Malware Lab PPPs v.1

October 1, 2015

Threat Research Lab

This document details the policies, procedures, and protocols for automated malware analysis in accordance with the recommendations in NIST Special Publication 800-83: Guide to Malware Incident Prevention and Handling, NIST Special Publication 800-53: Recommended Security Controls for Federal Information Systems and Organizations, and NIST Special Publication 800-61: Computer Security Incident Handling Guide.

Jody J. Forness
Senior Security Researcher
E: Jody.Forness@baesystems.com
T: +1 (646) 224 7724

Table of Contents

1 Purpose

The Threat Research Lab needs an environment to conduct automated analysis of malicious samples, in order to extract metadata, identify threat actors, and associate campaigns. Ultimately, we provide an internal intelligence feed back to the Security Operations Center for network correlation/analysis, while also augmenting the intelligence capabilities of the threat intelligence team. The end goal is to advance our offering and provide customers with comprehensive threat protection.

2 Authority

The Threat Research Lab is an outstanding team of ethical hackers with over 35 years of combined experience in application and Web security, vulnerability research, incident response, digital forensics, malware reverse engineering, machine learning, data analytics, and product development, working under the supervision of Brandon Edwards, Vice President of the Threat Research Lab.

Brandon Edwards is "recognized as one of the top vulnerability researchers in the world"[1] with ten years of experience in reverse engineering, exploit development, and cyber security research. Mr. Edwards co-founded Exodus Intelligence, providers of zero-day vulnerability intelligence. He's taught as an adjunct lecturer at NYU Polytechnic School of Engineering, presented at numerous industry conferences, and frequently judges CTF[2] competitions. Most recently, Mr. Edwards appeared in WIRED for his research on VoIP telephones[3].

Senior Security Researcher, Jody J. Forness, holds a Master of Forensic Sciences in High-Technology Crime Investigation from The George Washington University. Previously, Ms. Forness worked as a Digital Forensic Examiner on high-profile cases for Stroz Friedberg and as a Network Security Analyst for Northrop Grumman on a contract for the Department of the Treasury where she investigated over 120 computer intrusion incidents and analyzed more than 250 files for malware. Ms. Forness completed Department of Homeland Security, US-CERT Technical Mentoring Modules in Malware Handling and Storage, Malware Obfuscation, and Incident Handling Methodology.

3 Philosophy

The current philosophy in the cyber security industry is "defense-in-depth", which typically means layers of security to mitigate the risk that one or more layers may fail; however, this results in *more software and hardware*, on top of layers and layers of user applications, remote administration utilities, and "security tools", merely adding to the already abundant attack surface. The question no one is asking: Who wrote these programs?

In 2010, 40,000 students graduated with Computer Science degrees in the U.S.[4], educated in functional, fast, concise, well-documented software but NOT *secure software*. In 2011, 15,000 students graduated with Electrical/Computer Engineering degrees, educated in logic, design, and basic firmware programming[5]. As the cost of logic analyzers and chip-level interfaces decline, the easier it is to attack and persist in low-level hardware components, whose security is frequently overlooked in the design, sale, and deployment of these devices.

