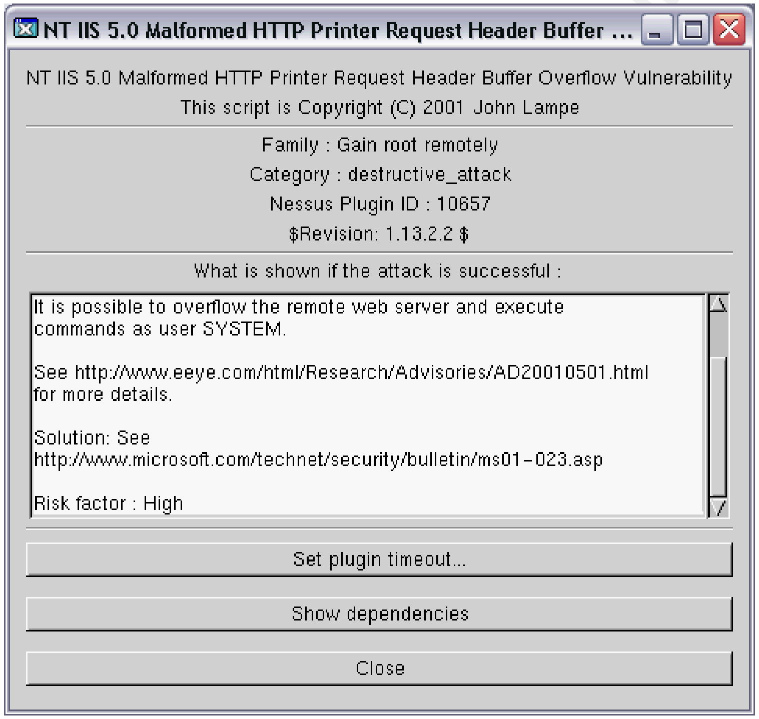
An Alert Example in Snort logs:

```
[**] spp_unidecode: Invalid Unicode String detected [**]
08/10-14:00:08.081179 10.1.1.100:54983 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:27359 IpLen:20 DgmLen:150 DF
***AP*** Seq: 0xDBB66F57  Ack: 0x6EFF5D34  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199099787 0
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Author retains full rights.

Simulated Attack 4 IIS 5.0 Malformed HTTP Printer Request Nessus Plugin ID: 10657



NT IIS 5.0 Malformed HTTP Printer Request Header Buffer ...

NT IIS 5.0 Malformed HTTP Printer Request Header Buffer Overflow Vulnerability
This script is Copyright (C) 2001 John Lampe

Family : Gain root remotely
Category : destructive_attack
Nessus Plugin ID : 10657
$Revision: 1.13.2.2 $

What is shown if the attack is successful :

```
It is possible to overflow the remote web server and execute
commands as user SYSTEM.

See http://www.eeye.com/html/Research/Advisories/AD20010501.html
for more details.

Solution: See
http://www.microsoft.com/technet/security/bulletin/ms01-023.asp

Risk factor : High
```

With Nessus client open and logged in:

1. Go to the **Plug-in tab**, press the **Filter** button and check "**ID Number**"
2. Enter the **Nessus Plug-in ID**: 10657 (this will Enable the Plug-in)
3. Go to the **target selection** tab, add in "**dummy web server**" laptop IP address
4. Start the scan
5. On the IDS, using **cat /var/log/syslog | grep <IP address of attacking laptop>** have Intrusion Analyst show the attack in log, if it was detected.
6. Go to **Plug-ins** tab and select "**Disable all**".
7. Repeat for next plugin

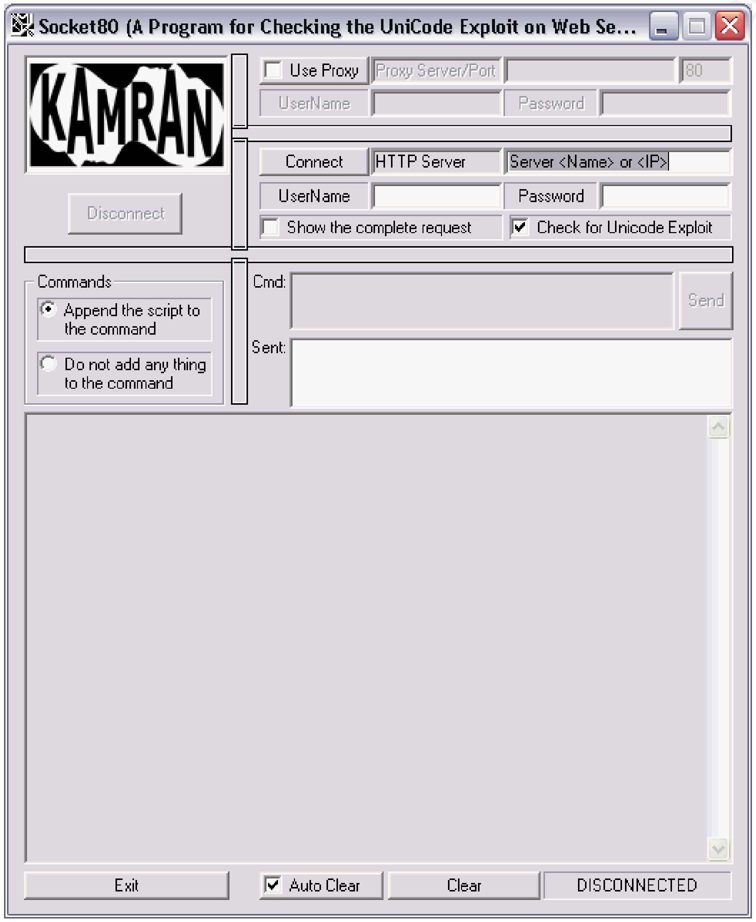
An Alert Example in Snort logs:

