**Volatility:**

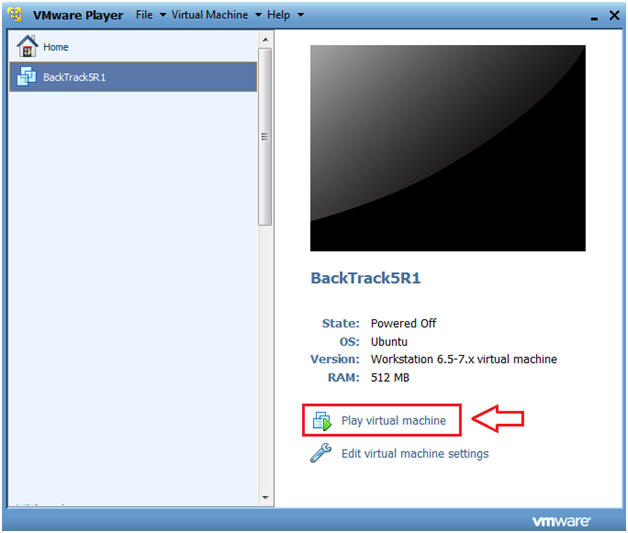
**Analyzing A SET Memory Capture from Windows XP SP2**

**Section 0. Background Information**

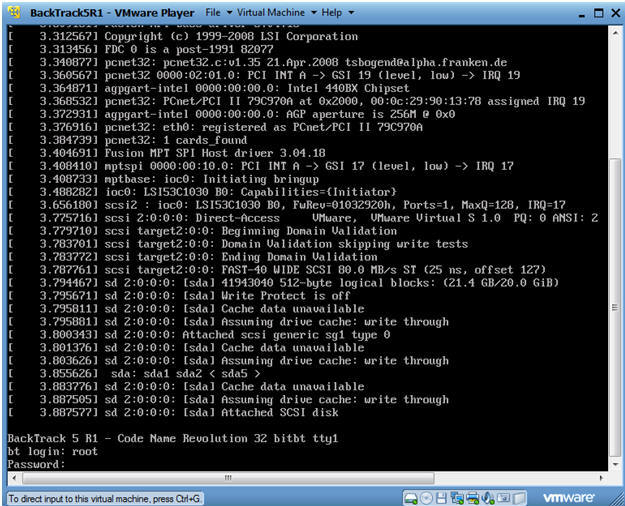
1. **Volatility Overview**
   * <https://www.volatilesystems.com/>
   * The Volatility Framework is a completely open collection of tools, implemented in Python under the GNU General Public License, for the extraction of digital artifacts from volatile memory (RAM) samples. The extraction techniques are performed completely independent of the system being investigated but offer unprecedented visibility into the runtime state of the system. The framework is intended to introduce people to the techniques and complexities associated with extracting digital artifacts from volatile memory samples and provide a platform for further work into this exciting area of research.
2. **Prerequisite Labs**
   * Social Engineering Toolkit (SET): Lesson 3: Create Malicious Weblink, Install Virus, Capture Forensic Images
3. **Lab Notes**
   * In this lab we will do the following:
     1. Download Volatility 2.2.
     2. Un-Tar Volatility 2.2.

**Section 1. Login to BackTrack**

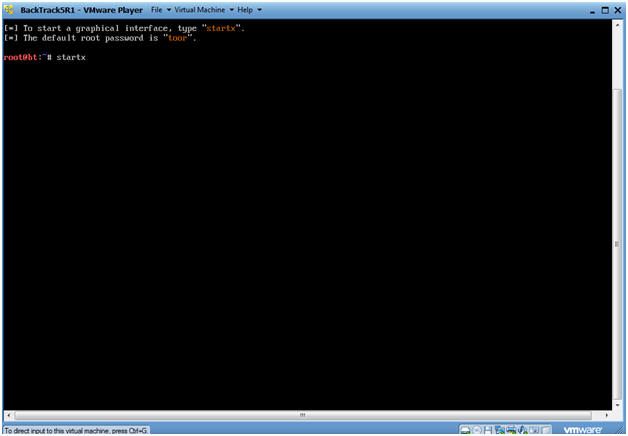
1. Start BackTrack VM Instance
   * **Instructions:**
     1. Start Up VMWare Player
     2. Select BackTrack5R1
     3. Play virtual machine



