Global Information Assurance Certification Paper

Name: Leon Morton
Version: 1.4b GSEC Practical Assignment
Title: HP IDS/9000 For Your HP-UX Enterprise

Abstract

Companies continue to suffer financial loss due to computer crimes, such as unauthorized use of computer systems and attacks at both the network perimeter and internally. The need for a reliable and efficient Intrusion Detection System (IDS) is a must. There are many IDS solutions for HP-UX systems. It is important to know which solution is best for you. The purpose of this paper is to describe the features and benefits of HP's IDS/9000 and how it can be used in your HP-UX enterprise to notify you of potential harm to your computing environment.

What is intrusion detection and why do you need an Intrusions Detection System (IDS)?

To help answer the question of why you need intrusion detection we will look at the threats facing business today. Some of the common threats revolve around:

- Financial Assets
- Intellectual Property
- Computing Resources
- Privacy

Financial assets range from intruders getting paid to compromise systems to electronically stealing credit card numbers. Computers are an integral part of our everyday business environment. More and more people are using the Internet for shopping and banking.

It has been reported, "the theft of proprietary information has been one of the costliest forms of computer crime." [1] Much of business today thrives on information and using that information to give you a competitive edge. Intellectual Property theft is at the center many sectors, from music sharing on the web to the Supreme Court's consideration of Congress's 1998 extension of the terms of U.S. copyright.

History has shown us that no computer enterprise is 100% secure no matter how much time and money you invest. Security incidents will continue to occur. The 2003 CSI/FBI report shows us that there are still an "overall number of significant incidents."[1] Once you have secured your network perimeter and hosts, the next step is to setup a detection system that can notify you if your network and/or hosts have been compromised. Without a detection system the enterprise will have no way of knowing if security incidents have occurred.

These systems can monitor for configuration changes and look for suspicious user activity. The art of intrusion detection is in the ability to define what is

considered normal activity from malicious activity or attack signature. IDS provides the ability to take an organization's security polices and create software that will determine if it is not being followed. [2]

Types of IDS

Today IDS typically falls into one of two types:

- Network IDS (NIDS)
- Host IDS (HIDS)

Network IDS works at the network packet level analyzing network traffic as it travels across the network. NIDS is typically implemented at the perimeter of your network. Host based IDS works by monitoring host system activity such as logon and logoff, as well as file access activity.[3] To achieve as much protection as possible your enterprise should be deploying NIDS at your network perimeter complemented with HIDS to protect individual systems. The focus of this paper will be on host based IDS for HP-UX.

Why HP-UX IDS/9000?

Before HP-UX IDS/9000 the primary host-based intrusion detection systems utilized one if not all the following tools:

- TCP Wrappers
- Port Sentry
- Manual log file monitoring
- Swatch
- Tripwire

TCP Wrappers provide access control to network services such as telnet and ftp. It can allow or deny services to clients. TCP Wrappers provide detailed logging of which clients connected with what service at any given time.

Port Sentry can detect if your host system is being scanned. It can log scan activity and respond by denying services to the offending client and even configure the host so that it cannot communicate data back to the potential intruder performing the scan.

Many of the standard log files such as syslog and sulog can be used to monitor system activity. Standard log files can provide login/logout information and much more. Manually doing this or even scripting this process is possible but quite tedious.

Swatch can be used to assist with log file monitoring. It can be configured to scan log files for specific patterns that could indicate an intrusion. Swatch can be

configured to respond to specific patterns it finds. For example it can execute commands and provide remote notification.

Tripwire is used to monitor file systems changes. It maintains a snapshot of the file system in one state and compares the state the next time it is run. Tripwire can be configured to provide notification upon detecting file changes. It can also be configured to reload an earlier file if it detects that file has changed.

The combination of the above listed tools can provide network traffic control, generate logging to record activity, respond to potential threats, and manage file changes.[4]

IDS/9000 can provide much of the important functionality listed by the other tools. With respect to network IDS, it can log network logon and logoff. However it does not provide templates to detect network scans. If you are placing your HP-UX system in a DMZ, scan detection functionality is essential. However if your HP-UX systems are in your private network this functionality is less important. Attacks to your system that occur internally are often done by personnel that have login information and do not need to scan a system. In my assessment more NIDS functionality in future releases would be an enhancement but not a necessity.