```
[**] WEB-IIS ISAPI .printer access [**]
08/10-14:35:52.589373 10.1.1.100:58984 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:11187 IpLen:20 DgmLen:510 DF
***AP*** Seq: 0x622E609C  Ack: 0x8CB327E3  Win: 0x16D0
    TcpLen: 32
TCP Options (3) => NOP NOP TS: 199314218 0
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Author retains full rights.

Simulated Attack 5 Socket80


Socket80 (A Program for Checking the UniCode Exploit on Web Se...
KAMRAN
□ Use Proxy
Proxy Server/Port
80
UserName
Password
Disconnect
Connect
HTTP Server
Server <Name> or <IP>
UserName
Password
□ Show the complete request
☑ Check for Unicode Exploit
Commands
● Append the script to the command
○ Do not add any thing to the command
Cmd:
Send
Sent:
Exit
☑ Auto Clear
Clear
DISCONNECTED

Using Socket80

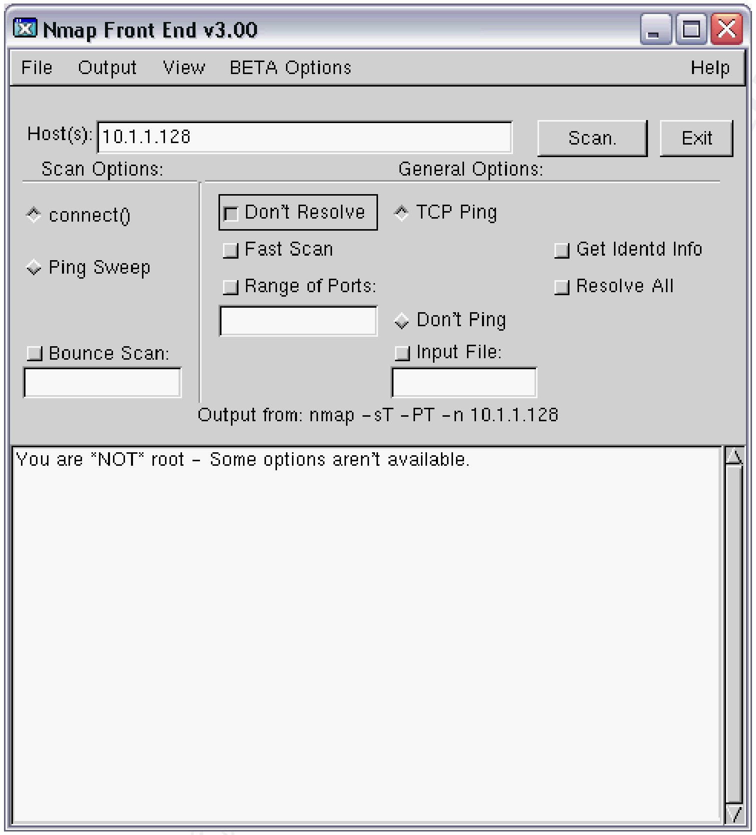
1. Enter the IP address of the HTTP Server to be attacked in the box with "Server <Name> or <IP>
2. Press the Connect button to check for Unicode Exploit

An Alert Example in Snort logs:

```
[**] WEB-IIS cmd.exe access [**]
03/10-14:04:46.283282 10.1.1.100:1026 -> 10.1.1.128:80
TCP TTL:126 TOS:0x0 ID:27291 IpLen:20 DgmLen:146 DF
***AP*** Seq: 0x8E95DD09  Ack: 0x731A5109  Win: 0xFAF0
TcpLen: 20
```

Simulated Attack 6 Nmap

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.



Host(s): 10.1.1.128

```
Output from: nmap -sT -PT -n 10.1.1.128
```

```
You are *NOT* root - Some options aren't available.
```