1. Login to BackTrack
   * **Instructions:**
     1. Login: root
     2. Password: toor or <whatever you changed it to>.

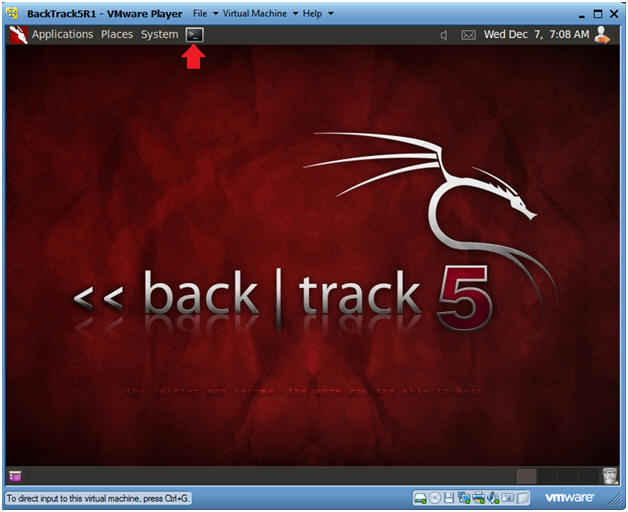


1. Bring up the GNOME
   * Instructions
     1. Type startx



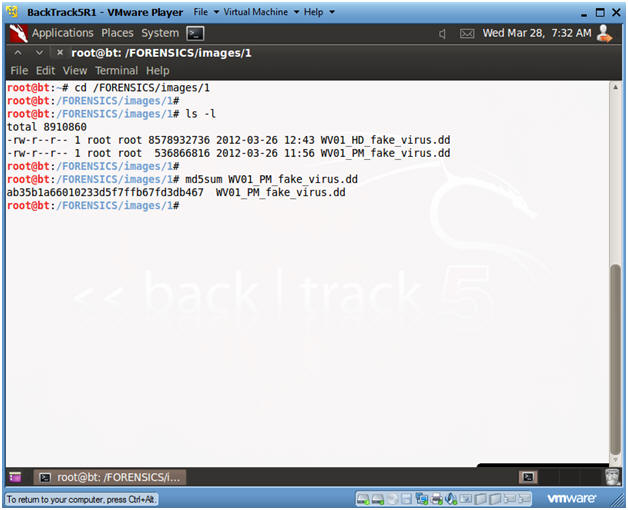
**Section 2. Bring up a console terminal**

