Building a Virtual Enterprise Network Environment for APT Experimentation

Kaila M. Perry, George Hsieh, Cody Butler
Norfolk State University
Norfolk, VA, USA
k.m.perry@spartans.nsu.edu, ghsieh@nsu.edu,
c.butler74291@spartans.nsu.edu

SeanMichael Yurko Galvin, Kevin S. Nauer
Sandia National Laboratories
Albuquerque, NM, USA
sygalvi@sandia.gov, ksnauer@sandia.gov

***Abstract*—Advanced Persistent Threat (APT) represents a class of the most sophisticated, targeted, stealthy, and potentially devastating cyber-attacks. Detecting these threats has proven to be very difficult due to the potency of these attacks and the general lack of defensive capabilities by the victims. Going forward, this asymmetric situation is expected to get worse as the attacking tools can be more readily acquired, modified, and enhanced to not only proliferate more broadly but also become more lethal. To reverse this trend, cyber defenders will need not only better tactics, techniques and procedures, but also a workforce better trained in preventing, defending against, detecting, responding to, and mitigating the effects of APTs. Given the nature of the APTs, training cyber defenders will not be an easy task. An effective learning environment must be provided for the learners to gain hands-on knowledge, skills, and experience. This paper presents the design and implementation of our first-iteration virtual lab environment for learning and experimenting with those APTs that are commonly used to attack enterprise networks. This virtual environment is designed to operate on the XenServer virtualized infrastructure, and composed of a number of building blocks (e.g., router/firewall, Microsoft Domain Controller, web and email servers, Kali Linux), as well as components for the Gh0st malware which is chosen as a case study of the APTs as it has been widely employed worldwide to take full control remotely of vulnerable Windows platforms.**

Keywords—Advanced persistent threat; cyber kill chain; Gh0st; malware; XenServer

I. INTRODUCTION

Advanced Persistent Threat (APT) is a term, believed to be first coined in 2006, for a set of stealthy and continuous cyber-attack processes, often orchestrated by a nation state or cybercrime syndicate, with significant resources and technical sophistication, against a specific high-value target. These threats use multiple attack techniques and vectors, and are conducted by stealth to avoid detection so that attackers can retain control over target systems unnoticed for long periods of time [1]-[4].

As a result, traditional defenses aimed at keeping known threats out of the network are no longer sufficient against such attacks, and the mean time between compromise and detection is 146 days on average globally according to a FireEye report released in 2016 [5]. There have been many well-publicized APT attacks in the past few years, including Stuxnet and attacks against the Target Corp. and Office of Personnel Management, just to name a few.

To counter these advanced threats, the cyber defenders must understand the TTP (Tactics, Techniques and Procedures) used for these insidious exploits. However, it is very challenging to build up one's expertise in this subject area due to the high degree of complexity, variety, and technical sophistication of APTs. Using hands-on labs for learning and experimentation is needed to help cybersecurity learner gain deeper understanding of how the APTs work.

This paper describes the design and implementation of the first-iteration virtual lab environment for learning and experimenting with those APT that are commonly used to attack enterprise networks. It was developed primarily through a M.S. Thesis research [6] at Norfolk State University (NSU) with technical assistance from Sandia National Laboratories (SNL).

This virtual environment is designed to operate on the XenServer virtualized infrastructure, and composed of a number of building blocks including router/firewall, Microsoft Domain Controller, web and email servers, and Kali Linux for penetration testing. The choices for the virtualization platform, router/firewall, and email server are derived from the Tracer FIRE cyberforensic training platform developed by SNL [7].

For this implementation, the Gh0st Remote Access Trojan (RAT) is chosen as an APT case study due to its prevalence in the wild, potential to cause devastating damages to the victims, and use of a variety of techniques to evade detection [8] [9]. Note that a similar Gh0st malware experimentation environment has been developed at NSU for desktop or laptop computers, utilizing VMware Workstation hypervisor, to support individual learners [10] [11]. The environment described in this paper is intended to run on larger scale virtualized or cloud infrastructures, and designed to be extensible such that additional components and capabilities can be integrated into the implementation in an incremental and iterative manner.