Nmap

1. Enter IP of dummy web server laptop in "Host(s)" box
2. Check "Don't Resolve"
3. Press scan button to portscan server

An Alert Example in Snort logs:

```
[**] [1:628:1]  <eth1> SCAN nmap TCP [**]
01/18-00:30:31.605200 10.1.1.100:42645 -> 10.1.1.128:1
TCP  TTL:47 TOS:0x0 ID:5350 IpLen:20 DgmLen:60
***A**** Seq: 0x69F6A128  Ack: 0x0  Win: 0x800  TcpLen: 40
TCP  Options (5) => WS: 10 NOP MSS: 265 TS: 1061109567 0
EOL
[Xref => http://www.whitehats.com/info/IDS28]
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

3 Assignment 3: Audit Evidence

A complete audit was done the system. The assignment only asks for detailed results from 10 of the checklist items, the following steps from the checklist are demonstrated:

Table 2 Results of Audit

Checklist Item Number	Item Audited	Results
1	IDS Policy	Pass (with comments)
2	IDS Procedure	Fail
4	IDS Physical Security	Pass
7	Time Synchronization (NTP)	Pass
9	Interfaces	Pass
11	SSH daemon	Fail
15	Snort Initialization and Configuration	Pass
18	Snort – Signature Update	Fail
20	Snort processing	Pass
21	Snort Attack Recognition	Pass

3.1 Checklist Item 1 - IDS Policy – Pass (with comments)

I obtained a copy of ACME's IDS Policy. It is actually a sub-section of their company's Systems and Network Security Policy. It was readily available to the system administrators and other IT staff. The excerpt for their IDS follows:

Audit Result 1: Intrusion Detection System Policy
This policy adequately covers all the important IDS requirements.

> "Intrusion Detection System (IDS)
> An intrusion detection system will be placed on a mirror port on the DMZ segment to monitor all traffic inbound for patterns that match known break-in attempts in order to work as the first line of monitoring defense controlled by ACME.
>
> The system will have two network cards, one residing on the DMZ segment and one on the internal segment. The card on the DMZ segment will be setup with promiscuous mode enabled and it will have no IP address assigned to it to prevent malicious individuals from trying to connect to it. All access to this server will be through SSH only.
>
> The software being use is called Snort and is available from http://www.snort.org Open source software was chosen for the ability to audit the code to ensure that there is no vulnerabilities. A second consideration is the scripting ability of the product that ACME can exploit when brand new vulnerabilities are discovered but not yet implemented in the current rule set of the IDS."
>
> This system is to be updated weekly and in case of newly discovered vulnerabilities, updated immediately to prevent a security gap from occurring between the discovery of a given vulnerability and the arrival of the patch that fixes it."

© SANS Institute 2003, Author retains full rights.

A brief interview with the CSO made me understand the expectations of the IDS. All items were covered in this policy, except one. The CSO made it clear to me that they wanted to know "everything possible harmful coming" at their web servers.

3.2 Checklist Item 2 - IDS Procedure - Fail

This part of testing proved to be quite difficult. ACME's IDS procedure document consisted of 2 separate documents located in a labyrinth of folders on a network share. These documents were "works in progress" and no value would be added to include them with this audit.

An informal interview with the Intrusion Analyst (who was also the system administrator of this system) gave me the impression that these documents were not followed and Change Control documents from a recent change could not be produced. The IDS procedures were not easy to find, and was left open to many interpretations.

3.3 Checklist Item 4 - IDS Physical Security – Pass

Outside the building

- Typical business area, lots of corporate buildings some restaurants etc…
- AMCE building has a sign with company logo, otherwise did not "stand out" in scenery
- 10 minute response from emergency services in this area
- There are security cameras viewing the parking lot

External security measures

- Sign stating premises were guarded by a security system
- Surveillance cameras upon entering lobby
- Card readers on all doors and elevators from 1st floor

Walking to the server room

(A temporary pass card is obtained, after the required security check is done)

- Camera in hallway leading to server room
- Card reader access on large steel door that my temporary card did NOT have access to, visitors must be accompanied by an escort with privileges

In the server room

- More cameras viewing most racks and all access terminals
- IDS system was in a locked rack, key was located in operations room (small room in server room, attended 24/7) and needed to be signed out with valid company ID, again my temporary pass was not sufficient to gain access to any keys
- Room didn't have any windows raised floor with no access out of the room from underneath, checked under tiles near walls / doors

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

- Drop ceiling, that I was told also wasn't connected / accessible from any adjoining rooms from a passing maintenance worker
- Console was up and running when we sat down

3.4 Checklist Item 7 - Time Synchronization - NTP – Pass

- Confirmed that NTP daemon was running

Audit Result 2: ps –ef | grep ntpd

```
root@ids:/var/log# ps -ef | grep ntpd
root      83     1  0  2002 ?        00:00:00 /usr/bin/ntpd
root@ids:/var/log#
```

- Confirmed that NTP daemon was latest version compared to http://www.eecis.udel.edu/~ntp/download.html and stratum was <6

Audit Result 3: ntpq –n –c rv

```
root@ids:~# ntpq -n -c rv
status=06f4 leap_none, sync_ntp, 15 events, event_peer/strat_chg,
version="ntpd 4.0.99k23 Sun Apr  8 15:36:29 PDT 2001 (1)",
processor="i686", system="Linux2.4.17", leap=00, stratum=4,
precision=-17, rootdelay=24.488, rootdispersion=147.568, peer=13708,
refid=192.168.44.89,
reftime=c1d5e3b9.f7229e90  Sat, Nov 23 2002 21:55:53.965, poll=10,
clock=c1d5e504.26413db7  Sat, Nov 23 2002 22:01:24.149, state=4,
offset=-2.637, frequency=-9.974, jitter=3.582, stability=0.007
root@ids:~#
```

3.5 Checklist Item 9 - Interfaces – Pass

- Checked interfaces

Audit Result 4 : ifconfig -a