HP IDS/9000 is designed to concentrate on alarming your HP-UX 11.x environment. IDS/9000 provides:

near real-time monitoring and detection	Reports intrusions as they occur so that immediate action can be taken to prevent malicious acts, loss of data, loss of financial assets, loss of intellectual property, loss of computing resources, and loss of private and public trust.
HP-UX 11i host operating environment detection	The builders of HP-UX are in the best position to monitor and protect it. As the control for your servers and network, this is the mission critical system that needs bulletproof protection.
enterprise detection and central monitoring	Many HP-UX servers can be protected, but one management console can monitor and alert for the entire system. An OpenView smart plug-in for IDS/9000 makes it possible to also monitor the system through OpenView. Administration can be reduced while supporting an HP

enterprise detection and central monitoring (continued)	virtual enterprise and Internet security strategy.
browser point and click	Easy administration and use is by point-and-click through the browser-based GUI interface. Alerts are presented through a browser for easy prioritization.
automated attack response	Attack alert HP-UX commands can be programmed to alert devices such as e-mail and pagers. They can also start policy programs, other HP-UX applications, and devices. Attack alert contributes to HP's "five nines" high availability strategy.
detection template with event correlation	Detection templates are the new technology that provides the building blocks for attack recognition coupled with event correlation. A smaller number of patterns detect a greater number of attacks. This reduces attack signature maintenance and ensures protection against attacks not yet invented.

Hewlett-Packard. "HP-UX Intrusion Detection System".

The major types of exploits detected by IDS/9000 are:

- Unauthorized access
- Privilege violations
- Trojan horse
- "Root" exploits
- Race condition
- Buffer overflow
- Password guessing
- Failed logins
- Failed SU attempts
- User A modifying User B's file
- Modification of critical system files and directories
- Creation of world writable files
- Creating setuid files
- File additions and deletions

Hewlett-Packard. "HP Intrusion Detection System/9000 Administrator's Guide".

IDS/9000 provides host-level security for your HP-UX systems. HP provides IDS/9000 at zero cost. IDS/9000 is a good IDS solution for HP-UX because of its integration with the kernel. The designers of IDS/9000 believe that the kernel has the only reliable view of the system state.[5] Other advantages of kernel audit are:

- Reduced side effects compared to library audit
- Data more reliable
- Better file system mapping
- More reliable capture of system before and after states

IDS/9000 monitors host for possible signs of intrusion. It continually monitors system log files and kernel audit data for suspicious activity. When suspicious activity is detected an alert is sent to the administrative interface. For an enterprise environment, alerts can be securely sent to a central IDS/9000 monitoring system or HP OpenView. In addition, responses to an alert can be configured to carry out commands such as disabling an account or paging administrative staff. The responses are designed to prevent further damage to the server.

Understanding IDS/9000

At the heart of the IDS/9000 system are the detection templates that are used to determine if an attack is occurring. Traditional IDS use attack "signatures" or a specific pattern or behavior of some sort to determine if something is happening. If the pattern or behavior is observed by the IDS an alert will be generated. Some examples of attack signatures include:

- Connection attempt from a reserved IP address
- Email containing a particular virus
- Piping commands to terminals
- Copying a shell

Some signatures may tell you which specific attack is occurring or what vulnerability the attacker is trying to exploit. Other signatures may just indicate that unusual behavior is occurring, without specifying a particular attack.[6]

The challenge with IDS that relies on signatures is that signatures can exhibit very specific behavior. With current estimates of signature count over one thousand and growing, maintenance costs of keeping up with current signatures are high. It also means that the systems are vulnerable to new attacks that may do damage before the signature file is updated.

Instead of specific attack signatures, HP adopted a new approach, which uses "detection templates" that focus on areas vulnerable to attack. This approach allows IDS/9000 to recognize most current attacks and positions the product to

respond to future attacks yet to be invented. Following is a list of current detection templates:

- Buffer overflow attacks
- Changes to log files
- Creation of setuid files
- Creation of world-writable files
- Modification of another user's files
- Modification of files/directories
- Monitor logins/logouts
- Monitor start of interactive sessions
- Race condition attacks
- Repeated failed logins
- Repeated failed su commands

Detection Templates in detail

This section will provide details about the detection templates and alerts. A description of the template is provided with the alert generated in the Network Node manager screen, alert format, and an example.