1. Bring up a console terminal

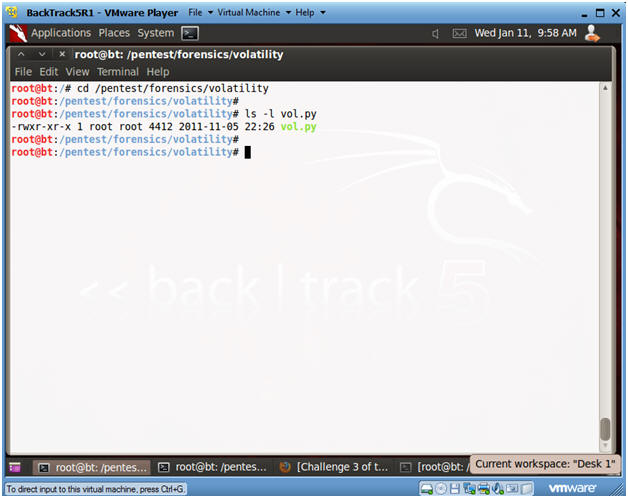


**Section 3. Using Volatility**

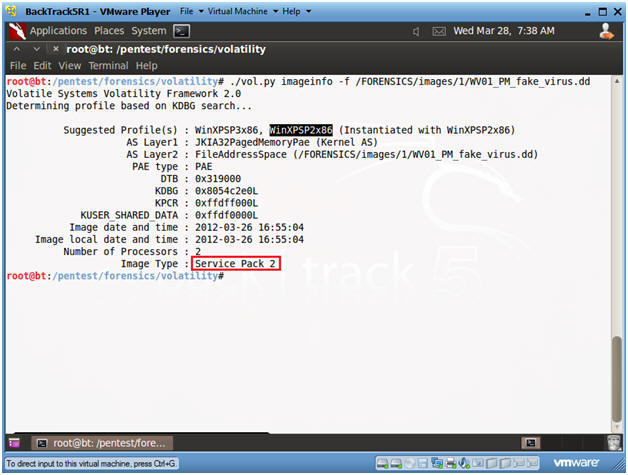
1. Navigate to Forensic Image
   * **Instructions**
     1. cd /FORENSICS/images/1
     2. ls -l
     3. md5sum WV01\_PM\_fake\_virus.dd



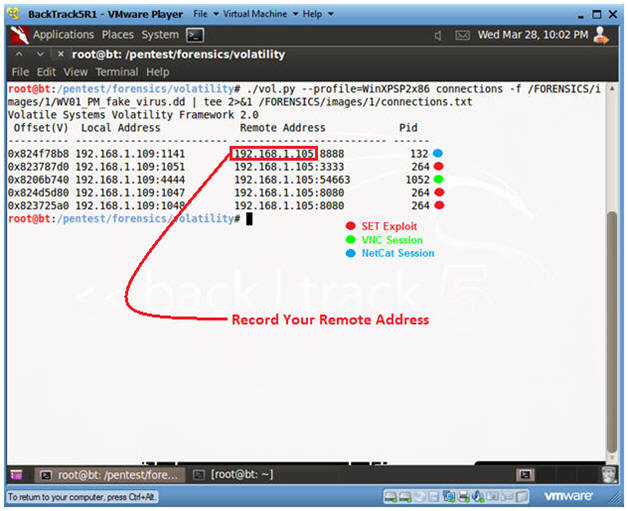
1. Navigate to Volatility
   * **Instructions**
     1. cd /pentest/forensics/volatility
     2. ls -l vol.py



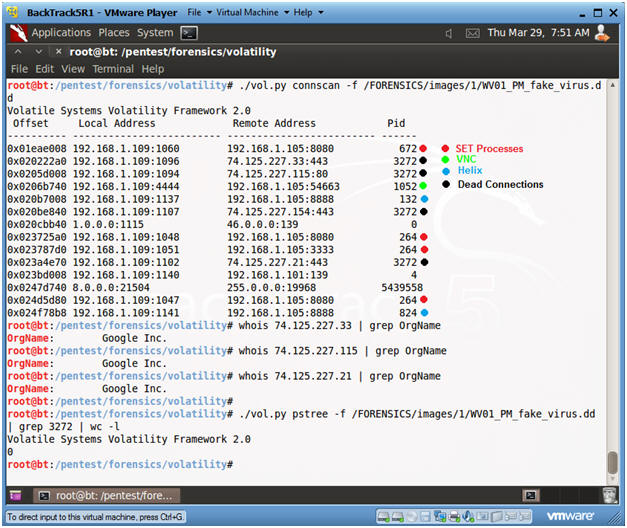
1. Obtain the image profile
   * **Instructions:**
     1. ./vol.py **imageinfo** -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd
   * **Notes(FYI):**
     1. The Volatility Framework tries to guess and tell you what image profile to use.
     2. We know that our Server is Windows XP running SP2.
     3. Volatility suggest that we either use the profile WinXPSP3x86 or WinXPSP2x86.
     4. We will use the **WinXPSP2x86** profile.