```
root@ids:/# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:02:A5:E3:9E:9E
          inet addr:10.1.1.253  Bcast:10.1.255.255 Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:220587600 errors:0 dropped:0 overruns:3 frame:3
          TX packets:9333358 errors:0 dropped:0 overruns:47 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:1521234547 (1450.7 Mb)  TX bytes:302419145 (288.4
Mb)
          Interrupt:10 Base address:0x2000

eth1      Link encap:Ethernet  HWaddr 00:00:D1:4A:27:C5
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:1039058530 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:1555747363 (1483.6 Mb)  TX bytes:0 (0.0 b)
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
           Interrupt:11 Base address:0x4000

lo         Link encap:Local Loopback
           inet addr:127.0.0.1  Mask:255.0.0.0
           UP LOOPBACK RUNNING  MTU:16436  Metric:1
           RX packets:98 errors:0 dropped:0 overruns:0 frame:0
           TX packets:98 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:0
           RX bytes:19790 (19.3 Kb)  TX bytes:19790 (19.3 Kb)
```

3.6 Checklist Item 11 - SSH Daemon – Fail

- Checked for ssh daemon running

Audit Result 5: ps –ef | sshd

```
root@ids:/# ps -ef | grep sshd
root         58     1  0  2002 ?        00:00:14 /usr/sbin/sshd
root       6373    58  0 Jan07 ?        00:00:00 /usr/sbin/sshd
root       6400    58  0 Jan07 ?        00:00:00 /usr/sbin/sshd
root@ids:/#
```

- Ran the following command and compared version with current version at http://openssh.org (OpenSSH 3.5 released October 14, 2002

Audit Result 6: ssh -V

```
root@ids:/# ssh -V
OpenSSH_2.9p1, SSH protocols 1.5/2.0, OpenSSL 0x0090601f
root@ids:/#
```

3.7 Checklist Item 15 - Snort - Initialization & Configuration - Pass

- Viewed **/etc/rc.d/rc.inet2** and found:

Audit Result 7: cat /etc/rc.d/rc.inet2

```
# Start snort interfaces
echo "Starting snort..."
/usr/local/sbin/snort -c /usr/local/etc/snort_eth1.conf -d -D -i eth1 -
I -l /var/log/alert_eth1/
```

3.8 Checklist Item 18 - Snort - Signature Update – Fail

- The System Administrator / Intrusion Analyst pointed me to this script: **/usr/local/sbin/update_rules** [Appendix 1]. A brief explanation was provided. The script runs nightly and using wget[15] it downloads the newest tarball of up to

[15] *wget - a utility to retrieve files from the World Wide Web*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

date rules, this is scheduled in a cron[16] job. It then unzips and unpacks the files. There was only 1 location where new rules could be obtained in the script, which is active with current rules.

3.9 Checklist Item 20 - Snort - Processing - Pass

- Used the following command to get the PID[17]:

Audit Result 8: ps -efl | grep snort

```
root@ids:/# ps -efl | grep snort
040 S root      23119      1  0  70    0     -   3427 cdrom_ Nov18 ?
00:11:42 /usr/local/sbin/snort -c /usr/local/etc/snort.conf -d -D -i
eth1 -I -l /var
100 S root      23267     85  0  69    0     -    448 do_for 06:00 ?
00:00:00 /bin/sh -c /usr/local/sbin/snort_parser 1> /dev/null 2>
/dev/null
root@ids:/#
```

- PID is highlighted.
- Next command restarts the Snort process to dump statistics run as follows:

Audit Result 9: kill -HUP <pid>

```
root@ids:/# kill -HUP 23119
root@ids:/#
```

- Using **cat /var/log/syslog** the administrator located the snort statistics entry:

Audit Result 10: cat /var/log/syslog

```
===============================================================================
Snort analyzed 1898239 out of 1898239 packets, dropping 0(0.000%) packets

