Network Forensics Project : Hunter

Made by : daniel kovalevsky
Class : 7736/21
Teacher : arel regev

Intro:

This script was made to run in a endless loop and to monitor the network , while monitoring it scans for malicious activity that accrued in the network and check's with the ioc list if matched Informs the user about the malicious activity like , urls , files , and ips saves it to logs and removes files that are larger than 1mb.

The script starts with variables for later use

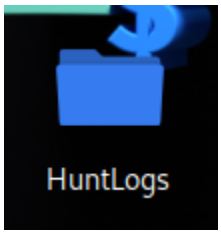
```
#!/bin/bash

################################
# Script Made By Daniel Kov     #
################################

#uses pwd to indicate where the script ran
HOME=$(pwd)
#gets the date
DATE=$(date)
#used to scan the network with user ip
IPSRC=$(ifconfig | grep inet | head -1 | awk '{print $2}' | awk -F. '{print $1"."$2"."$3"."0"/"24}')

############################################
# All Logs Will Be Saved To /HuntLogs/Logs   #
############################################


#variable for file loop
n=1
#variable for loop
loop_number=1
```

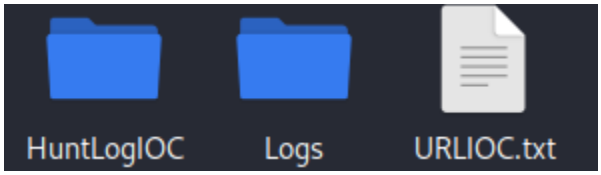
Function Download:

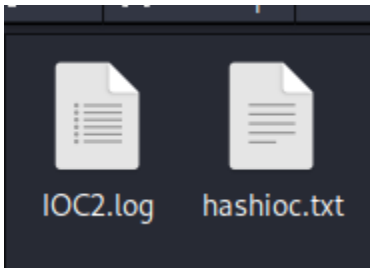
This function creates folders for the script to store the data that was extracted, then downloads the ioc lists and checks if it was downloaded already ,

```
#function that downloads ioc lists
function download()
{
    #create dir report
    mkdir $HOME/HuntLogs &> /dev/null
    #create dir for IOC's
    mkdir $HOME/HuntLogs/HuntLogIOC &> /dev/null
    #create dir for logs
    mkdir $HOME/HuntLogs/Logs &> /dev/null

        #checks if the ioc list already exists
        if [ -e "$HOME/HuntLogs/HuntLogIOC/IOC2.log" ]
        then

            echo "[+] IOC list Found"

        #if not found downloads
        else

            #downloads the ioc list
            wget https://feeds.dshield.org/top10-2.txt -O $HOME/HuntLogs/HuntLogIOC/IOC2.log &> /dev/null

            #extracting the url ioc list from it
            cat $HOME/HuntLogs/HuntLogIOC/IOC2.log | awk '{print $2}' | grep -v "NX" | sort | uniq >> $HOME/HuntLogs/URLIOC.txt 2> /dev/null
        #ends the statement
        fi
```

```
        #ends the statement
        fi

        #checks if the ioc list already exist
        if [ -e "$HOME/HuntLogs/HuntLogIOC/hashioc.txt" ]
        then

        echo "[+] HASH IOC Found"
        #if not found downloads
        else

        #downloads hash ioc list
        wget https://raw.githubusercontent.com/Neo23x0/signature-base/master/iocs/hash-iocs.txt -O $HOME/HuntLogs/HuntLogIOC/hashioc.txt 2> /dev/null

        fi


}
#calling the function
download
```