Buffer Overflow attacks- A common way to increase privilege on a system is to gain normal user access then exploit a buffer overflow to gain higher access. Buffer overflow occurs when a system buffer receives more data than it is able to properly process. Normal program execution is overwritten with the extra data. This is because many of the standard library functions of the program do not provide boundary checks of the buffer. These functions "operate on null-terminated strings, and do not check for overflow of the receiving string." [7] Attackers take advantage of buffer overflow by providing extra data in the form of code instructions. The code is then executed by the application. In many cases the code instructed an application program to supply a shell back to the attacker. This template watches how setuid programs are running and will alert on the following actions:

• Setuid programs executing programs other than themselves
• A program "unexpectedly" gaining user ID 0 privileges

The alerts generated: "Potential Buffer Overflow" or "Unexpected Change in Privilege"

"Potential Buffer Overflow" detail alert format:
Potential buffer overflow detected with UID:UID(GID:GID) EUID:EUID(EGID:EGID) executing BINARY1 with arguments ARGLIST1 now executing: BINARY2 with arguments ARGLIST2 as PID:PID

"Potential Buffer Overflow" detail alert example:

Potential buffer overflow detected with UID:114(GID:20) EUID:0(EGID:20) executing /tmp/sh(1,75,"40000006") with arguments["/tmp/sh"] now executing: /usr/bin/ls(1,249,"40000007") with arguments ["ls"] as PID:4602

"Unexpected Change in Privilege" detail alert format:
Unexpected change in privilege detected with UID:UID(GID:GID) EUID:EUID(EGID:EGID) executing BINARY1 with arguments ARGLIST1 and system call SYSCALL

"Unexpected Change in Privilege" detail alert example:
Unexpected change in privilege levels with UID:100(GID:20) EUID:0(EGID:20) executing /usr/bin/ksh(1,42246,"40000003") with arguments["/usr/bin/ksh", "-c", "foobar"] and system call kern_setuid as PID:19854

Changes to log files- Log files monitor system activity and are meant to be appended to and not overwritten. Many tools are designed to modify log file content and time stamps. Attackers will often times try to hide their tracks by modifying log files such as syslog, sulog, btmp, utmp, and wtmp. These log files contain critical system activity and user session information. This template monitors log files for changes other than appending to them. The alert generated: "Append-only file being modified"

"Append-only file being modified" detail alert format:
User UID ACTION FILENAME executing PATH1(FILEINFO1) with arguments ARGLIST1 as PID:PID

"Append-only file being modified" detail alert example:
User 0 created the file (and overwrote any existing file) named "/var/adm/sulog" executing /usr/bin/vi(1,14665,"40000005") with arguments ["vi", "/var/adm/sulog"] as PID:2232

Creation of setuid files- Typical back doors left on systems that have been breached are Set User ID (SUID) files. SUID files execute commands with the permissions of the owner of the file and not the permissions of the user executing it. A good example of this is user password modification. A regular user cannot directly modify the /etc/passwd file directly but can run the passwd command and modify the file. This works because the SUID of the passwd command is root. So effectively the user is root when they run the passwd command to modify /etc/passwd. This template monitors for the creation of SUID files and modification of the owner for the file.
The alert generated:
"Creation of setuid files"

“Creation of setuid files” detail alert format:
User UID enabled the setuid bit on file PATH1 executing PATH2(FILEINFO2) with arguments ARGLIST2 as PID:PID

“Creation of setuid files” detail alert example:
User 0 enabled the setuid bit on file "/etc/xxx" executing /usr/bin/chmod(1,2093,"40000005") with arguments ["chmod", "u+xs", "/etc/xxx"] as PID:2216

Creation of world-writable files- Privileged user and system files that are world writable can be exploited and used as backdoors into a system. Attackers are often time in a rush to get into and out of a system. They will often times run “chmod 4777” to manipulate file privilege and system access. For example if a root user is logged into a system and steps away from the console, an attacker could copy over a root shell and setting SUID and full access permissions. If a root user’s console or terminal is world-writable, an attacker could send commands to the root user terminal that would execute. This template monitors for creation of world writable files and change of owner on a file that is world writable. The alert generated:
“World-writable file created”

