Global Information Assurance Certification Paper

Sans Security Essentials GSEC Practical Assignment Version 1.2d
Current as of December, 2000 (amended April 27, 2001)

Aide: Poor man's file integrity checker

Tapan Meshram
1st May, 2001

Introduction:

The three basic dimensions of the information system that we are trying to protect are confidentiality, integrity and availability. In order to safeguard our information, the security infrastructure should consist minimally of firewall (perimeter devices), intrusion detection system (network and host both), anti virus tool and file integrity checker.

A firewall enforces rules on outgoing and incoming traffic. An intrusion detection system isolates malicious activity by matching network traffic with the available signatures in its database of network and host attacks. An anti virus tool protects one from virus code by matching it with the available virus definition.

However an attacker can spoof an IP address to exploit the vulnerability of firewall. Further, if the exploit is not recognized by the intrusion detection system then it will pass the check and sensitive information will get exposed. Similarly, a virus can infect and change our files if it escapes the anti-virus tool. Thus, inspite of these three tools the system may get compromised. At this point a file integrity checker comes in. The file integrity checker can keep a watch on key system binaries, libraries, header files - actually all files that are expected to remain the same over a period of time - and its reports can identify the change made to the file.

Thus, a file integrity checker lies at the core of a security infrastructure and is a fairly accurate way to identify the compromise of our information system. (Refer to the section "principle" for details)

Structure of the document:

Section 1: Discusses the various open source tools (except Aide) available for file integrity checking.

Section 2: Discusses the principle, features & limitations of Aide (a free replacement for Tripwire).

Section 3: Gives a step by step methodology to overcome some of the limitations.

Section 4: Lists the future features, the final word and references.

Note:

- I have access to only a Linux machine so all commands are valid for Red hat Linux box.
- Commands to be executes on the shell prompt are in italics
- The information on supported platforms for each tool is from the documentation available for the tool.
- In order to save space I have used " | " symbol instead of "and " or a line break.

SECTION 1

Tool: tripwire 2.3.1-2
Download: http://www.tripwire.org/
Supported platforms: Windows NT 4.0, Windows 2000, Solaris (SPARC) 2.6, 7.0 & 8.0, IBM AIX 4.3, HP-UX 11.0, Linux.
Description: Tripwire verifies system integrity based on certain policies. Initially a database is created containing attributes of critical files and directories. Subsequently the properties in the database are compared with the current attributes and thus any changes, including addition and deletion, are detected and reported. These reports can be used by the system administrator, to verify the system's integrity. The commercial version of the product has additional features such as sending the report by email to a specified mail address, encrypting the resultant database and report files to prevent access by unauthorized individuals.

Tool: Fcheck 2.07.45
Download: http://sites.netscape.net/fcheck/
Supported platforms: Linux, Solaris, SunOS, AIX, BSD, HP/UX, SCO, and various Windows machines running Perl 4.0 or better.
Description: Fcheck is an open-source tool, written in Perl. It allows administrators to enforce policies on servers and detect host intrusions. The method of verifying integrity is similar to Tripwire, i.e. it maintains a database and compares file checksums against those stored in the database. The advantage of Fcheck is that it has a simpler design and can monitor the system more frequently, as often as every minute. Moreover, if any changes are detected the resultant report can be sent utilizing any event management system, such as sending it directly to a printer or by email to a specific mail address. Please note that it is advised to install the security patch for this version, since it is vulnerable to local exploit.

Tool: filetraq 0.2
Download: http://filetraq.xidus.net/
Supported platforms: Linux Red Hat 6.0
Description: FileTraq is a simple shell script. It is most beneficial if run periodically using cron. It works in the following way. A backup copy is maintained of each system file that it is supposed to track. When FileTraq is run, it compares the current copy to the file with the backup copy using 'diff'. If any discrepancies are found, these are reported, and a dated backup of the original is also available. FileTraq is basically meant to maintain a history of changes to specified files. This serves the dual purpose of detecting intruders as well keep track of one's own changes to the files. An advantage is that a backup is available in case the file gets deleted by mistake. However, please note that FileTraq handles only text files, and cannot be applied to system binaries. Another disadvantage is that if the backup maintained of the files is not properly managed, then it may end up utilizing a lot of disk space.