The paper is organized as follows. In Section II, we will discuss the motivation and objectives for the APT experimentation environment. In Section III, we will describe the design and system configuration of our first-iteration implementation of this environment. In Section IV, we will describe the steps to setup and configure the virtual environment. In Section V, we will discuss the operation of the Gh0st malware case study. Then we conclude the paper with a summary, a discussion on our lessons learned, and future plans.

This work was supported in part by U.S. Department of Defense (award #FA8750-15-2-0120), and U.S. Department of Energy/NNSA (award #DE-NAOO02686).

II. Motivation and Objectives

A. Motivation

For many people, the best way to really learn something is by actually doing it. The motivation for this research came primarily from the Tracer FIRE (Forensic Incident Response Exercise) platform and the many training events that utilized the platform and various training scenarios.

The forensic exercises in Tracer FIRE were originally designed for incident responders and analysts in government agencies [5]. Throughout the years, the SNL team has made great efforts in extending an educational and training capability for the platform. As a result, Tracer FIRE is now often used for training high school and college students as well as cybersecurity practitioners.

The Tracer FIRE platform and its programmatic approach offer several key strengths:

1) It has a broad but well-defined learning objective which is focused on host forensic analysis.
2) It provides a very comprehensive set of forensic analysis tools that are accessible in one virtual environment.
3) It provides a feature-rich, scalable and extensible platform upon which new capabilities and training scenarios can be readily developed and deployed to stay current with new attacks observed in the wild.
4) The training scenarios can be customized for different audiences with varying degrees of knowledge ranging from beginners to experts.

On the other hand, the Tracer FIRE platform and its programmatic approach have several limitations in their current implementation:

1) It is not designed as a continuous training environment, since its most suitable delivery mode is pre-scheduled exercise or competition events.
2) It is not designed as a self-paced learning environment, since the participants are usually put through the same pace and training scenarios at each event.
3) It is not easy to scale up the number and size of the training events given the costs involved in personnel, travel, equipment, etc.
4) It can be quite challenging for participants to grasp the concepts and learn the skills as intended at the event, given the time constraints, especially if they do not have sufficient background knowledge or familiarity with the tools.

Overall, the design and implementation of our virtual lab environment - for learning and experimenting with the APTs that are commonly used to attack enterprise networks - are motivated and influenced by the lessons learned from the Tracer FIRE platform and its technical and programmatic approaches.

There is another important factor that motivates and informs our research and development. This is based on our observation that the traditional cybersecurity training approach is not very effective in dealing with APTs which generally have a higher level of complexity and technical sophistication, broader scope and longer timespan of compromises.

There are many cybersecurity courses offered by various academic institutions through former educational curricula or degree programs. There are also many cybersecurity training courses, some over a very short period of time (e.g., days), provided by commercial companies or training organizations, for professionals or people seeking cybersecurity certifications at varying levels of expertise.

The training or educational courses are generally organized based on job categories or certification specializations. For example, different courses are offered for penetration testing, intrusion detection, forensic analysis, incident response, etc. While they can be very beneficial, especially in building up one's expertise in the selected subject areas, this stovepipe approach does not provide a comprehensive and cohesive coverage of the entire lifecycle of APTs which is best described by a cyber kill chain [12].

To overcome this deficiency, we set out to use the cyber kill chain concept as the organizing principle for our research and development of the APT virtual lab environment that we hope is more conducive to meeting the particular needs of APT training and experimentation.

B. Objectives

To guide our research and development efforts, we have established the following long-term key objectives for the enterprise APT experimentation environment.

1) Focus on enterprise APTs.
2) Provide an enterprise-like environment that is sufficiently realistic to represent APT kill chains, but compact enough so the learner can concentrate on the key concepts.
3) Build on enterprise level virtualization or cloud computing infrastructures to support scalability in both capabilities and capacities.
4) Use common systems, software and tools as much as possible across the entire environment to help reduce the learning curves.
5) Use open source software and tools as much as possible to help reduce the ownership cost.
6) Support self-paced and continuous learning.
7) Support multiple scenarios (or learning modules) on the same platform.