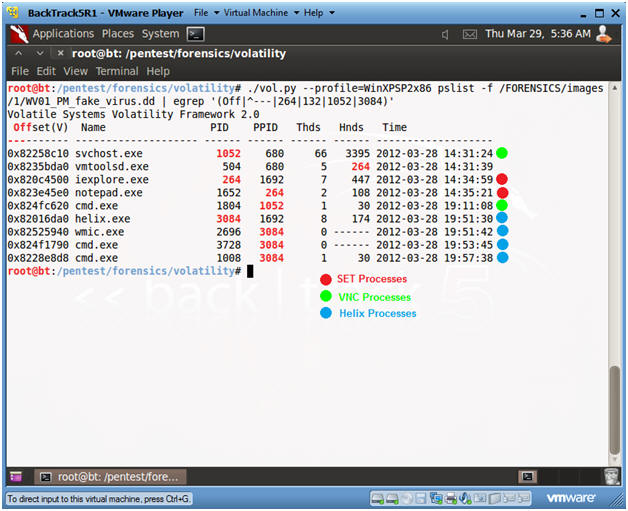
1. View Open Connections with Volatility
   * **Instructions**
     1. **./vol.py --profile=WinXPSP2x86 connections -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd | tee 2>&1 /FORENSICS/images/1/connections.txt**
        + This command both displays connection output on the screen and write the output to a file called connections.
        + ./vol.py --profile=WinXPSP2x86 **connections**
          - Use the WinXPSP2x86 profile and display connections.
          - -f, this flag is used to  specify the image file.
          - tee, means to display on screen and send to an output file.
          - 2>&1, means to place standard error(2) into standard output(1).
     2. **Record Your Remote Address!!!** (See Picture)
     3. Record the Pid's of your SET(264), VNC(1052) and NetCat(132) Process.
   * **Notes(FYI):**
     1. Port 8888 was used by NetCat when we dumped both Physical Memory and the Hard Disk to the BackTrack Server.
     2. Port 3333 and 8080 were created by the SET exploit.
     3. Port 54663 was used by the VNC session.



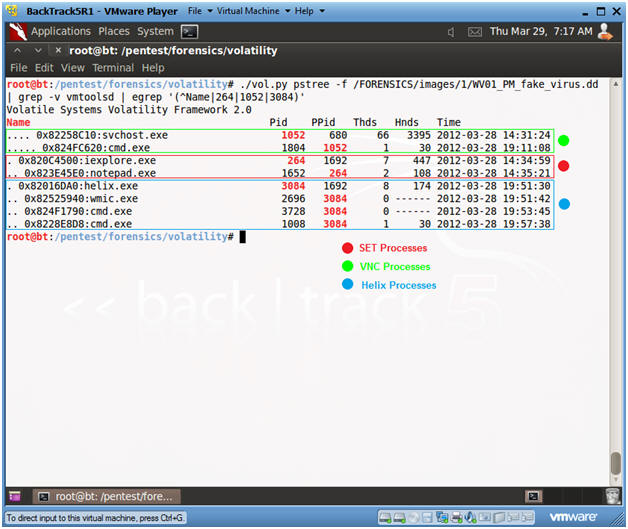
1. View Open and Dead Connections with Volatility
   * **Instructions:**
     1. ./vol.py **connscan** -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd
   * **Notes(FYI):**
     1. "connscan" shows both open and closed connections.
     2. The same SET, VNC and Helix process IDs.
     3. In addition, see the dead connections that are still listed from past Google Web Browsing.