“World-writable file created” detail alert format:
User UID ACTION FILENAME DESCRIPTION executing PATH1(FILEINFO1) with arguments ARGLIST1 as PID:PID

“World-writable file created” detail alert example:
User 0 created "/etc/xxx" with world-writable permissions executing /usr/bin/touch(1,27,"40000005") with arguments ["touch", "/etc/xxx"] as PID:2213

Modification of another user’s files- If you have files that users other than the owner can modify, this template provides the data as to what user modified the file and when they did it. If an attacker has gained access to your system as a regular user they will attempt to find ways to get privileged system access. An example could be modifying root’s $HOME/.profile. If this file gives user write privileges, an attacker can modify this file to later obtain root access to the system. The alert generated:
“Non-owned file being modified”

“Non-owned file being modified” detail alert format:
User UID ACTION FILENAME owned by UID:UID2 executing PATH1(FILEINFO1) with arguments ARGLIST1 as PID:PID

“Non-owned file being modified” detail alert example:
User 0 changed ownership of "/dev/zero" owned by UID:2

executing /usr/bin/chown(1,1628,"40000008") with arguments
["chown", "root", "zero"] as PID:13620

Modification of files/directories- This template can notify on changes made to a file or directory. Changes include addition or deletion, file permissions, and writing to a file or directory. If your system has been compromised an attacker will often times try to hide their tracks and leave backdoors into the system. Attacker activities to a compromised system include modifying log files, adding privileged user accounts, and modifying critical system files. The alert generated: "Filesystem change detected"

"Filesystem change detected" detail alert format:
User UID ACTION FILENAME executing PATH1(FILEINFO1) with
arguments ARGLIST1 as PID:PID

"Filesystem change detected" detail alert example:
User 0 created the file (and overwrote any existing file)
named "/etc/passwd" executing
/usr/bin/vi(1,14665,"40000005") with arguments ["vi","/etc/passwd"] as PID:2220

Monitor logins/logouts- This template can alert you to all login and logout sessions for users you specify. Remotely capturing login in sessions can greatly assist with user management. It is generally good to specify when all system users are allowed access to a system. If you notice user activity during unusual time periods, this could be an indication of an attacker. The alert generated: "Login/Logout: "USERNAME""

"Login/Logout: "USERNAME"" detail alert format:
User "USERNAME" ACTION on DEVICE (Remote:NETADDR HOSTNAME)

"Login/Logout: "USERNAME"" detail alert example:
User "root" logged in on pts/3 (Remote:127.0.0.1
machine.hp.com)

Monitor start of interactive sessions- This template is very similar in behavior to the previous template. It will additionally log switch user (su) activity. Capturing "su" activity of users not allowed to become other users can be a key indicator of unauthorized use of a system. Often time this unauthorized use is by an unauthorized user. The alerts generated:

- "Successful su detected"
- "User login detected"

"Successful su detected" detail alert format:
User "FROMUSER" switched to user "TOUSER" on DEVICE

"Successful su detected" detail alert example:

User "root" switched to user "ids" on 2

“User login detected” detail alert format:
User "USERNAME" logged in on DEVICE (Remote: NETADDR HOSTNAME)

“User login detected” detail alert example:
User "root" logged in on pts/3 (Remote:127.0.0.1 machine.hp.com)

Race condition attacks - Also known as “*Time of check to time of use” (TOCTTOU).*[9] A simple way to think of this type of attack is to imagine you are a programmer finishing up your program. You have looked over the program code one final time before you compile it and are sure of what it is supposed to do. You are about to exit your editor and compile your program but then the phone rings. You turn away from your editor to answer the phone. Then someone quickly slips in, adds some unexpected code to your program and runs away before you finish the call. You being unaware of any changes to your code compile the program. When you run the program unexpected things happen. The period from when you last checked the code to when you compiled and ran it is called a “race condition”. An example for computer systems in which this can happen is when a program checks to see if a file exist before it changes ownership of the file. The time between the check and the ownership change could be enough to allow an attacker to modify the file resulting in behavior that could cause a security breach. The alert generated:
“Filename mapping change”