Breakdown by protocol:                Action Stats:
    TCP: 1807857    (95.239%)          ALERTS: 3052
    UDP: 24864      (1.310%)          LOGGED: 3040
   ICMP: 789        (0.042%)          PASSED: 0            ARP: 6852      (0.361%)
   IPv6: 0          (0.000%)
    IPX: 0          (0.000%)
  OTHER: 57877      (3.049%)
DISCARD: 0          (0.000%)
===============================================================================
Fragmentation Stats:
Fragmented IP Packets: 0          (0.000%)
    Fragment Trackers: 0
   Rebuilt IP Packets: 0
   Frag elements used: 0
Discarded(incomplete): 0
   Discarded(timeout): 0
```

[16] *The Unix clock daemon that executes commands at specified dates and times according to instructions in a "crontab" file.*

[17] *PID – process identification*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

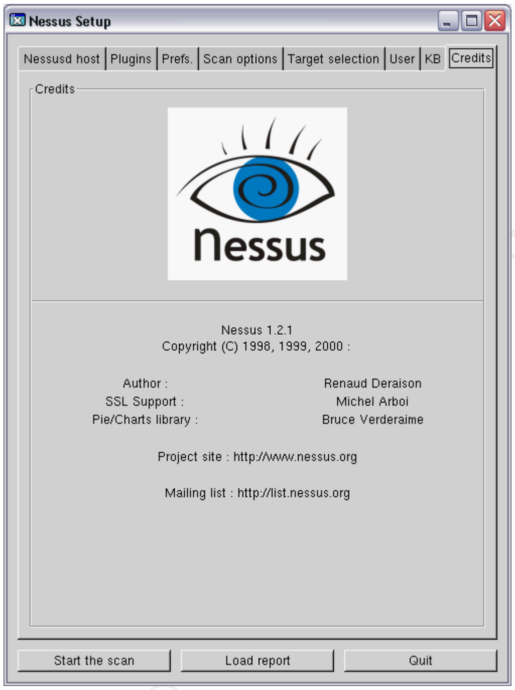
```
 Frag2 memory faults: 0
============================================================================
TCP Stream Reassembly Stats:
        TCP Packets Used: 1807850     (95.238%)
         Stream Trackers: 115606
          Stream flushes: 0
           Segments used: 0
   Stream4 Memory Faults: 0
============================================================================
```

The highlighted dropping 0(0.000%) packets is that part we were looking for.

3.10 Checklist Item 21 - Snort - Attack Recognition – Pass

Using Nessus[18] with the web attack plugins selected IDs (10370, 10526, 10537, 10657, and 11028), Socket 80 and Nmap; we scanned a dummy web server on our laptop. From the Snort logs we can see all attacks and also the portscan is detected.

Figure 2: Nessus



Nessus 1.2.1
Copyright (C) 1998, 1999, 2000 :

Author : Renaud Deraison
SSL Support : Michel Arboi
Pie/Charts library : Bruce Verderaime

Project site : http://www.nessus.org

Mailing list : http://list.nessus.org

Detected attacks in /var/log/syslog:

[18] *Nessus Introduction - http://www.nessus.org/intro.html*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

NESSUS SCANS

Results of Simulated Attack 1 IIS .HTR overflow Nessus Plugin ID: 11028

```
root@ids:/# cat /var/log/syslog | grep 10.1.1.100

[**] WEB-IIS .htr access [**]
03/10-14:28:54.651395 10.1.1.100:58827 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:13958 IpLen:20 DgmLen:173 DF
***AP*** Seq: 0x48639169  Ack: 0x86FE4F8C  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199272429 0
```

Results of Simulated Attack 2 IIS Dangerous Sample files Nessus Plugin ID: 10370

```
root@ids:/# cat /var/log/syslog | grep 10.1.1.100

[**] WEB-IIS iissamples access [**]
03/10-14:09:33.851405 10.1.1.100:55247 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:34772 IpLen:20 DgmLen:110 DF
***AP*** Seq: 0xFECC6556  Ack: 0x770E02A5  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199156359 0
```

Results of Simulated Attack 3 IIS Directory Traversal Nessus Plugin ID: 10537

```
root@ids:/# cat /var/log/syslog | grep 10.1.1.100

[**] spp_unidecode: Invalid Unicode String detected [**]
08/10-14:00:08.081179 10.1.1.100:54983 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:27359 IpLen:20 DgmLen:150 DF
***AP*** Seq: 0xDBB66F57  Ack: 0x6EFF5D34  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199099787 0
```

Results of Simulated Attack 4 IIS 5.0 Malformed HTTP Printer Request Nessus Plugin ID: 10657

```
root@ids:/# cat /var/log/syslog | grep 10.1.1.100

[**] WEB-IIS ISAPI .printer access [**]
08/10-14:35:52.589373 10.1.1.100:58984 -> 10.1.1.128:80
TCP TTL:62 TOS:0x0 ID:11187 IpLen:20 DgmLen:510 DF
***AP*** Seq: 0x622E609C  Ack: 0x8CB327E3  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 199314218 0
```

SOCKET80 ATTEMPT

Results of Simulated Attack 5 Socket80

```
root@ids:/# cat /var/log/syslog | grep 10.1.1.100

[**] [1:1002:3]  <eth1> WEB-IIS cmd.exe access [**]
01/17-12:58:56.170052 10.1.1.46:38081 -> 10.1.1.110:80
TCP TTL:62 TOS:0x0 ID:46636 IpLen:20 DgmLen:102 DF
***AP*** Seq: 0x649F1191  Ack: 0x2C3EAE9A  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 612345786 0
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

NMAP SCAN

Results of Simulated Attack 6 Nmap

```
[**] [1:628:1]  <eth1> SCAN nmap TCP [**]
01/18-00:30:31.605200 10.1.1.46:42645 -> 10.1.1.110:1
TCP TTL:47 TOS:0x0 ID:5350 IpLen:20 DgmLen:60
***A**** Seq: 0x69F6A128  Ack: 0x0  Win: 0x800  TcpLen: 40
TCP Options (5) => WS: 10 NOP MSS: 265 TS: 1061109567 0 EOL
[Xref => http://www.whitehats.com/info/IDS28]
```

3.11 Measure Residual Risk

The findings from the audit are that overall the ACME system could do better with their IDS system. The IDS system itself did very well on meeting control objectives, Snort is a good choice for flexibility and powerful scanning to meet the company's needs and fit into their structure of layered security. With all of our controls in place eliminating all risk is not feasible so we will try to minimize it wherever we can.