---

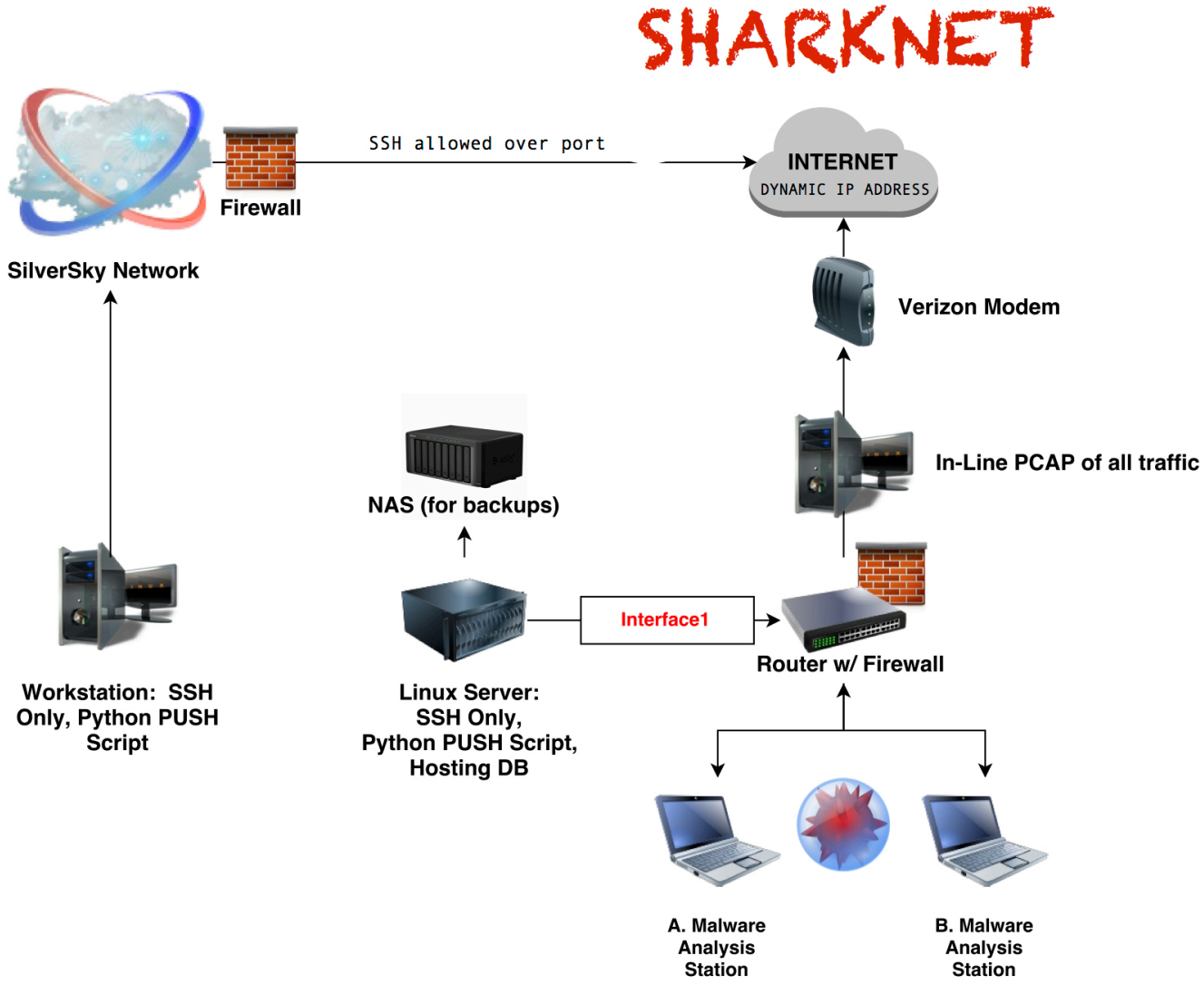
[1] http://www.silversky.com/news-and-events/press-releases/SilverSky-Announces-Expanded-Lab-Capabilities-Hire-of-Threat-Prevention-Expert-Brandon-Edwards
[2] Capture the flag competitions simulate real-life computer security attack/defend challenges, providing a safe environment to improve one's skillset.
[3] http://www.wired.com/2014/06/desk-phone-hacks
[4] http://www.geekwire.com/2014/analysis-examining-computer-science-education-explosion
[5] http://www.asee.org/papers-and-publications/publications/college-profiles/2011-profile-engineering-statistics.pdf

Our “less-is-more” philosophy and preference for in-house tooling stems from extensive industry experience that reiterates the truly small role that security has played in the software and hardware development process and the dangers of assumed security. In accordance with Threat Research Lab policy, all hardware equipment and supporting software undergoes a thorough audit to assess the additional attack surface and to analyze the risks to the overall network. Through source code analysis, it’s possible to understand the coding habits, pitfalls, shortcuts, and temperaments of individual developers, common types of vulnerabilities, and in short, the overarching environment that makes exploitation possible.

- ✓ More software means *more* attack surface.
- ✓ Open-source software means *known* attack surface.
- ✓ Closed-source software represents *unknown, exploitable* attack surface, which is potentially worse.

The design for the Malware Lab incorporates the “less-is-more” philosophy, balancing industry-standard policies with practical, workable procedures, utilizing protocols for isolation, automation, and risk mitigation.

4 Design



The Malware Lab is an isolated analysis network that connects to the Internet via non-attributable DSL. The network consists of (1) a Linux server, (2) Verizon DSL, (3) router/firewall, (4) malware analysis workstations, (5) full packet capture and (6) an internal workstation. Data flows outbound-only to mitigate the risk from potentially untrusted resources. All communications will utilize SSH over port XXXX. Samples identified are transferred from internal production servers to an intermediate, internal workstation, and from there, outbound to Sharknet. A proprietary rest API will submit samples for analysis and retrieve the sanitized results.