Tool: OsirisScripts 1.3.0
Download: http://www.shmoo.com/osiris/
Supported platforms: Sparc Solaris 2.6 and 2.8, FreeBSD, OpenBSD, BSDI, Linux, Windows NT 4.0, Windows 2000 and MacOSX

Description: OsirisScripts is another tool written in Perl. It is meant to keep track of changes to executable files by maintaining a list of MD5 hashes of the files, and using the same list to compare the MD5 hashes of the current files. In this way it is able to detect whether a new file has been added or an existing file been deleted. It is also able to report changes to files in case the MD5 hashes do not match or there are changes in the properties of a file such as modification date. The basic advantages of Osiris are that it is open-source and the reports are simple and well organized. However, Osiris is currently meant for manually scanning a single system and does not allow automation. The new release has several new features, such as allowing use of Haval and SHA hashes, and maintaining a counter to keep track of Osiris' progress when it is indexing files.

Tool: Sentinel 1.2.0c
Download: http://zurk.netpedia.net/zfile.html
Supported platforms: Redhat Linux v6.0, slackware Linux v3.x & 4.xb and IRIX (v5.2 and v6.x).
Description: Sentinel is a fast utility developed on the lines of Tripwire, which may be used to track changes to files or complete drives. Like Tripwire, it detects changes by comparing with file information stored in a database. However, it utilizes RIPEMD-160bit MAC checksum generation algorithm, instead of the standard MD5 128 bit checksum, thereby making it more secure. This means that it is more difficult for an intruder to avoid detection of the changes made by him, since the checksum cannot be easily manipulated. An added feature of Sentinel is a user-friendly graphical interface called gSentinel, though it is still under the process of being developed.

Tool: Sherpa 0.1.3
Download: http://www.nbank.net/~rick/sherpa/
Supported platforms: Redhat 5.x/6.x, SuSE 6.0 and all Modern unix boxes.
Description: Sherpa is a tool developed using Perl. It can be utilized to configure and maintain a database, which has a list of files and directories and the corresponding permissions and ownership attributes, which can be set by the administrator as per requirements. Sherpa tracks changes to the files/directories by comparing current attributes with those stored in the database. An added feature is that Sherpa allows an administrator to do basic system checks such as world-writable files, SUID and SGID files, directory permissions, .rhosts and hosts.equiv files, etc.

SECTION 2
AIDE (Advanced intrusion detection environment)
Creators
Rami Lehti, rammer@cs.tut.fi & Pablo Virolainen, pablo@cs.tut.fi

Supported platforms: Basically AIDE runs on any modern Unix operating systems. People are using Aide on the following operating sytems: Solaris 2.5.1,2.6,7 | Linux 2.2.x, 2.0.x | FreeBSD 2.2.8,3.4 | Unixware 7.0.1 | BSDi 4.1 | OpenBSD 2.6 | AIX 4.2 | TRU64 4.0x

Principle: Increased safety with a file integrity checker

Once an intruder has access to the system, he will try to hide his presence. This is done to hide his footsteps and to keep the system compromised for a long period. A system administrator would use system tools like ps, netstat, lsof and who to identify the existing network connections and services. But a smart intruder will replace these tools with his trojaned version so that the user cannot view a suspicious activity. This can be achieved easily using rootkits available on hacking websites.

Though it is easy to manipulate the file dates and size but manipulating a cryptographic checksum of the file can be a difficult task. Moreover, manipulating multiple cryptographic checksums like md5 can be exponentially difficult. Aide generates about 7 checksums, if we compile it with mhash library support (enabling mhash support is explained later, in the "step by step" section). So by the comparison of the old checksum with the recent checksums, one can identify the changed files with a high degree of confidence.

AIDE constructs a database of the files specified in aide.conf, aide's configuration file. The AIDE database stores various file attributes including: permissions, inode number, user, group, file size, mtime and ctime, atime, growing size and number of links. AIDE also creates a cryptographic checksum or hash of each file using one or a combination of the following message digest algorithms: sha1, md5, rmd160, tiger (crc32, haval and gost can be compiled in, if mhash support is available).

An Aide database (initial snapshot of the system) should be created before the system is brought on the network. The program should take another snapshot later and list the differences in a report. The database should contain information about the key system binaries, libraries, header files and all other files, which are expected to remain same over a period of time. As far as possible we should avoid files, which keep changing (e.g.: log files, mails spools, proc file-systems, etc).

Current Features [1]

- Multiple integrity checking algorithms (even more with mhash support)
- Ability to output the database to stdout/file
- Easy configuration through a powerful configuration file
- Compressed databases (zlib support)

Limitations

The binary, configuration file, policy file and database of the file integrity checker itself can be manipulated. In case of Aide, the policy file and configuration file are same. The following section will help you to target last three problems. It gives a method to transfer the configuration file and database to a floppy disk for increased security.