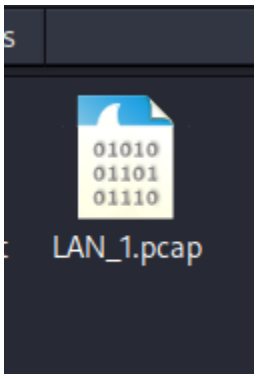


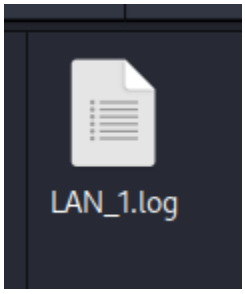




After that the script runs the tshark loop live with the user network And saves the recording in pcap file and txt , and stores the process id of the tshark to a variable.

```
#runs the tshark
while true; do
    echo "////// Hunter Is *Live* //////"
    echo "[+] Capturing Network"
    echo -e "[+] Logs Saved At: $HOME/HuntLogs , Loop number $loop_number"
        echo ""


    # Declare file names as variables
    pcap_file="LAN_$n.pcap"
    # Declare file names as variables
    log_file="LAN_$n.log"


    # Here we capture the packets using Tshark, for 30 seconds, using specific filters, and it will save it into both pcap and log files to use. Note that
    tshark -i eth0 -a duration:30 -w "$HOME/HuntLogs/$pcap_file" -T fields -e frame.number -e ip.src -e ip.dst -e http.user_agent "net $IPSRC" > "$HOME/Hun

    # here we get the PID of the command that was executed, in this case the shark command
    tshark_pid=$!

    # Sleep for 30 seconds to allow tshark to capture packets
    sleep 30
```






```
////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 1

Analysing Network: 192.168.71.0/24
```

Function Malicious:

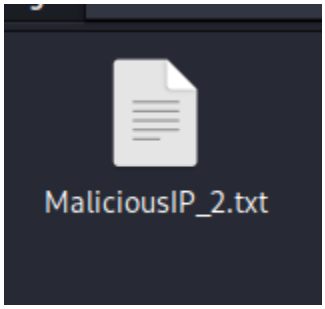
This function extracts from the pcap file communication data
And save it to txt file , after that it comperes the communication
Between the log file and ioc list to see if malicious ip was found.
If malicious ip was found it will inform the user and will save it to a log.

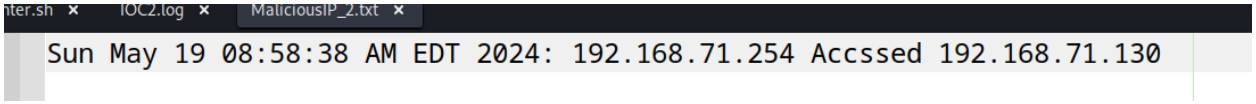
```
#function that checks if tshark found mailcious ip
function Malicious()
{
    echo "Analysing Network: $IPSRC"
    tshark -r $HOME/HuntLogs/$pcap_file -Y 'ip.addr' -T fields -e 'ip.src' -e 'ip.dst' >> $HOME/HuntLogs/HuntLogIOC/newfile$N.txt

        cat $HOME/HuntLogs/HuntLogIOC/newfile$N.txt | awk '{print $1,"Accssed",$2}' >> $HOME/HuntLogs/HuntLogIOC/Sorted$N.txt 2> /dev/null

            #for ioc list
            for NETIOC in $(cat $HOME/HuntLogs/HuntLogIOC/IOC2.log | awk '{print $1}')
            #does
            do
                #if ip from ioc list found
                if
                    grep -q $NETIOC "$HOME/HuntLogs/HuntLogIOC/newfile$N.txt"

                then
                    #informs the user the ip that found and saves to logs
                    HIP=$(cat "$HOME/HuntLogs/HuntLogIOC/Sorted$N.txt" | grep -w $NETIOC | head -1)
                    echo -e "[!] Warning Malicous Ip Detected"
                    echo -e "$DATE: $HIP " | tee -a $HOME/HuntLogs/Logs/MaliciousIP_$n.txt

                #ends the if statment
                fi
            #ends the for loop
            done

    #making blank space
    echo""

}
#calls the function
Malicious
```

After adding second ip to the ioc list

```
Analysing Network: 192.168.71.0/24
[!] Warning Malicous Ip Detected
Sun May 19 08:58:38 AM EDT 2024: 192.168.71.254 Accssed 192.168.71.130
```

The log it saves to





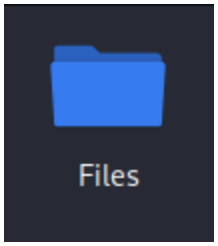
Function Files:

This function downloads from recorded tftp,http,smb,imf and saves the files into Files Folder and deletes everything that's above 1mb file