“Filename mapping change” detail alert format:
UID:UID (EUID:EUID) Reference:PATHARG currently kern_SYSCALL1:PATH1(FILEINFO1) was kern_SYSCALL2:PATH2(FILEINFO2) program running is PATH3(FILEINFO3) with arguments [ARGLIST3] ATTACKER was UID:A_UID running PATH4(FILEINFO4) with arguments [ARGLIST4]

“Filename mapping change” detail alert example:
UID:0 (EUID:0) Reference:/dev/emsagent_fifo currently kern_open:/dev/emsagent_fifo(8,1282,"40000003") was kern_mknod:/dev/emsagent_fifo(0,-1,"ffffffff") program running is /etc/opt/resmon/lbin/emsagent(1,1429,"40000003") with arguments ["/etc/opt/resmon/lbin/emsagent"] probable ATTACKER was UNKNOWN

Repeated Failed Logins- If an attacker is trying to gain access to your system by guessing passwords, this template could be used to alert you that a threshold of login failures has been reached. Many tools are available that can be used to

provide a systematic password-guessing scheme known as a “brute force” attack. HP-UX uses the crypt() function to encrypt passwords. “The algorithm that crypt() uses is based on the Data Encryption Standard (DES)”.[8] The key length of DES is 56 bit or 8 keyboard characters. Considered strong at the time it was created, DES encryption can be quickly cracked today using an exhaustive key search. The alert generated:
“Failed login attempts”

“Failed login attempts” detail alert format:
More than LIMIT failed logins by user USER (REMOTE: HOST IP)

“Failed login attempts” detail alert example:
More than 2 failed logins by user root (REMOTE: machine.hp.com 127.0.0.1)

Repeated failed su attempts- Under friendly circumstances a number of failed “su” attempts by an authorized user could indicate that they are having problems remembering the password. This could result in the authorized user writing the root password down, changing the password to something simple, or creating a back door into the system. Password assistance needs to be given to an authorized user that shows many failed “su” attempts in the sulog to ensure that system authentication is not compromised. Often times a non-authorized user or attacker has gained access to a system as a regular user. The attacker will attempt to “su” to root by guessing the password. This template will alert if a specified number of repeated failed su attempts are detected. The alert generated:
“Multiple failed su attempts by FROMUSER”

“Multiple failed su attempts by FROMUSER” detail alert format:
User "FROMUSER" had at least MAXCOUNT failed su attempts in
the past TIME. Targets included USERLIST

“Multiple failed su attempts by FROMUSER” detail alert example:
User "ids" had at least 2 failed su attempts in the past
24h. Targets included ["root"]

**NOTE: The Race Condition and Buffer Overflow templates impose the most system overhead so use them wisely.

Getting started with IDS/9000.

The following are prerequisites for running IDS/9000
version 2.0:
• HP-UX servers running HP-UX 11.0, 11i, or higher for each
host system; the same for the administration console.
• Java™ JRE or Java JDK version 1.3.1.
• 25 MB disk space for each host and 25 MB for one management system.

Installation involves:

- Installing the System Manager server software
- Installing the agent software on each host
- Generate key pairs for System Manager and agent host

Once the installations are done, the first thing to determine in an IDS/9000 environment is what to monitor for, and secondly how do I respond to an incident. IDS/9000 uses a client/server model in which a client runs IDS/9000 agent software. The agent software is configured to execute a "surveillance schedule" or a collection of detection templates running within a time window. If an alert is detected the agent will log incidents locally and will forward the alert to the IDS/9000 server or System Manager. The agent communicates via Secure Sockets Layer (SSL) back to the System Manager.

The IDS/9000 System Manager has five operations screens to manage the product:

- System Manager
- Schedule Manager
- Host Manager
- Network Node
- Preferences

System Manager- The System Manager displays what schedules are applied to clients, the status of the agent, the host name and IP, and alert information.