III. System Design and Configuration

A. System Design

In this section, we will describe the major components of the first-iteration enterprise APT experimentation environment.

Virtualization infrastructure. XenServer, Version 6.5, is used as the virtualization platform for the virtual machines needed to create the enterprise network. It is a "complete virtualization platform, optimized for both Windows and Linux virtual

servers, with all the capabilities required to create and manage a virtual infrastructure" [13]. Similar to VMware's ESXi, XenServer is a Type 1 hypervisor that runs directly on the hardware of the host to provide an efficient and scalable platform. Unlike VMware, it is an open source product supported by Citrix. XenServer is also used by Sandia's Tracer FIRE system.

In conjunction with XenServer, XenCenter is used to manage the virtualization infrastructure and the virtual machines running on it. XenCenter is a "graphical, Windows-based user interface" that allows users to "deploy, manage, and monitor" the virtual machines [13]. For the APT experimentation environment, the XenCenter application is installed and operated on a dedicated desktop computer running Microsoft Windows 7 OS.

Router/firewall. The first-iteration enterprise APT learning and experimentation environment is designed to place the VMs in three separate networks: EvilNet (attacker), GullibleNet (third-party), and EnterpriseNet (internal). Routing and access control among the networks are implemented by a VyOS based router/firewall running in a VM. VyOS is an open source Linux-based network operating system that provides software-based network routing, firewall, and VPN functionality" [14]. Comparing with other open source router/firewall implementation, VyOS is more similar to traditional hardware routers such as Cisco routers. This similarity is a major advantage for VyOS as it is easier to adapt and support VyOS by people who are already familiar with the traditional hardware routers/firewalls. Note that VyOS is also used by Sandia's Tracer FIRE system.

Email server. Spear-phishing has been a very popular and effective method for the bad guys to launch cyber-attacks. Because a license key is needed to use Microsoft Exchange email server and Outlook email client, the open-source iRedMail [15] is chosen as the email server for the APT experimentation environment.

The iRedMail email server can run on many variants of Linux or UNIX operating systems, including Red Hat Enterprise Linux, CentOS, Debian, Ubuntu, FreeBSD, and OpenBSD. It is a full-featured email server providing password security and encryption, antivirus protection and privacy. Both Ubuntu and CentOS were attempted as the operating system for the VM running the iRedMail server, but Ubuntu 14.04.4 was ultimately chosen because it worked properly and readily with the XenServer and network attached storage (NAS) device used in our APT experimentation environment.

HTTP/FTP server. Malicious payload is often downloaded or transferred to the victim's machine from a web or FTP server. An Ubuntu based VM is set up in the APT experimentation environment to run the Apache 2 HTTP Server [16] which is installed and configured as a part of the open source LAMP (Linux, Apache, MySQL, PHP) platform [17]. The VM is also configured to run the vsftpd (very secure FTP daemon) [18] which is a GPL licensed FPT server for UNIX or Linux systems.

Domain controller/DNS server. Most enterprise networks deploy Microsoft Active Directory services running on domain controllers to centrally manage the users and computers in their Microsoft centric software environments, and DNS services to translate domain names to IP addresses.

For the enterprise APT experimentation environment, a VM is set up to run Microsoft Windows Server 2008 R2 operating system with both domain controller and DNS server roles activated and configured.

Penetration testing. A VM is set up to run Kali Linux [19] which is a Debian-derived Linux distribution designed for digital forensics and penetration testing. Kali Linux is preinstalled with over 300 penetration-testing programs, and it is one of the most popular and widely used penetration testing toolkit.

Gh0st malware. There are two main components of a Gh0st RAT system: the client (which is the C2 host), and the server (which is the compromised host) [9]. The server is a small Microsoft Windows DLL that runs as a Windows service and is automatically started when the host starts. Upon startup, it connects to a pre-programmed C2 host and awaits further instructions. The C2 component is a standard Windows application. It provides a GUI to list all the compromised hosts and perform 13+ types of attacks upon the compromised host remotely to access, tamper, eavesdrop, and exfiltrate data, or disrupt its operation.