- Items 1 and 2 (Policy and Procedure) on the checklist are very important to keep the system running properly and smoothly. ACME's IDS Policy didn't measure up too bad; it covered the main points need for a functional Policy. The only part that stood out to need revision was the following; their policy states "monitor all traffic inbound for patterns that match known break-in attempts". All attacks are not just break-in attempts, there are a variety of others that would be useful to know, like DOS[19], reconnaissance[20] and according the CSO, he wanted to know. I would recommend that they revisit this and broaden this view in their Policy. I expect this could be updated in 1 day by the time that revisions could be made and approved before actually being added to the final document.
- The lack of proper Procedure documentation is in critical need of change. Proper Procedures can save the company on wasted time researching system installs/upgrades and the possible consequences of doing one of these incorrectly. Extensive research would have to be done in all areas of this system. The partially completed documents I viewed could be used to achieve a useful and accurate Procedure document. I would estimate from experience that this would take 4 days to complete.
- Another risk to the system that cannot be eliminated by this control but limited was an outdated version of SSH. Newest releases of software can introduce new vulnerabilities, but I think it would be more beneficial to have this upgraded. The costs estimate with this is again just hours. The software being used is open source so it is free, but a system administrator would have to spend 2 hours fixing and testing it. To minimize risk regular upgrades of software on this system must be done to prevent vulnerable software running on this machine.
- The automatic rule updated for the Snort system has worked well for the system. The problems I could see with this are that there was only one source for the rules to

[19] *Denial of Service (DoS) Attack Resources* http://www.denialinfo.com/
[20] *Network Reconnaissance Techniques* http://www.insecure.org/nmap/OSDEM_Presentation/

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

be updated from. If for some reason the site became unavailable rules would not get updated that night. This could cause an important new attack signature being missed and creating a breakdown in this layer of our security. I would suggest finding an alternate site for the script to use as a backup. This is estimated at 4 hours to find and test a reliable site and update the script.

3.12 Is the System Auditable

Most of the audit was carried out to satisfy our control objectives. However there were a few checklist items that couldn't be carried out completely. Due to the lack of IDS procedure documentation and change control evidence we could not verify either of the controls pertaining to these. Our control objective to confirm that the system OS matches up to a secure installation / configuration according ACME's policy was only verified by a signed form from the administrator that built the server. This does not ensure that he actually hardened the server but just filled out the paperwork. Since the system was production and a scheduled maintenance window was out of our time line to conduct the audit we were not able to test the failover capabilities of the system. Overall I would say that the audit of this system was acceptable.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

4 Assignment 4: Audit Report or Risk Assessment

4.1 Executive Summary

This audit was conducted on the ACME Intrusion Detection System running Snort. An audit checklist was followed to prove if control objectives, measuring many aspects of the system, were met. Over a 4 day period information was gathered from employees and the IDS system itself. Detailed examination of policy, procedure and change control showed some deficiencies but physical security, system security and Snorts results proved the systems strength. Although the need for clear and accurate documentation and methodologies is essential, an outdated version of the SSH software, used for remote management, was the biggest security concern. With a few exceptions control objectives were satisfied. These results will give ACME the facts needed to ensure that this system is running to the best of its ability.

4.2 Audit Report

The audit of the ACME IDS went well. In completing the checklist, we were able to achieve our objectives. Based on Objective and Subjective Controls a complete audit was fulfilled. In accordance to our findings we will be able to provide possible risks to the systems and recommendations to alleviate these. An estimation of the costs involved will also be included based on time that it should take to complete. This can be put into monetary costs by applying the company's hour rate.

The physical IDS performed well during our audit, with only one exception. I will go over our findings in detail explaining the results of our audit, the risks involved with the findings, if any, and recommendations to help solve the problem, the costs involved with each and compensating controls.

Policy – Pass (with comments):

Results in Section 3.1

The IDS Policy in place could use some revisions. After talking with the CSO I found that ACME wants to know "everything possible harmful coming at their web servers". The line in the IDS Policy "monitor all traffic inbound for patterns that match known break-in attempts" is too specific for what is actually expected from the system.

Knowledge of passive scans might be of some importance because this is often someone doing reconnaissance on network and servers. This could give the Intrusion Analyst a "heads up" to an oncoming attack.

I would recommend that the IDS Policy be revised to suite the needs of ACME by being less specific to the types of attacks you are looking for.

Revisions should only take about 2 days to complete changes and get signed off.

This should not be out of the reach of ACME to achieve quickly.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Procedure – (Fail)

Results in Section 3.2

IDS Procedure was difficult to do a thorough audit on due to the current layout. A collection of 2 unfinished documents, that were not easily located, showed me that adequate Procedure documentation did not exist.

This documentation is very important to the initial installation and upkeep of the system. The point form, incomplete documents were left open to interpretation for a person installing or upgrading hardware/software on this system. This could result in a false sense of security if the system is not doing what is expected of it and could result in long down time in the event of an incident.

I would highly recommend that this document be completed in an easy to read format and is put in an accessible location to anyone who would need it.