Schedule Manager- The Schedule Manager allows you to specify what you are looking to get alerted on within a time interval. You configure detection template properties, such as what files to monitor for change. Here you can determine which detection templates will make a surveillance group, which groups make a particular schedule, and when the schedule will run. In an effort to make IDS/9000 work out of the box, HP has included many useful predefined surveillance schedules that can be immediately applied to monitored host.

Host Manager- The Host Manager screen is where host to be monitored by IDS/9000 are added and removed.

Network Node- The Network Node screen is where IDS/9000 alerts and error messages are sent in near real-time. Alerts are categorized as:

- Red (severity 1) which is a direct system compromise
- Yellow (severity 2) which is a non-fatal alert
- Blue (severity 3) which is informational

The Alerts screen provides alert category, attacker (host ID or IP), attacker type, and date/time stamp.
The Error tab of the Network Node screen provides information about errors reported from the IDS/9000 agent.

Preferences- The Preferences screen allows you to specify operation preferences and choose which columns and data appear in above mentioned configuration screens.

Response Programs

Response Programs allow you to extend IDS/9000 alert functionality by allowing automated reactions to alerts. In addition to alerts being reported to the System Manager, possible Response Program actions could be:

- Forward alert information via pager or email
- Change system attributes to halt further attacks
- Stop system processing
- Disable network connections
- Restore a system to a good known state

Response Programs can be custom coded to respond to alert events. IDS/9000 can run Response Program shell scripts and executables reducing risk to your business.

IDS/9000 integration with HP OpenView Operations (OVO)

If you are running OVO in your enterprise you can integrate IDS/9000 into the OVO environment allowing you to directly manage alerts from the OpenView management server. HP OVO offers a Smart Plug-in (SPI) for IDS/9000 with components for log file, process, alert monitoring, and overall application availability. IDS/9000 uses the HP OpenView API to forward alerts to OVO agent, which then forwards the alert to the OpenView management server.[10]

Conclusion

In addition to hardening your HP-UX systems and providing network based intrusion detection, IDS/9000 for your host will complement your security strategy. What separates IDS/9000 from other IDS solutions is that it is made by HP for HP-UX systems. It's near real time monitoring, kernel integration, ease of setup, integration with HP OpenView, and free price tag, makes it an excellent solution for your HP-UX enterprise.

References

1) Computer Security Institute. “CSI/FBI Computer Crime And Security Survey” May 29, 2003. URL: http://i.cmpnet.com/gocsi/db_area/pdfs/fbi/FBI2003.pdf

2) Pipkin, Donald. Information Security. Upper Saddle River: Prentice Hall. 2000. Page 191.

3) Internet Security Systems. “Network versus Host-Based Intrusion, A Guide to Intrusion Detection Technology”. October 1998. URL:http://documents.iss.net/whitepapers/nvh_ids.pdf

4) Cole, Eric; Fossen, Jason; Northcutt, Stephen; Pomeranz, Hal. SANS Security Essentials with CISSP CBK, Volume 1 The SANS Institute. 2003

5) Crosbie, Mark; Kuperman, Benjamin. “A Building Block Approach to Intrusion Detection”. 2001.URL: http://www.raid-symposium.org/raid2001/slides/crosbie_kuperman_raid2001.pdf

6) Frederick, Karen. “Network Intrusion Detection Signatures, Part One”. 2001. URL: http://www.securityfocus.com/infocus/1524

7) Security Focus. “Smashing The Stack For Fun And Profit” November 9, 1996. URL: http://www.securityfocus.com/archive/1/5667

8) Garfinkel, Simson; Spafford, Gene. Practical UNIX and Internet Security. Sebastopol: O’Reilly. 1996. Page 247

9) Lowery, Craig. “A Tour of TOCTTOUs”. 2002 URL: http://www.sans.org/rr/paper.php?id=1049

10) Hewlett-Packard. “Smart plug-in for hp IDS/9000”. 2002. URL: http://www.openview.hp.com/products/spi/spi_ids/

Other Sources

Hewlett-Packard. “HP-UX Intrusion Detection System”. 2002. URL:http://www.hp.com/products1/unix/operating/infolibrary/briefs/intrusiondetectionpb.pdf

Hewlett-Packard. “HP Intrusion Detection System/9000 Administrator’s Guide”. 2002. URL: http://docs.hp.com/hpux/pdf/J5083-90007.pdf