Similar to the desktop based Gh0st RAT experimentation environment [11], two VMs running Windows 7 Professional (64-bit) OS are configured to run the Gh0st server and Gh0st client, respectively, within the enterprise APT experimentation environment.

B. System Configuration

In this section, we will describe the hardware configuration, physical network configuration, and virtual networks and hosts configuration for the enterprise APT experimentation environment.

Hardware configuration. The experimentation environment is consisted of the following subsystems:

1) Compute: two Dell T710 workgroup servers are used to provide the computing resources. Each server is equipped with 2 Intel Xeon CPUs / 12 cores, 128 GB RAM, and 550 GB hard disk storage. Both T710 servers are installed with XenServer 6.5 hypervisor, and placed in a pool to allow the VMs to run on either host.
2) Storage: one Dell PowerVault NX3100 network attached storage appliance is used to provide 24 TB of shared storage for the VMs. The NAS device runs the Windows Storage Server 2008 Standard Edition operating system which supports multiple file sharing protocols, including NFS and CIFS (Common Internet File System). The device provides two 1 Gbps Ethernet ports, both of which are connected to the Ethernet switch.
3) Control: one Dell T3500 workstation is used to run XenCenter 6.5 application on Windows 7 Professional operating system. The T3500 is equipped with 1 Intel Xeon CPU / 4 cores, 12 GB RAM, and 930 GB hard disk storage.

Physical network configuration. A Cisco Catalyst 2960 Ethernet switch is used to provide layer-2 connectivity among the compute, storage, and control subsystems at 1 Gbps port speed.

Fig. 1 shows the physical network and hardware configuration for the enterprise APT experimentation environment. For simplicity, only one T710 server is included.

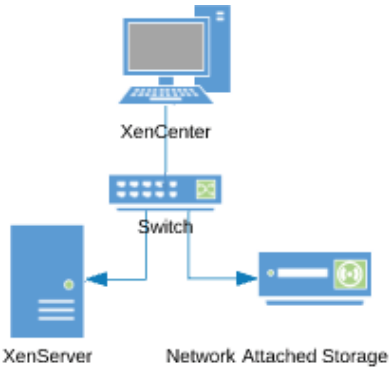

Fig.1. Hardware configuration

Static private IP addresses are assigned to the three subsystems as shown in Table 1.

TABLE 1. HARDWARE CONFIGURATION

Hardware Device	Role	IP Address
Dell T710 Server	Compute: XenServer	192.168.2.101
Dell NX3100 NAS	Storage	192.168.2.50
Dell T3500 Workstation	Control: XenCenter	192.168.2.1

Virtual networks and hosts. The first-iteration virtual environment for enterprise APT experimentation is consisted of three networks and nine virtual machines, as shown in Fig. 2.

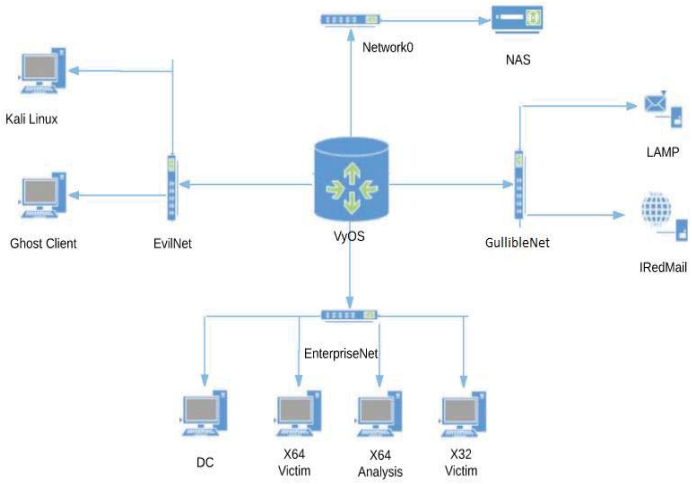

Fig. 2. Virtual networks and hosts

Static private IP addresses are assigned to the three virtual networks as shown in Table 2. Note that the default gateway addresses are assigned to the three (virtual) NICs on the VyOS router, with each NIC connecting to a separate virtual network.