An estimation of 5 days would be acceptable to create a high-quality document.

This should not be out of the reach of ACME, but if necessary a Technical Writer could be contracted to properly complete this.

Physical Security – (Pass)

Results in Section 3.3

The location of the building and amount of security and precautions in place, gave me confidence in the ability of this system to be truly physically protected.

The only risks I can see are cases of force that would be highly unlikely due to the nature of the systems i.e. not a military installation.

Although this control did pass it is important to keep this high grade of security and never let your guard down. It would be very costly to completely eliminate this risk but to minimize it I would suggest to keep employees aware of security threats like Social Engineering and ensure Policy and Procedure is followed at all times. This could be very cost effective only requiring a periodic email or meeting to update staff or make them aware of new threats.

Time Synchronization – NTP – (Pass)

Results in Section 3.4

The configuration of the NTP is very important to this system and was running a configured properly.

The ability of all systems to keep consistent synchronized time will keep all log files in sync and in the event of an incident, they would be very easy to correlate and keep events in order.

As long as NTP is kept up to date and periodically checked for synchronization I don't foresee any issues with this.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

It should be in the administrator's weekly tasks to confirm this so I don't see any further costs involved.

Secondary time servers in the event of a failure and local radio sources in addition to Internet based systems could minimize risks involved here.

Interfaces – (Pass)

Results in Section 3.5

Having the servers interfaces running and configured properly is imperative to the operation of the IDS. Both internal and sniffing interfaces were up and running as expected.

SSH - initialization & configuration – (Fail)

Results in Section 3.6

SSH allows us to connect securely to the server for administration purposes. The audit showed that the OpenSSH version running on the server was out of date and susceptible to compromise as it contains at least one major security vulnerability.

This could lead to total system takeover rendering the IDS and alerts useless.

Even though the interface that allows SSH connections is internal I would recommend that this be immediately upgraded to the latest stable version of OpenSSH. To avoid this happening again in the future a method of keeping this software up to date should be added into the system's Procedure document. To upgrade this I would allocate ½ day for installation, configuration and testing. The costs of adding it to the Procedure document are included with its costs.

To minimize these risks a script could be put into place that checks for new releases and system administrators should read security lists to keep up to date on current vulnerabilities.

Snort – initialization & configuration – (Pass)

Results in Section 3.7

Snort initialization and configuration gave us the expected results. The system is configured to comply with Policy. The choice to use Snort is a good one. Snort is very powerful and configurable.

If code and alerts are kept up to date this system it should be an integral part of ACME defense.

The cost of the software is nothing so only upkeep is needed. This would be included in the costs of a system administrator and Intrusion Analyst to cover this system.

One recommendation I would put forward to minimize risks to this system is the addition of some of Snorts compliment software to include a database (MySQL[21]), mailed alerts and a web front-end graphical interface (Apache[22] and ACID[23]) for the Intrusion Analyst.

[21] *MySQL OpenSource Database - http://www.mysql.com/*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

All of this software can be obtained free but would have to be chosen carefully to meet all needs. The costs involved with this would be another server and time to install, configure and test the system. I would estimate 4 days to accomplish this. The benefits of this are that the system would be easier for the Analyst to manage and the possibility of attacks being missed would decrease.

Snort - signature update – (Fail)

Results in Section 3.8

The method used to update the snort rule database was creatively thought out. The system receives its new rules nightly and automatically and applies them. The only thing wrong with this was the source for new rules, or the lack of. To pass this control 2 sources were needed. This would provide redundancy in case of an outage on one of these sites.

If the current site were unreachable, it would be possible for the system to miss the nightly update and miss a new rule. This could result in a false sense of security and the IDS missing an attack or intrusion attempt.

For a reliable second source to be found and the script updated, configured and tested and I would allocate 1 day.

Snort – processing – (Pass)

Results in Section 3.9

The server configuration with the current hardware is processing all of the packets Snort gets. The goal of 0% packets dropped is being met. As demand increases to the segments being monitored, testing should be done to ensure this system remains sufficient. If this does become a concern, bigger, faster hardware would solve it quickly. The costs would depend on the upgrades involved and cannot be estimated at this time.

If upgrades were needed and costs were too high more, cheaper systems could be used to cover what was needed. This would reduce chances of Snort dropping packets in a larger environment.

Snort – attack recognition – (Pass)

Results in Section 3.10

All attacks and scans that we tried were recognized and logged by Snort. New exploits come out everyday, so this is one to keep on top of. If a new exploit were to be launched against a system on the network that Snort was "watching" we better be sure that we get a positive identification. As new attacks are made public the Intrusion Analyst must be sure that Snort will alert on them.

This should be part of the procedure and covered already. Changes to the system, configuration and testing should fit into the system administrator and Intrusion Analyst's

[22] *Apache HTTP Server - http://www.apache.org/*
[23] *Analysis Console for Intrusion Databases (ACID) - http://www.andrew.cmu.edu/~rdanyliw/snort/snortacid.html*

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

duties. By being informed on new attacks and getting updated rules into the IDS you can greatly increase your chances of alerting on all potential problems.