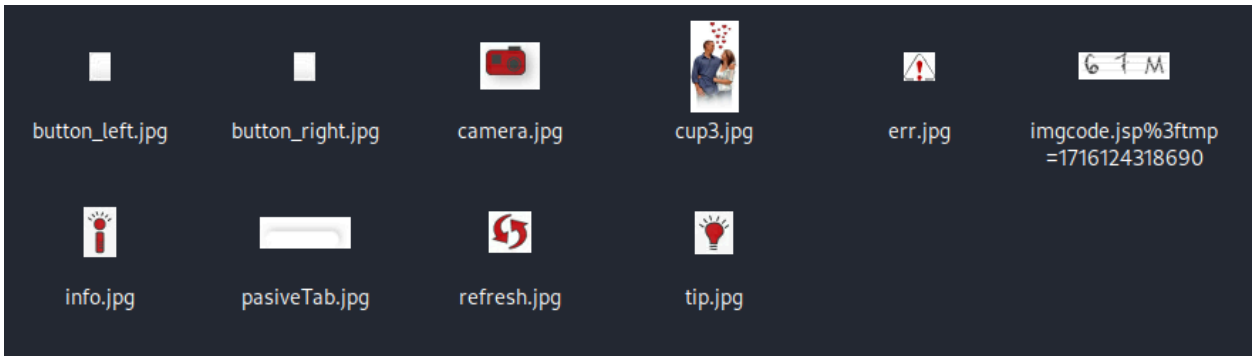
```
#function that extracts files found in the monitoring
function FILES()
{
    #variable to use the 1mb delete later on
    LIMIT=1000000
    #makes dir to where files will be downloaded
    mkdir $HOME/HuntLogs/Files 2> /dev/null

        #extracts from tshark files found in http , tftp , smb ,imf
        tshark -r $HOME/HuntLogs/$pcap_file --export-objects tftp,$HOME/HuntLogs/Files  &> /dev/null

        tshark -r $HOME/HuntLogs/$pcap_file --export-objects http,$HOME/HuntLogs/Files  &> /dev/null

        tshark -r $HOME/HuntLogs/$pcap_file --export-objects smb,$HOME/HuntLogs/Files   &> /dev/null

        tshark -r $HOME/HuntLogs/$pcap_file --export-objects imf,$HOME/HuntLogs/Files   &> /dev/null
```

```
    #for loop in dir Files
        for filelong in $(find "$HOME/HuntLogs/Files")
        do
            sleep 0.1

            #shortcut for the filename
            fileshort=$(basename "$filelong")

            #if statment for extracting size per file
            if [[ -f "$filelong" && $(stat -c%s "$filelong") -ge $LIMIT ]]; then

                #removes the files found with higher than 1mb
                rm "$HOME/HuntLogs/Files/$fileshort" && echo "[+] Deleted $fileshort : Larger Than 1mb" | tee -a $HOME/HuntLogs/Logs/RemovedFiles_$n.txt || ech

            #ends the if statment
            fi

        #ends the for loop
        done
```

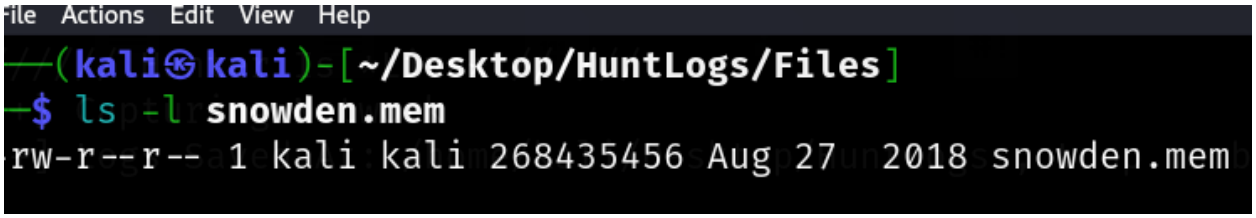
```
#blank space
echo ""

}
#calling the function
FILES
```



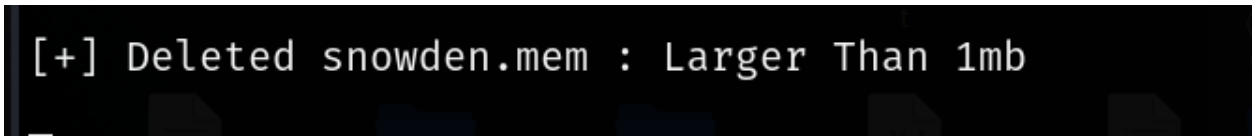
The downloaded files



Adding Snowden.mem to the folder.