TABLE 2. VIRTUAL NETWORKS

Network	Subnet ID (/24)	Default Gateway Address
EnterpriseNet	192.168.10.0	192.168.10.1
EvilNet	10.10.1.0	10.10.1.1
GullibleNet	172.16.2.0	172.16.2.1

The IP address, software configuration, and role of each of the nine VMs are listed in Table 3.

TABLE 3. VIRTUAL HOST CONFIGURATION

VM	Role	IP Address	Software
VyOS	Router/ firewall		Ubuntu 14.04.4/ VyOS
DC	AD/ DNS	192.168.10.50 / 192.168.10.11	Windows Server 2008 R2
X64 Victim	Gh0st victim	192.168.10.111	Windows 7 (x64)
X32 Victim	Gh0st victim	192.168.10.112	Windows 7 (x32)
X64 Analysis	Forensics	192.168.10.106	Windows 7 (x64)
Kali Linux	Pen testing	10.10.1.7	Kali Linux
Gh0st Client	Gh0st C2	10.10.1.45	Windows 7 (x64)
LAMP	Web server	172.16.2.15	Ubuntu 14.04.4/ LAMP
iRedMail	Email server	172.16.2.109	Ubuntu 14.04.4/ iRedMail

IV. SETUP AND CONFIGURE SYSTEM

In this section, we will provide high-level description of the major steps in setting up and configuring the enterprise APT experimentation environment.

A. *XenServer & XenCenter*

To install and configure XenServer and XenCenter, follow the Citrix XenServer 6.5 Quick Start Guide. Load from a XenServer 6.5 Installation disk onto the Dell T710 server, and follow the step-by-step instructions to configure the XenServer: set the IP address to 192.168.2.101 and root password.

To install XenCenter onto the Dell T3500 workstation, which was already running Windows 7 OS, use the same XenServer installation disk. Execute "XenCenterSetup.exe" in the "client_install" folder, and follow the Setup wizard. Configure the IP address for the workstation to 192.168.2.1.

To connect the XenCenter to the XenServer, besides having the two machines on the same subnet, add a new server to the XenCenter along with the IP address and root credentials of the XenServer. When the XenServer is successfully connected to the XenCenter, it will be shown on XenCenter's dashboard with a green dot, indicating the XenServer is ready for additional virtual machines to be installed and managed on the server.

B. *NAS Storage*

Configure the IP address of the device to 192.168.2.50, and create a NFS share (server) to provide shared storage for VMs running on XenServer.

C. *Image Libraries*

It is most convenient to create VMs from ISO images stored in file shares. To simplify this process, first set up a share on XenCenter: (a) Create a new ISO library folder on XenCenter and use Computer Management to allow sharing using the CIFS

protocol. (b) Mount the share as a storage repository on XenServer by specifying the IP address, share name, and user credentials of XenCenter.

Similarly, create a new CIFS share for an ISO library on NAS, and mount this share as storage repository on XenServer by specifying pertinent NAS information as shown in Fig. 3.

Fig. 3. Mount NAS share on XenServer

D. VyOS VM

Create a new VM to run Ubuntu 14.04.4. Install VyOS. Configure VyOS with four network interfaces, as shown in Table 4, and enable and configure DHCP service.

TABLE 4. HARDWARE CONFIGURATION

Interface	Network	Address
eth0	NAS	192.168.2.1
eth1	EnterpriseNet	192.168.10.1
eth2	EvilNet	10.10.1.1
eth3	GullibleNet	172.16.2.1

E. DC VM

Create a new VM to run Windows Server 2008 R2. Activate the DC and DNS Services. Configure the IP address for the DC to 192.168.10.50 and the preferred DNS to 192.168.10.11. Set the domain name to (factitious) dc.lee.com.

F. iRedMail VM

Create a new VM to run Ubuntu 14.04.4. Install iRedMail. Configure the email server's IP address to 172.16.2.109, and domain name to remail.lee.com. Add two more domains under the server's control: dc.lee.com and support.lee.com, for users and technical support of the organization, respectively.