[1] Lehti, Rami. 'README' with the aide-0.7.tar.gz source code.

SECTION 3

Aide step by step

1. Download

The latest version of Aide (size=220KB, format= tar.gz) can be downloaded from the following address:

ftp://ftp.cs.tut.fi/pub/src/gnu/aide-0.7.tar.gz.

http://www.cs.tut.fi/~rammer/aide-0.7.tar.gz.

Aide should be ideally started on the machine before it is connected to the network. Therefore we will transfer it to the target machine (now onwards called the host) using a floppy.

2.Making a list of important files

Lets first identify the suid & sgid files and list them in a file called imp_files.txt

Note: This list is not exhaustive.

Use the following commands:

find / -type f -perm -4000 -print >impfiles.txt

find / -type f -perm -2000 -print >>impfiles.txt

Other important files are as following:

Find the following files and add them to impfiles.txt using an editor like pico.

/etc/passwd and /etc/shadow files | /bin/ls | /bin/netstat | /usr/bin/lsof | /usr/bin/who | /var/log/btmp | /var/run/utmp | Any root (/) level "dot" files .rhosts, .profile etc. | System start up files (rc.*) | Application binary that has been installed | Compiler, linker and associated libraries | /etc/hosts | /etc/aliases (mail aliases) | /etc/services | /usr/bin/who |.cshrc (C-shell initialization file) | .forward (Tells /usr/lib/sendmail where to forward your electronic mail) | .kshrc (Korn shell initialization file) | .login (C-shell initialization command script, which executes at login) | .netrc (FTP initialization and macros; can also store clear text passwords) | .profile (Bourne and Korn shell initialization commands) | .rhosts (Contains the names of remote accounts that can log in using rsh and rlogin without a password) |.xinitrc (Start-up file for xinit) | /opt (for any optional OS components installed) .

3.Begin Installation

First, go to the directory where the compressed file is lying:

cd /{source directory}..

4.Uncompress the file:

tar -zxvf aide-0.7.tar.gz

This will generate a directory in called aide-0.7 .Go to this directory

cd aide-0.7/

5. Configuring Aide

Configure Aide with the mhash support (If you want to use mhash support the you must have Mhash library version 0.8.1 or newer installed. You can get it from http://mhash.sourceforge.net/) with the following command:

./configure --with-mhash

6.Compile it with:

make

7. Install it with:

make install

8.Transfer to floppy:
As explained earlier it is not safe to keep the configuration file and the initial snapshot (the database of file permissions with checksums) on the host.
We need to ensure that the configuration file called aide.conf is on a floppy. So we will put a fresh floppy into the drive and mount it on /mnt/floppy using the following command:

mount /dev/fd0 /mnt/floppy

Lets now transfer the configuration file to the floppy. Assuming I am in the directory aide-0.7, the command will be:

mv doc/aide.conf /mnt/floppy/aide.conf

9. Modifying configuration file:
Add the files we need to monitor at the end of the configuration file using the following command:

cat /mnt/floppy/aide.conf imp_files.txt >> /mnt/floppy/aide.conf

9.1 Rule manipulation

9.1.1 To define a rule using the defined variables (eg: p, c, md5, etc) do :

All=p+i+n+u+g+s+m+c+a+md5+sha1+tiger+rmd160

This line defines group All. It has all attributes and all md checksum functions. If you absolutely want all digest functions then you should enable mhash support and add +crc32+haval+gost to the end of the definition for All. Mhash support has to be enabled at compile time.

9.1.2 To add the "All" rule to /data/secret/ we can use the following command:

/data/secret All

9.1.3 To ignore a particular file we can use the following syntax:

!/data/secret

9.1.4 To include files in a particular directory and not in its children directories do:

=/etc

Then only files present in /etc is taken into the database and it's children are ignored.
Note: The files we added to the configuration file are very important so use the rule " All " for them.

9.2 Sending database to floppy
We would now like that the database generated by Aide should go to the floppy so make the following changes to the configuration file:

database_out=file:@@{TOPDIR}/doc/aide.db.new

should become

database_out=file:/mnt/floppy/aide.db.trust

This will bring the output database to our floppy when we run Aide.