The removal of larger than 1mb file



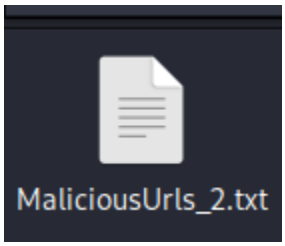
Function FileIOC:

This function sorts the hashioc and saves it to new file
After that it runs md5sum on the files folder saves it to a txt , and uses it to compare between the downloaded files and the hash ioc
If there was a match informs the user about it and store its into a log

```
#function that indicates the user if a malicious file found
function FileIOC()
{

    #sorts the ioc list
    cat $HOME/HuntLogs/HuntLogIOC/hashioc.txt | awk '{print $1}' | grep -v "#" | grep -v '^$' | awk -F";" '{print $1}' >> $HOME/HuntLogs/HuntLogIOC/HA

    #runs md5sum on the dir than saves it to txt report
    md5sum $HOME/HuntLogs/Files/* > $HOME/HuntLogs/HuntLogIOC/File_M5_hash.txt 2> /dev/null

    #checks if the hashes were extracted if does , makes a loop
    if [ -e $HOME/HuntLogs/HuntLogIOC/File_M5_hash.txt ]
    then
    #for loop that runs on hash ioclist
    for M5 in $(cat $HOME/HuntLogs/HuntLogIOC/HASHIOC.txt)
    do
        #if malicious file found
            if
                grep -q $M5 $HOME/HuntLogs/HuntLogIOC/File_M5_hash.txt
            then
                #informs the user the file found at which date
                grepout=$(cat $HOME/HuntLogs/HuntLogIOC/File_M5_hash.txt | grep -w $M5)
                echo "[!] Warning Malicous file located"
                echo "$DATE: $grepout" | tee -a $HOME/HuntLogs/Logs/MaliciousFiles_$n.txt
            #ends the if statment
            fi
    #ends the for loop
        done
fi
```

After adding a random url into the ioc list and entering it from other pc

```
[!] Warning Malicious Url Detected
Sun May 19 09:11:11 AM EDT 2024: www.perekbet.co.il
```




```
Sun May 19 09:11:11 AM EDT 2024: www.perekbet.co.il
```

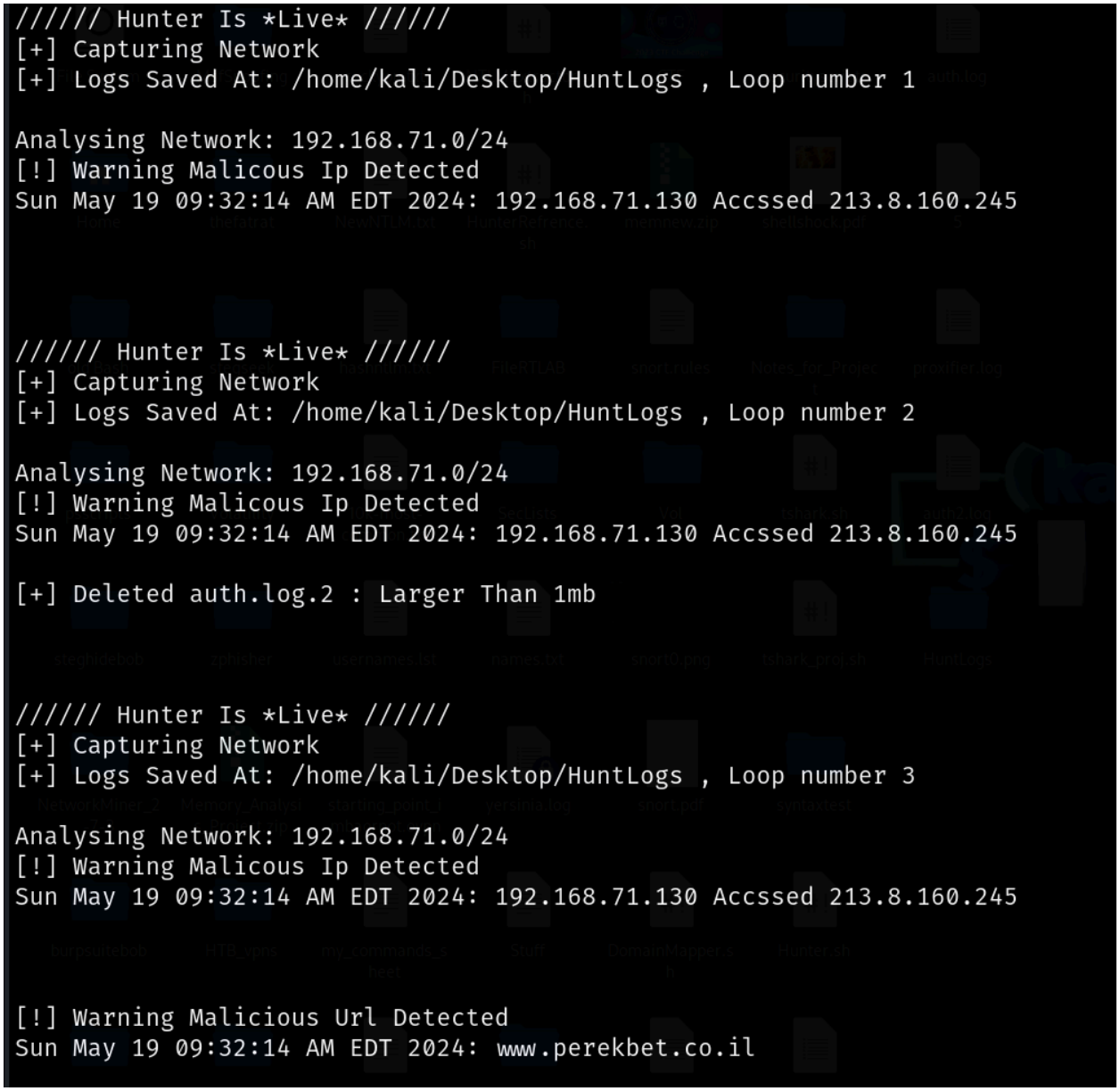
Then the script stops the process and loops the script again
With increasing the loop value and the number value by 1

```
# Kill the tshark process
    kill $tshark_pid  2> /dev/null

#note the variables on top. after every loop, it will add +1 to the current value. first loop would be 1.
    ((n++))
    #same here
    ((loop_number++))

#black space
echo ""

#stops the while true
done
```

```
└─$ ./Hunter.sh
////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 1
```

```
////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 2
```