Create two email user accounts: student@dc.lee.com and tech@support.lee.com, to represent a victim (student) and an attacker (impersonating a tech support person), respectively.

G. LAMP VM

Create a new VM to run Ubuntu 14.04.4. Install LAMP. Configure the web server's IP address to 172.16.2.15. Install vsftpd FTP server and configure its address to 172.16.2.15.

H. Gh0st VMs

Create two new VMs to run Windows 7 (x64). Install Gh0st client software on the "Gh0st Client" VM and configure its IP address to 10.10.1.45. Configure the IP address for the "X64 Victim" VM to 192.168.10.111. Further detail for setting up and configuring the Gh0st VMs can be found in [11].

V. EXECUTE SIMULATED GH0ST ATTACK

The simulated Gh0st attack is summarized by the following phases of a cyber kill chain, similar to the one described in [11]:

1) Weaponization. On the "Gh0st Client" VM, Attacker uses gh0st.exe to generate a (malicious) executable named "Putty.exe" which is embedded with Attacker VM's IP address (10.10.1.45) and port number (80), as shown in Fig. 4. The file name and port number are intentionally chosen to trick victims into thinking they are legitimate. A legitimate sounding name "Microsoft Devices Managers" is also chosen for the service display of the persistent RAT once it is installed on Victim's system.

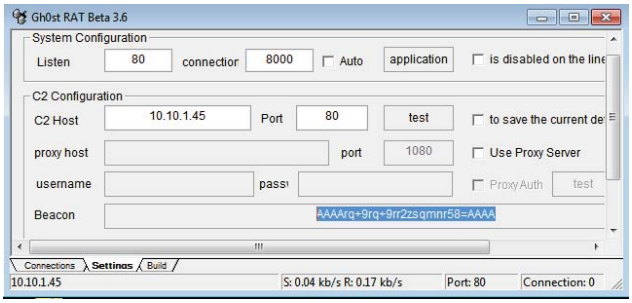

Fig. 4. Create malicious payload

2) Set up C2. On the "Gh0st Client" VM, Attacker keeps gh0st.exe running and waits for compromised hosts to report in.

3) Set up delivery method. From the "Gh0st Client" VM, Attacker uploads the malicious Putty.exe file to the LAMP VM (172.16.2.15). Next, Attacker composes and sends a phishing email to Victim (student@dc.lee.com) disguised as coming from tech support (tech@support.lee.com). The phishing email instructs Victim to download and install an attached (malicious) Putty.exe file, as shown in Fig. 5, or click an embedded link to an FTP site to download and install (malicious) Putty.exe.

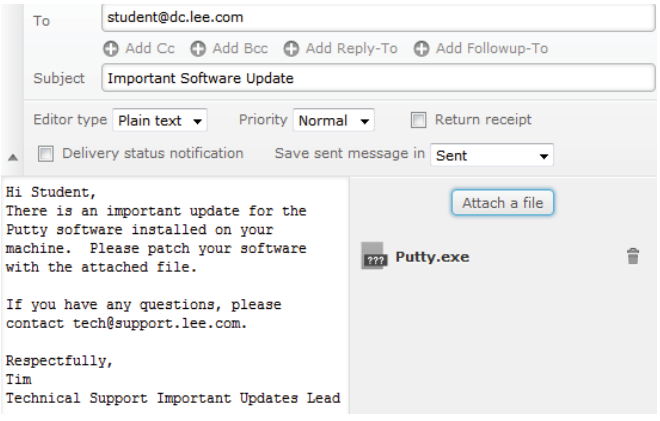

Fig. 5. Phishing email

4) Delivery. On the "X64 Victim" VM, Victim (student@dc.lee.com) receives the phishing email, and clicks on the attached file or embedded link to save the (malicious) Putty.exe on the victim machine.

5) Installation. On the "X64 Victim" VM, Victim runs the saved Putty.exe file as administrator which causes the malicious payload to be installed on the victim machine.

