Script which uses volatility 2.x + yara rules to extract tons of useful info

```
#!/bin/bash

# check if we have at least one file
if [ $# -eq 0 ]
  then
    echo "No memory dump supplied"
    exit 0
fi
DUMP=$1


# check if it exists
if test -f "$DUMP"; then
    echo "Starting volatility analysis of $DUMP"
else
    echo "$DUMP doesn't exist"
    exit 0
fi
# Displays the step and add nice separations in the report
function title(){
    #echo -e "\n"
    #printf "[INFO] Extracting $1"
    echo "[INFO] Extracting $1"
    printf "%0.s#" {1..125}>>$DUMP.volatility.report
    echo -e "\n\t$1" >>$DUMP.volatility.report
    printf "%0.s#" {1..125}>>$DUMP.volatility.report
    echo -e "\n">>$DUMP.volatility.report
}


# Cleaning up the mess
function clean_up(){
    echo -e "\n"
    echo -e "\n">>$DUMP.volatility.report
    sed -i '/Volatility Foundation/d' $DUMP.volatility.report
    rm $DUMP.output
}


# Removes the old reports if exist
if test -f $DUMP.volatility.report; then
    rm $DUMP.volatility.report
fi

if test -f $DUMP.yara.report; then
    rm $DUMP.yara.report
fi

echo "[INFO] Yara malware scan search in parallel"
```

```
vol.py -f $DUMP yarascan --yara-file="/opt/yara/rules/malware_index.yar" &>
$DUMP.yara.report &
P1=$!
# Detect Profile
echo "[INFO] Searching profile for $DUMP"
OUTPUT=`vol.py -f $DUMP imageinfo &>$DUMP.output`
PROFILE=`cat $DUMP.output| grep -oP "Profile\(s\) :\s+\K\w+"`


title "Analysis of $DUMP\n\tProfile: $PROFILE"

# output basic info into main.report - process
title "All processes"
PSSCAN=`vol.py -f $DUMP --profile=$PROFILE psscan &>>
$DUMP.volatility.report`
title "Running processes"
PSTREE=`vol.py -f $DUMP --profile=$PROFILE pstree &>>
$DUMP.volatility.report`
title "CMD lines"
PSDOT=`vol.py -f $DUMP --profile=$PROFILE cmdline &>>
$DUMP.volatility.report`
title "Interesting processes"
PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE pstotal -S &>>
$DUMP.volatility.report`
title "Hidden process"
PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE psxview &>>
$DUMP.volatility.report`

#Network Analysis
title "Open Connections-XP/2003"
PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE connections &>>
$DUMP.volatility.report`
title "TCP Connections"
PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE connscan &>>
$DUMP.volatility.report`

#Creates the folder if it doesn't exist
DUMPDIR="$DUMP-PROCs"
title "All process (dump in $DUMPDIR)"
if [ ! -d ./$DUMPDIR ];then
    mkdir "./$DUMPDIR"
fi
PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE procdump --dump-dir ./$DUMPDIR
&>> $DUMP.volatility.report`
title "Process dot"
PSDOT=`vol.py -f $DUMP --profile=$PROFILE psscan --output=dot --output-
file=./$DUMPDIR/psscan.dot &>> $DUMP.volatility.report`
title "SSL Certs"
PSDOT=`vol.py -f $DUMP --profile=$PROFILE dumpcerts -D ./$DUMPDIR --ssl &>>
$DUMP.volatility.report`
# Run yarascan
```

```
wait $P1
title "Yara malware"
cat $DUMP.yara.report >> $DUMP.volatility.report

#title "Yara all"
#PSTOTAL=`vol.py -f $DUMP --profile=$PROFILE yarascan --yara-
file="/opt/yara/rules/malware_index.yar" &>> $DUMP.volatility.report`

# Hash generation
#remove old hash
if test -f ./$DUMPDIR/hashlist;then
    rm ./$DUMPDIR/hashlist
fi
title "Hash generation"
for file in `ls ./$DUMPDIR/executable*.exe`; do shasum
$file&>>./$DUMPDIR/hashlist; done;
# add links to virustotal
sed -i 's/^/https:\/\/www.virustotal.com\/gui\/file\//' ./$DUMPDIR/hashlist
cat ./$DUMPDIR/hashlist &>> $DUMP.volatility.report
clean_up
```

the script :

volatility-check.sh.zip

Available on github: https://github.com/D4thToMS/cybersecurity/blob/main/volatility-check.sh