4.1 Linux Server

The Linux server serves as the primary storage location for malicious samples, analysis results, and intelligence information. The server is running a custom Linux kernel implementing limited package support, compiled with all available anti-exploitation/hardening mechanisms (STACKPROTECTOR, PIE, BINDNOW, RELRO) and with

GRSEC[6] patches incorporating full ASLR/NX[7], PaX[8], and strict AppArmor[9] rules. Currently, the host-based iptables firewall allows inbound SSH connections over port XXXX, and drops all remaining inbound connections.

In an effort to restrict administrator-level privileges by users, there is no root user account on the server. There are individual accounts and each user authenticates via SSH key *and* password.

The server hosts several databases containing source metadata, encrypted samples, DNS information, and analysis results. Supporting software, including graph/relational databases, container environments, and tooling, was carefully chosen to limit additional, known attack surface.

A custom rest API will PUSH new samples to individual malware workstations for automated analysis and will retrieve the sanitized analysis results.

4.2 DSL

The Malware Lab utilizes a non-attributable DSL from Verizon and a Verizon-provided Actiontec modem. All unnecessary services on the modem are disabled including Wi-Fi and remote administration. Default passwords were changed; however, our initial research indicates that the admin password is stored as base-64 encoded plain text in an .xml on the device, which is accessible via local telnet. We are reverse engineering the firmware to determine its susceptibility to attack from malware within the network and to identify other potential vectors of attack. Research is ongoing.

```
LOAD:0041AF34  # ============== S U B R O U T I N E =======================================
LOAD:0041AF34
LOAD:0041AF34
LOAD:0041AF34                  .globl proto_Readline
LOAD:0041AF34 proto_Readline:                          # CODE XREF: sub_40D64C+58↑p
LOAD:0041AF34                                          # sub_40D64C+37C↑p ...
LOAD:0041AF34
LOAD:0041AF34                  li      $gp, 0x3FB5C
LOAD:0041AF3C                  addu    $gp, $t9
LOAD:0041AF40                  addiu   $sp, -0x48
LOAD:0041AF44                  sw      $ra, 0x48+var_4($sp)
LOAD:0041AF48                  sw      $fp, 0x48+var_8($sp)
LOAD:0041AF4C                  sw      $s0, 0x48+var_C($sp)
LOAD:0041AF50                  move    $fp, $sp
LOAD:0041AF54                  sw      $gp, 0x48+var_30($sp)
LOAD:0041AF58                  sw      $a0, 0x48+arg_0($fp)
LOAD:0041AF5C                  sw      $a1, 0x48+arg_4($fp)
LOAD:0041AF60                  sw      $a2, 0x48+arg_8($fp)
LOAD:0041AF64                  lw      $v0, 0x48+arg_4($fp)
LOAD:0041AF68                  sw      $v0, 0x48+var_20($fp)
LOAD:0041AF6C                  lw      $v0, 0x48+arg_0($fp)
LOAD:0041AF70                  lw      $v0, 4($v0)
LOAD:0041AF74                  move    $a0, $v0
LOAD:0041AF78                  li      $a1, 3
LOAD:0041AF7C                  la      $v0, fcntl
LOAD:0041AF80                  move    $t9, $v0
LOAD:0041AF84                  jalr    $t9 ; fcntl
```

Figure 1. Excerpt from disassembly of Verizon Wireless DSL Gateway firmware.

[6] Grsecurity is a set of Linux kernel patches that incorporate memory corruption defenses, filesystem hardening, enhanced role-based access control, and gcc plugins that make the system a hostile environment for attackers (http://www.grsecurity.net).
[7] Address Space Layout Randomization randomly arranges the address space positions of key data areas of a process, including the base of the executable and the positions of the stack, heap, and libraries (http://www.wikipedia.org/wiki/Address_space_layout_randomization). The NX bit allows the operating system to mark certain areas of memory as non-executable (http://www.wikipedia.org/wiki/NX_bit).
[8] PaX implements least privilege protections for memory pages (http://www.wikipedia.org/wiki/PaX).
[9] AppArmor is a kernel enhancement to bind access control attributes to programs rather than to users (http://www.wiki.ubuntu.com/AppArmor).

4.3 Router/Firewall

The FortiNet FortiGate 110C routes traffic between individual subnets and drops all inbound connections except SSH over port XXXX. The firewall will allow all outbound and established/related connections. Additionally, remote management interfaces and non-essential features are disabled.