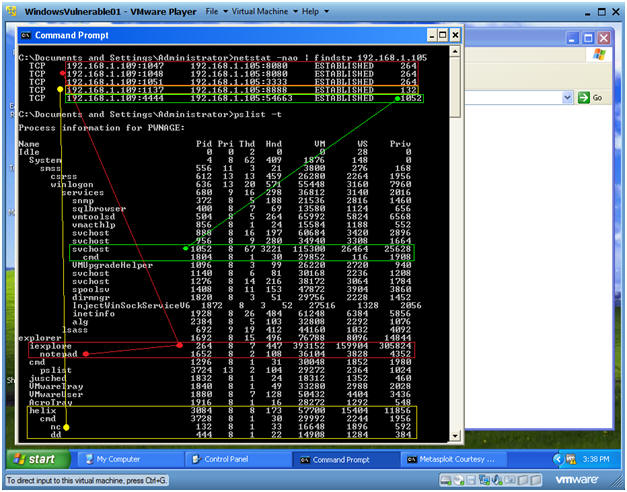
1. View Processes with Volatility
   * **Instructions:**
     1. ./vol.py --profile=WinXPSP2x86 pslist -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd | egrep '(Off|^---|**264|132|1052|3084**)'
        + Replace 264 with your SET Process ID
        + Replace 132 with your Netcat Process ID.
        + Replace 1052 with your VNC Process ID.
        + Replace 3084 with your Helix Process ID.
     2. ./vol.py --profile=WinXPSP2x86 pslist -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd > /FORENSICS/images/1/pslist.txt
        + Display the results to an output file called pslist.txt
   * **Notes(FYI):**
     1. SET Processes are associated with the PID of 264.
     2. VNC Processes are associated with the PID of 1052
     3. Helix Processes are associated with the PID of 3084



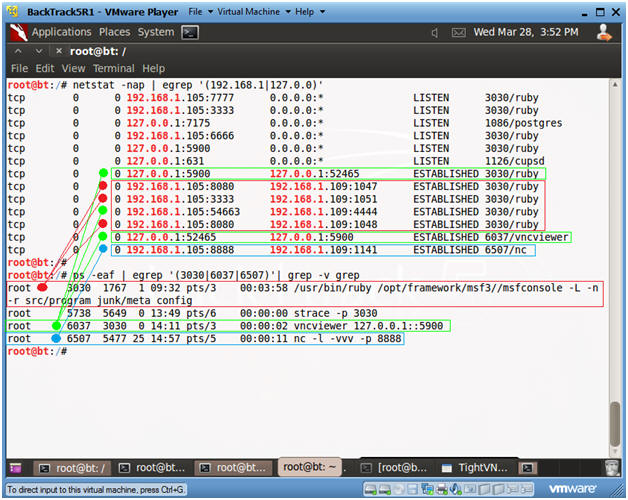
1. View Open Connections with Volatility
   * **Instructions:**
     1. **./vol.py pstree -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd | grep -v vmtoolsd | egrep '(^Name|264|1052|3084)' | tee 2>&1 /FORENSICS/images/1/pstree.txt**
        + PID 264  - SET Processes
        + PID 1052 - VNC Processes
        + PID 3084 - Helix Processes
        + tee 2>&1, display result to output file "/FORENSICS/images/1/pstree.txt".



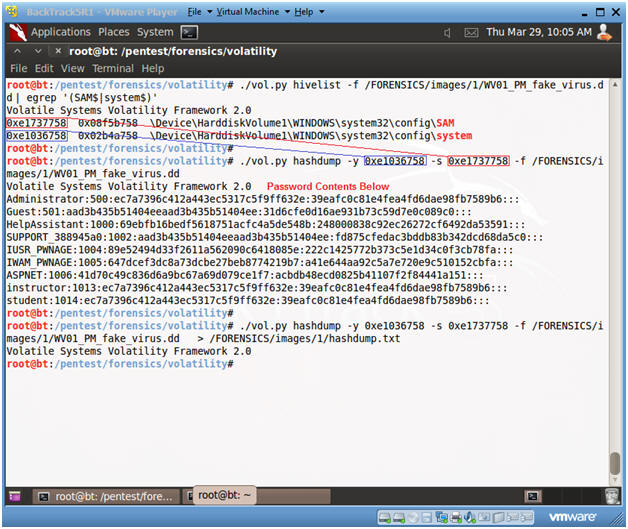
1. Comparing Volatility to netstat and pslist on the Victim Machine
   * **Note(FYI):**
     1. The below analysis will be conducted on WindowsVulnerable01.
     2. Replace **192.168.1.105** with the Remote Address obtained in (Section 3, Step 4).
     3. In addition to gathering a live image, it is always a good idea to run system commands on the victim's machine to have a second source of validation.
     4. **netstat -nao | findstr 192.168.1.105**
        + In my case, 192.168.1.105 was the IP Address of the BackTrack Attacker.
        + Process **264** aligns with the previous **volatility** results that was associated with the **SET processes**.
        + Process **1052** aligns with the previous **volatility** results that was associated with the **VNC processes**.
        + Process **3084** aligns with the previous **volatility** results that was associated with the**Helix processes**.
     5. **pslist -t**
        + Digging further notice PID 264 aligns with the iexplore process, and its' child PID is the mysterious notepad process.