4.3 Summary

As a whole this system fits into the Defense in Depth methodology well. If the above exceptions are brought up to speed, this IDS should meet and exceed any expectations placed upon it. This is not the end though. The audit process should be ongoing; we have just established a baseline to go on next time. Hopefully we can catch problems before they happen and bring potential risks down. No system can be 100% secure but if you know you systems strengths and weaknesses it will put you on a level playing field with the bad guys.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

5 Appendices

5.1 Appendix 1 – Rule updater

```
#!/usr/bin/perl

## Change working directory
chdir "/usr/local/etc" or die "Cannot change working directory: $!\n";

## Get current rules file
system "wget http://www.snort.org/dl/signatures/snortrules.tar.gz";

## Unzip and cat rules
system "tar xzf snortrules.tar.gz";
chdir "/usr/local/etc/rules" or die "Cannot change working directory: $!";
system "cat backdoor.rules ddos.rules dos.rules exploit.rules ftp.rules icmp.rules scan.rules
shellcode.rules telnet.rules web-cgi.rules web-iis.rules web-misc.rules > ../new.rules";

## Cleanup downloaded files
chdir "/usr/local/etc" or die "Cannot change working directory: $!";
system "rm -Rf /usr/local/etc/rules";
system "rm snortrules.tar.gz";

## Open new rules file for comparison
open(NEW, "< new.rules") or die "Could not read new rules file! $!\n";

## Make sure rules file exists
if (not -f "/usr/local/etc/snort.rules") {
        system "touch snort.rules";
}

## Compare new rules to original rules and save new rules in array
while ($line_new = <NEW>) {

        ## Skip line if empty
        next if ($line_new =~ /^$/);

        ## Skip line if commented
        next if ($line_new =~ /^#/);

        ## Get rule string
        $line_new =~ /.*/;
        $new = $&;
        $new =~ s/\s*classtype:.+?;//;

        ## Open original rules file for comparison
        open(ORIG, "< snort.rules") or die "Could not read original rules file! $!\n";

        ## Step through rules
        while ($line_orig = <ORIG>) {

                ## Skip line if empty
                next if ($line_orig =~ /^$/);
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
        ## Crop line if commented
        if ($line_orig =~ /^#/) {
            for ($line_orig) {
                s/^#//;
                s/^\s+//;
            }
        }

        ## Get rule string
        $line_orig =~ /.*/;
        $orig = $&;

        ## Look for match
        if ($new eq $orig) {
            $match = 1;
        }
    }

    ## Close file
    close (ORIG);

    ## Store rule if no match
    unless ($match == 1) {
        $rules[$n] = $new;
        $n++
    }

    ## Undefine match variable
    undef $match;

}

## Close file
close(NEW);

## Remove temporary rules file
system "rm new.rules";

## Open original file for append
open(ORIG, ">> snort.rules") or die "Could not append to original rules file! $!\n";

## Append the new rules
foreach $rule (@rules) {
    print ORIG "$rule\n";
}

## Close file
close(ORIG);

## Restart Snort
system "/usr/local/sbin/snort -c /usr/local/etc/snort_eth1.conf -d -D -i eth1 -I -l /var/log/alert ";
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

6 References

"Snort website" URL: http://www.snort.org

"Michael Sobirey's Intrusion Detection Systems page" URL: http://www-rnks.informatik.tu-cottbus.de/~sobirey/ids.html (Dec 2002)

"The SANS Security Policy Project " URL: http://www.sans.org/resources/policies/

"IT Security Policies, Network Security Policies & Effective Delivery" URL: http://www.network-and-it-security-policies.com/ (Nov 2002)

"ISACA Standards for Information Systems Control Professionals" URL: http://www.isaca.org/standard/iscontrl.htm (Oct 2002)

Bois, Justin. "Protect Yourself" URL: http://www.sans.org/rr/physical/protect.php (April 4, 2002)

"CIS Level-1 Benchmark and Scoring Tool For Linux" URL: http://www.cisecurity.com/bench_linux.html

"High-Availability Linux Project " URL: http://linux-ha.org/

"Time Synchronization Server" URL: http://www.ntp.org

"Snort Definition on searchsecurity.com" URL: http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci789029,00.html

"Linux Security" URL: http://www.linuxsecurity.com

OpenSSH URL: http://openssh.org/

Nmap introduction URL: http://www.insecure.org/nmap/intro

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Corcoran, Tim. "An introduction to NMAP" URL:
http://www.sans.org/rr/audit/nmap2.php

Wassom, Darrin "Auditing a Distributed Intrusion Detection System: An Auditors Perspective" URL: http://www.giac.org/practical/Darrin_Wassom_GSNA.doc

"Nessus Security Scanner" URL: http://www.nessus.org

"Importance of Backups" URL: http://www.pcguide.com/care/bu/exer-c.html

"Astalavista security archive" URL: http://www.astalavista.com/

Rain Forest Puppy, "A look at whisker's anti-IDS tactics" URL:
http://packetstormsecurity.nl/papers/IDS/whiskerids.html

"The Slackware Linux Project" URL: http://www.slackware.org

"The Center for Internet Security" URL: http://www.cisecurity.org

"Linux User's Manual" URL: http://www.nevis.columbia.edu/cgi-bin/man.sh

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.