9.3 Storing result in a file on floppy

We would also want the results to be stored to a file so that we can go through the changes. Write the following in the report_url block

report_url=file:/mnt/floppy/result.txt

And comment out the default output line.

 # report_url=stdout line

10.Initialise the database

Remember to use the configuration file lying on the floppy with:

aide -c /mnt/floppy/aide.conf --init

The above step will put aide.db.trust & result.txt in the floppy
The result.txt will be empty and aide.db.trust is the snapshot of the attributes of the file along with the checksums.

11.Check, trim, update (CTU) cycle

The procedure to find out the changes in file attributes and violations is very easy. First, we would like to use the aide.db.trust database for comparison with the new database. So change the configuration file to use aide.db.trust as the database.

11.1 Access the mounted floppy

cd /mnt/floppy/

11.2Open the configuration file in any editor (we are using the pico editor)

pico aide.conf

11.3 Change input database:

database=file:@@{TOPDIR}/doc/aide.db

to

database=file:/mnt/floppy/aide.db.trust

11.4 Change output database:

database_out=file:/mnt/floppy/aide.db.trust

to

database_out=file:/mnt/floppy/aide.db.new

11.5 Save and close

For the Pico editor use

Ctrl-X

Y
Return

12 Checking
We can run the check on file modifications using the following command
aide -c /mnt/floppy/aide.conf --check

13. Sample Report
The last command will generate a report in the file results.txt on the floppy. To illustrate the report format I had put check only on /var/log/btmp . This file is responsible to store all failed log in attempts. The sample report is as following.

AIDE found differences between database and filesystem!!
Start timestamp: 2001-04-26 13:46:36
Summary:
Total number of files=1,added files=0,removed files=0,changed files=1
Changed files:
changed: /var/log/btmp
Detailed information about changes:
File: /var/log/btmp
Atime: old = 2001-04-25 13:49:59, new = 2001-04-26 13:12:09
End timestamp: 2001-04-26 13:46:38

---

14. Update only
Alternatively one can also use the update command
aide -c /mnt/floppy/aide.conf --update
The update command also does the same thing as check but it also creates a new database. The input and output databases must be different.

SECTION 4
Future features in Aide

> The creators of Aide are doing a great job and the features we may have in future are, Multiple database retrieval backend | Encrypted databases | Return Codes | Symlink change | Windows NT port | Email report | More elaborate report options | Recurse=n | Interactive db update | Is file really deleted or has the configuration changed? | Re-write for smaller memory footprint. [2]

[2] Lehti, Rami. 'README' with the aide-0.7.tar.gz source code.

Final words:
Do continue the CTU (check, update, trim) cycle on the configuration file regularly to keep your report short and relevant. Please make the best use of this document and immediately protect your information with Aide.

References

1. McClure Stuart, Scambray Joel & Kurtz George. Hacking Exposed Network Security Secrets and Solutions, 1st Edition. New Delhi: Tata McGraw-Hill Edition 2000.

2. Garfinkel Simson & Spafford Gene. Practical Unix & Internet Security, 2nd Edition. O'reilly Publication. August 1999.

3. Gumienny, Michael. "Who's Changing Your Systems?". 20th Sep, 1999. URL: http://www.geocities.com/fcheck2000/ (1st May, 2001)

4. Weatherford, Jeremy. "filetraq.xidus.net". 10th Jan, 2000. URL: http://filetraq.xidus.net/README (1st May, 2001)

5. "Osiris". 8th May, 2001. URL: http://www.shmoo.com/osiris/ (1st May, 2000)

6. Lehti, Rami. "Aide". 4th Aug, 2000. URL: http://www.cs.tut.fi/~rammer/aide.html (1st May, 2001)

7. Lehti, Rami. "The Aide manual". 7th Feb, 2001. URL: http://www.cs.tut.fi/~rammer/aide/manual.html (1st May, 2001)

8. "Max Vision's Whitehats". 30th Apr, 2001. URL: http://www.whitehats.com > free tools > IDS File Integrity (1st May 2001)

9. Lehti, Rami. 'README' with the aide-0.7.tar.gz source code. URL: ftp://ftp.cs.tut.fi/pub/src/gnu/aide-0.7.tar.gz. or http://www.cs.tut.fi/~rammer/aide-0.7.tar.gz. (1st May, 2001)

10. Paula McKeehan "Tripwire – An Integrity Assessment Tool". January 24, 2001. URL: http://www.sans.org/infosecFAQ/audit/tripwire.htm (1st May 2001).