6) Establish C2 channel. After the malicious payload is installed on "X64 Victim" VM, it sends a beacon message to the "Gh0st Client" VM to establish a C2 channel. When completed, a message will be displayed on the GUI for the gh0st.exe app to show the established C2 connection to "X64 Victim" VM (192.168.10.111), as shown in Fig. 6.

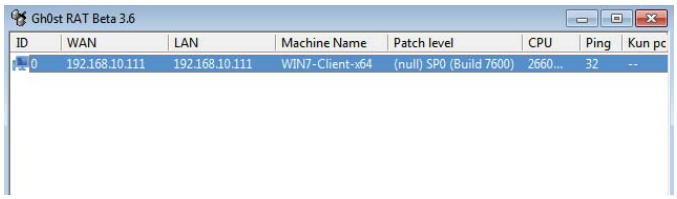

Fig. 6. Gh0st C2 channel established

7) Actions on objectives. Attacker uses the gh0st.exe app on "Gh0st Client" VM to remotely take control of the victim machine and execute eleven different operations such as keylogging and file manipulation, as shown in Fig. 7.

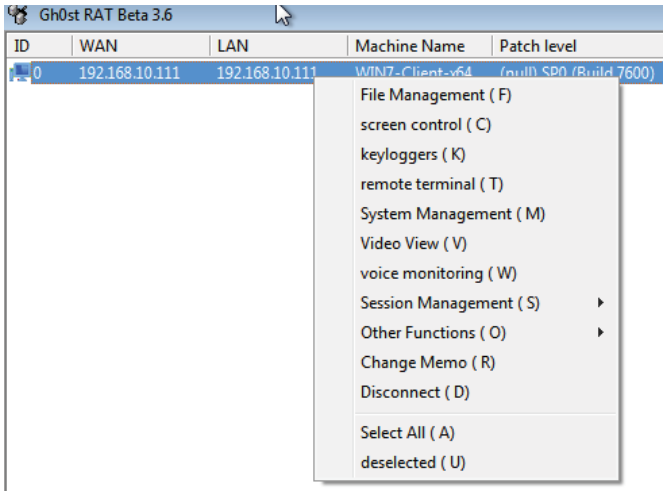

Fig. 7. Gh0st RAT capabilities

VI. SUMMARY

Advanced persistent threat has been a serious threat to cybersecurity since 2000s and will remain so for many years to come. Its scope, technical sophistication, and TTP are constantly evolving and expanding, with increasing investments by more bad actors worldwide especially crime syndicates and nation states. This presents a formidable challenge for cyber defenders to stay abreast of the APT landscape and develop effective detection, analysis, and mitigation techniques.

To help students and practitioners in the cybersecurity field gain in-depth knowledge and skills in how APT works, it is both necessary and beneficial to provide virtual learning and experimentation environments so they can perform hands-on exercises in a safe environment.

This paper presents the implementation of our first-iteration virtual lab environment for learning and experimentation of enterprise APTs such as Gh0st RAT. This virtual lab environment is designed to emulate key capabilities and vulnerabilities in enterprise networks in a compact but realistic manner. It is also designed to be scalable and extensible to support future needs in both capabilities and capacities.

This research and development effort has provided the following major observation, experience, and lessons learned:

1) Hands-on and experiential learning is very beneficial for cybersecurity training, especially in dealing with complex topics such as APTs.

2) Designing and implementing a virtual lab environment to facilitate hands-on learning of APTs provide even greater benefits in gaining more in-depth knowledge and practical and broad-based skills.

3) Using cyber kill chain as an organizing principle helped significantly to establish the initial focus of the design and implementation.

4) Selecting a real-world APT such as Gh0st RAT as a case study helped significantly to define the initial scope of the design and implementation.

5) Although XenServer was a good choice for the Tracer FIRE platform a few years ago, it is not a full-fledged cloud platform. To support multiple user communities and different APT case studies, a cloud platform such as OpenStack [20] is desirable for the virtual lab environment.

Going forward, we plan to continue enhancing and expanding the virtual lab environment:

1) Migrate to OpenStack which is the most popular and widely used open-source cloud operating system.

