# IDS Module 2 Lab

**Audrey Long**

**06/07/2020**

Audrey Long
JHU SU20 IDS Module 2 Lab Activity
Due: 8 JUN 2020 @ 1159pm EST

Please submit screenshots of all TripWire outputs (install, initialization, execution) as well as the answers to the questions in a single PDF or MS Word file on Blackboard

**Reading**

**1. Start by watching this brief video "What is hashing?" if you're not already familiar with hash functions.**

**What is TripWire?**

**Please answer the following questions:**
**2a. What are the components of TripWire as a HIDS?**

Tripwire is an integrity intrusion detection system which alerts when something changed within the system that should not have changed. Tripwire is a free and open sourced security and data integrity tool used for monitoring and alerting on systems and is equipped to deal with file integrity checking, cryptographic checksums, alerts on critical changed files.
The components of Tripwire which enable it as a host intrusion detection system include the constant monitoring of file integrity, alerting on critical files which have been tampered with or are changed. This system also checks on critical checksums, hashes, and can check other components within the system.

**2b. Describe how a "message digest" function is used by TripWire to protect data.**

The message digests used in TripWire to protect data use all sorts of checksumming algorithms. The TripWire source code comes equipped with RSA Data Security, MD5, MD4, and MD2, SHA, Snefru, and Haval message digest algorithms. TripWire defaults to the MD5 hashing algorithm on critical files and is also checked with Snefru, which is a cryptographic hash function which supports 128 bit and 256 bit output. The TripWire database checks all critical files for tampering by checking the hash.

**Working with TripWire**

**Working with Tripwire please provide screenshots upon the completion of each of the following steps (and explain as necessary).**

**3. Is TripWire currently installed on your Ubuntu VM? If it is not installed, install it by using the following**
command: sudo apt-get install tripwire. Verify that TripWire has been installed successfully.

```
student@student:~$
student@student:~$ sudo apt-get install tripwire
Reading package lists... Done
Building dependency tree
Reading state information... Done
tripwire is already the newest version (2.4.3.1-2).
0 upgraded, 0 newly installed, 0 to remove and 563 not upgraded.
student@student:~$
```

**4. Use the 'init' command to initialize the TripWire configuration.**

```
student@student:/var/lib/tripwire$ sudo tripwire --init
Please enter your local passphrase:
Parsing policy file: /etc/tripwire/tw.pol
Generating the database...
*** Processing Unix File System ***
### Warning: File system error.
### Filename: /root/.bash_history
### No such file or directory
### Continuing...
Wrote database file: /var/lib/tripwire/student.twd
The database was successfully generated.
student@student:/var/lib/tripwire$
```

**5. Into which directory does TripWire write it's "policy" and "configuration" files?**

```
student@student:/var/lib/tripwire$ ls
report  student.twd  student.twd.bak
student@student:/var/lib/tripwire$
student@student:/var/lib/tripwire$
student@student:/var/lib/tripwire$ cd /etc/tripwire/
student@student:/etc/tripwire$ ls
no-dictionary.txt  site.key          tw.cfg     tw.pol      twpol.txt
no-directory.txt   student-local.key  twcfg.txt  tw.pol.bak
student@student:/etc/tripwire$
```

In the /etc/tripwire folder we find the sample tw.cfg configuration file and the twpol.txt policy file when we create the policy file it ends up in /etc/tripwire as a tw.pol file

**6. If you want to expand the number of directories TripWire monitors, how would you do it?**

You would need to modify the configuration file.

**7. Create a suspicious incident on your Ubuntu VM by creating a new directory on your machine, updating the policy file, then reinitialize the database. Hint: use "vim" to add new directory/rulename,(e.g., ~$ vim /etc/tripwire/twpol.txt), then use sudo twadmin –m P /etc/tripwire/twpol.txt to update policy file, finally use sudo tripwire - -init to reinitialize the database.**

The following was added to the policy file

```
# Ruleset for Wordpress
(
  rulename = "Wordpress Ruleset",
  severity= $(SIG_HI)
)
{
```

```
        /var/www            -> $(SEC_CRIT);
}
```

### 8. Run TripWire and verify it detects the newly created directory.

```
Rule Name                     Severity Level    Added    Removed  Modified
---------                     --------------    -----    -------  --------
Other binaries                66                0        0        0
Tripwire Binaries             100               0        0        0
Other libraries               66                0        0        0
Root file-system executables  100               0        0        0
Wordpress Ruleset             100               2        0        0
(/var/www)
Tripwire Data Files           100               0        0        0
System boot changes           100               0        0        0
(/var/log)
Root file-system libraries    100               0        0        0
(/lib)
Critical system boot files    100               0        0        0
Other configuration files     66                0        0        0
(/etc)
Boot Scripts                  100               0        0        0
Security Control              66                0        0        0
Root config files             100               0        0        0
Devices & Kernel information  100               0        0        0
Invariant Directories         66                0        0        0

otal objects scanned:  44944
```

```
Total violations found:  2

===============================================================================
Object Summary:
===============================================================================

-------------------------------------------------------------------------------
# Section: Unix File System
-------------------------------------------------------------------------------

-------------------------------------------------------------------------------
Rule Name: Wordpress Ruleset (/var/www)
Severity Level: 100
-------------------------------------------------------------------------------

Added:
"/var/www"
"/var/www/hakase-labs.txt"

===============================================================================
Error Report:
===============================================================================

-------------------------------------------------------------------------------
  Section: Unix File System
-------------------------------------------------------------------------------

1.   File system error.
     Filename: /root/.bash_profile
     No such file or directory

-------------------------------------------------------------------------------
*** End of report ***
```

**TripWire Review**

**9. Will TripWire be able to detect all threats on a particular host or network? Please explain.**

No, Tripwire is a host intrusion detection tool which checks the integrity of the file system defined in the configuration and policy files. Aside from checking files added or modified in the specified configuration file this tool does not monitor network traffic, or contains any SIEM behaviors. This tool's intentions are for file system integrity.

**10. What are the ramifications if TripWire were to use a "broken" hash function and how might this benefit an attacker? (If you'd like a hint, you might want to look up pre/post-image collision resistance).**

The ramifications when using a broken hash function to benefit an attacker is the failure to verify the integrity of the files in the system using tripwire. Essentially collision resistance equates to different inputs which result in the same hash output, this is bad if you are trying to verify the integrity of the files for the hash will result in the same "good hash" even with altered inputs to the file. We know there are 2^256 possible hashing outcomes and many files which can map to a hash. It can never be guaranteed that identical hashes result in the same file inputs but it can be computationally infeasible for an attacker to brute force a hash with strong hash functions being utilized using pre image resistance by giving an input to find a second input with the same hash function or with collision resistance.

**11. Describe how TripWire was able to detect the suspicious incident you generated in step 7.**

In step 7 we added a new rule to the policy table which then started monitoring the file location specified with a high severity. Since we created a new rule TripWire will actively start monitoring that file location once we activate the tool to see if anything got added, modified, or deleted to that location when we generate the error report.

**12. How would you try to evade creating an "incident" if you were attacking a system you knew was protected by TripWire?**

If I was evading creating an incident as an attacker I would add a rule to the policy file to ignore the file locations I was particularly interested in. I would then run the tool to remove any previous instances and then delete the bash history.

**References:**

https://www.linuxjournal.com/article/2160

https://www.howtoforge.com/tutorial/how-to-monitor-and-detect-modified-files-using-tripwire-on-ubuntu-1604/