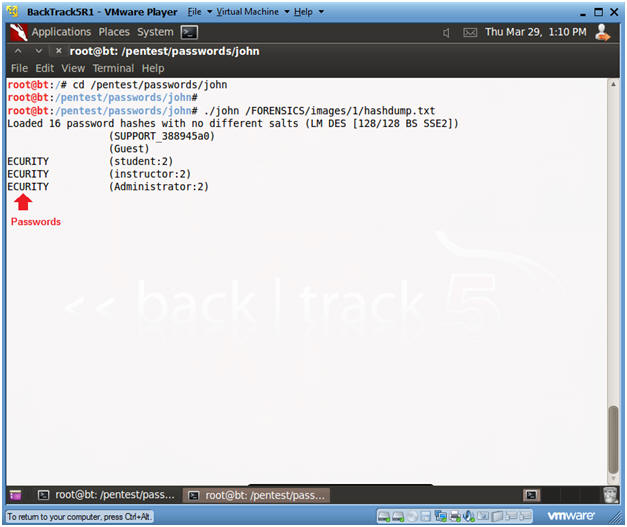
1. Let's Analyze what the Attacker's Network Connections
   * **Note(FYI):**
     1. netstat -nap | egrep '(192.168.1|127.0.0)'
        + Port 5900 and 54663 are used by the VNC Processes
        + Port 8080 and 3333 are used by the SET Processes
        + Port 8888 is the by the Helix Process.
     2. ps -eaf | egrep '(3030|6037|6507)' | grep -v grep
        + PID 3030 is the Metasploit Framework that controls SET.
        + PID 6037 is the vncviewer that controls VNC.
        + PID 6507 is the NetCat listener process waiting for the image.



1. Using Volatilities' hivelist and hashdump
   * **Instructions:**
     1. ./vol.py hivelist -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd | egrep '(SAM$|system$)'
        + Record the memory address in the first column for SAM.
        + Record the memory address in the first column for system.
     2. ./vol.py hashdump -y **0xe1036758** -s **0xe1737758** -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd
        + After the -y, specific the system memory address you recorded.
        + After the -s, specific the SAM memory address you recorded.
     3. ./vol.py hashdump -y **0xe1036758** -s **0xe1737758** -f /FORENSICS/images/1/WV01\_PM\_fake\_virus.dd > /FORENSICS/images/1/hashdump.txt
        + Place output in a file called hashdump.txt.



1. Using John the Ripper to crack the hashdump
   * **Instructions:**
     1. cd /pentest/passwords/john
     2. ./john /FORENSICS/images/1/hashdump.txt
   * **Note(FYI):**
     1. **This could takes hours.  Continue to the Proof of Lab section**.



**Section 4. Proof of Lab**

1. Proof of Lab
   * **Instructions**
     1. Open a new terminal (See Section 2, Step 1)
     2. cd /FORENSICS/images/1/
     3. ls -lrta \*.txt
     4. md5sum WV01\_PM\_fake\_virus.dd
     5. date
     6. echo "Your Name"
        + Replace the string "Your Name" with your actual name.
        + e.g., echo "John Gray"
   * **Proof Of Lab Instructions:**
     1. Do a PrtScn of the below commands
     2. Paste into a word document
     3. Upload to Moodle