2) Incorporate more varieties of enterprise APTs such as Poison Ivy.

3) Incorporate incident response tools to assist detection and forensic analysis of enterprise APTs.

4) Incorporate malware detection, analysis and reverse engineering tools and capabilities to help users learn and to defend against the APTs.

We also plan to continue using this virtual lab environment and its enhanced versions to support our research, education, and workforce development efforts. Furthermore, we plan to assess the efficacy of this type of virtual learning and experimentation environment and associated programmatic approach for enterprise APTs via student research projects, courses, workshops and outreach activities.

REFERENCES

[1] FireEye, APT1: Exposing One of China's Cyber Espionage Units, Oct 25, 2004, https://www.fireeye.com/content/dam/fireeye-www/services/pdfs/mandiant-apt1-report.pdf [accessed April 29, 2017].

[2] E. Cole, *Advanced persistent threat: understanding the danger and how to protect your organization*. Newnes, Dec. 2012.

[3] A.K. Sood, and R.J. Enbody, "Targeted cyberattacks: a superset of advanced persistent threats." *IEEE security & privacy*, 2013(1), pp. 54-61, Jan. 2013.

[4] A.K. Sood, and R.J. Enbody, *Targeted Cyber Attacks: Multi-staged Attacks Driven by Exploits and Malware*. Syngress, 2014.

[5] FireEye, M-TRENDS EMEA: Lessons learned from data breaches in 2015, June 2016, https://www2.fireeye.com/WEB-RPT-M-Trends-2016-EMEA.html [accessed March 6, 2017].

[6] K. Perry, An Integrated Cyber Security Learning and Experimentation Environment, M.S. Thesis, Computer Science Department, Norfolk State University, Norfolk, VA, July 2016.

[7] B. Anderson, K. Nauer, W. Lee, J.T. McClain and R. Abbott, Tracer FIRE Cyberforensic Training Platform, Sandia National Laboratories, SAND2015-3374C, 2015, https://www.osti.gov/scitech/biblio/1251138 [accessed April 29, 2017].

[8] Information Warfare Monitor, Tracking GhostNet: Investigating a Cyber Espionage Network, March 2009, http://www.nartv.org/mirror/ghostnet.pdf [accessed March 6, 2017].

[9] M. Spohn, Know Your Digital Enemy: Anatomy of a Gh0st RAT, whitepaper, McAfee, April 2012. http://docplayer.net/176479-Know-your-digital-enemy.html [accessed March 6, 2017].

[10] V. Boinapally, Developing a Simulated Attacker Framework and Testbed for Cybersecurity Learning and Experimentation, M.S. Project, Computer Science Department, Norfolk State University, Norfolk, VA, Dec. 2016.

[11] V. Boinappally, G. Hsieh and K. Nauer, Building a Gh0st Malware Experimentation Environment, to appear in the Proceedings for the SAM 2017 Conference, July 2017, Las Vegas, NV.

[12] E. Hutchins, M. Cloppert , R. Amin, Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains, whitepaper, Lockheed Martin Corp., Nov. 2011, http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf [accessed March 7, 2017].

[13] Citrix Product Documentation. XenServer 6.5. https://docs.citrix.com/en- us/xenserver/xenserver-65.html [accessed May 3, 2017].

[14] VyOS. https://vyos.io/ [accessed May 3, 2017].

[15] iRedMail - Free, Open Source Mail Server Solution. http://www.iredmail.org [accessed May 3, 2017].

[16] Apache HTTP Server Project. https://httpd.apache.org [accessed May 3, 2017].

[17] How to Install LAMP on Ubuntu. http://howtoubuntu.org/how-to-install-lamp-on-ubuntu [accessed May 3, 2017].

[18] vsftpd - Probably the most secure and fastest FTP server for UNIX-like systems. https://security.appspot.com/vsftpd.html [accessed May 3, 2017].

[19] KALI by Offensive Security. https://www.kali.org [accessed May 3, 2017].

[20] OpenStack. https://www.openstack.org [accessed May 7, 2017].