And so on…
Full run with adding each loop something else to demonstrate real scenario



Update: After Adding Cosmetics only the functions are the same.:

The Added Stuff :

```
#color code for coloring
R="\e[31m"
E="\e[0m"
G="\e[32m"
C="\e[36m"
P="\e[35m"
O="\e[33m"
LR="\e[1;31m"
LP="\e[1;35m"
LB="\e[1;34m"
LC="\e[1;36m"
LG="\e[1;32m"
```

```
printf ${LP}
figlet "Tshark Live Script"
printf ${E}


echo "  ____________________"
echo "  | ________________ |"
echo "  | |$ tshark~~~~~| |"
echo "  | |~~~~~~~~~~~~~| |"
echo "  | |~~~~~~~~~~~~~| |"
echo "  | |~~~~~~~~~~~~~| |"
echo "  | |~~~~~~~~~~~~~| |"
echo "  |____________________|"
echo "     _[________]_"
echo "  ___[____________]___"
echo " |          [______] []|__"
echo " |          [______] []|  \__"
echo "L____________________J      \ \___\/"
echo "  ____________________       /\ "
echo " /####################\     (__)"
```

The color that were added to warnings , and few more stuff

```
echo -e "[${LR}!${E}] Warning ${LR}Malicous${E} Ip Detected"
```


(kali㉿kali)-[~/Desktop]
$ ./Hunter.sh
Tshark Live
Script
$ tshark~~~~~

```
////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 1

Analysing Network: 192.168.71.0/24
[!] Warning Malicous Ip Detected
Wed May 22 04:27:44 AM EDT 2024: 192.168.71.130 Accssed 213.8.160.245



////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 2

Analysing Network: 192.168.71.0/24
[!] Warning Malicous Ip Detected
Wed May 22 04:27:44 AM EDT 2024: 192.168.71.130 Accssed 213.8.160.245



////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 3

Analysing Network: 192.168.71.0/24
[!] Warning Malicous Ip Detected
Wed May 22 04:27:44 AM EDT 2024: 192.168.71.130 Accssed 213.8.160.245



////// Hunter Is *Live* //////
[+] Capturing Network
[+] Logs Saved At: /home/kali/Desktop/HuntLogs , Loop number 4

Analysing Network: 192.168.71.0/24
[!] Warning Malicous Ip Detected
Wed May 22 04:27:44 AM EDT 2024: 192.168.71.130 Accssed 213.8.160.245

[+] Deleted Memory_Analysis_Project.zip : Larger Than 1mb
[+] Deleted networkminer.zip : Larger Than 1mb
[+] Deleted snowden.zip : Larger Than 1mb
```