Our security audit of this device indicates that the command-line interface is well abstracted, and individual commands are stripped of useful functionality. We were able to read the firmware off the SPI flash chip, which contains sensitive information, such as hardcoded PKI keys. Research is ongoing to determine necessary steps to mitigate the router's attack surface and to prevent exploitation from sophisticated threat actors.

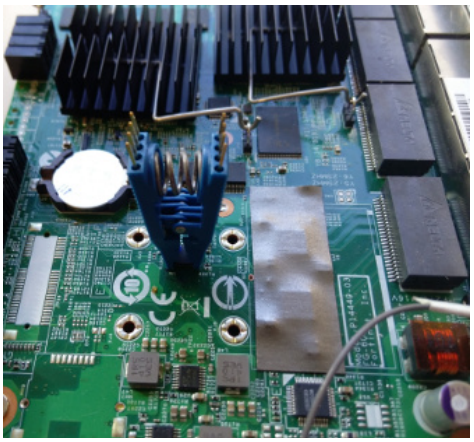


Figure 2. Reading the SPI flash chip.

4.3.1 LAB Zone

The LAB zone consists of individual, isolated hard lines for researchers in the New York City office. Each line lies on its own subnet, and intra-zone communication is disabled. Additionally, the LAB zone may only make outbound connections to the Internet. It may not connect into the server. Researchers utilize independent, anonymous computer resources to access this line, denoted by a red Ethernet cable clearly marked "Malware Lab – Do Not Use". MAC address filtering is enabled on the firewall as an additional precaution against unauthorized users physically connecting to this network.

Researchers may conduct malware analysis on these machines or general security research that may be blocked by the corporate firewall.

4.3.2 MALWARE Zone

The MALWARE zone consists of isolated workstations that primarily are used for automated malware analysis conducted on bare metal and within virtual machines (VMs). They exist on a single, isolated subnet. Additionally, the MALWARE zone may only make outbound connections to the Internet. It may not connect into the server.

4.3.3 Server

The server resides on its own subnet. Firewall rules allow SSH connections over port XXXX to any computer in the LAB zone and MALWARE zone to PUSH samples and retrieve sanitized analysis results. There are no inbound connections allowed from either zone into the server.

4.4 Malware Analysis Workstations

Malware analysis workstations are dedicated, anonymous resources, subject to strict malware handling protocols. Once an asset is used for malware analysis, it will never be recycled for use within the corporate network. The hardware will be considered compromised.

Malware analysis is conducted within a VM. Due to known VM escapes via VM user-interface "helpers", VMware Tools and Virtual Box Extension Pack will not be installed. Shares are not permitted between the host and guest operating system. External devices will not be connected during live analysis, and any subsequent use will be subject to the guidelines below. Samples and results will be transferred in/out of a dead .vmdk via Disk Editor and

the VirtualBox command-line manager. Analysts may use any number of tools inside the VM for static and dynamic analysis. Notable results should be saved to a designated folder for dead extraction.

Known-good images will be kept offline for the virtual machines and base install. Analysis will utilize snapshot capabilities to "rollback" machines to known-good states. Snapshots will be labeled appropriately and deleted once the analysis is complete. Machines will be re-imaged following high-risk events and every six months.

Bare-metal analysis may be conducted on designated, disposable laptops for suspected vm-aware samples and will be re-imaged following analysis.

Ideally, the server will be utilized to transfer samples; however, the Imation IronKey thumbdrive[10] may also be used, reformatting after high-risk exposure. Samples will be renamed with the extension ".malware" and transported in an encrypted .zip file. Additionally, verification of IronKey firmware and filesystem integrity will be performed after data transactions using the Facedancer[11], a specialized USB emulation attack/defense device capable of monitoring and logging USB behaviors.

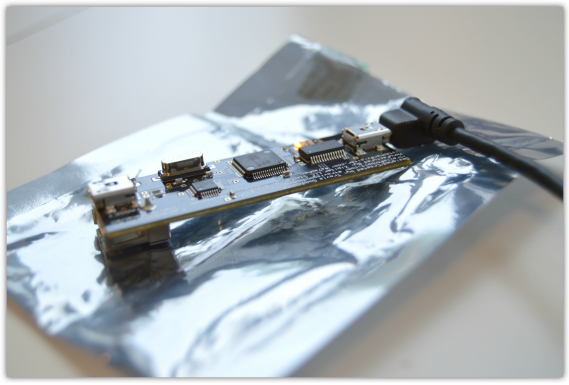


Figure 3. The Facedancer21

4.5 In-Line Packet Capture

The PCAP workstation is an OpenBSD[12] box sitting between the Verizon modem and the FortiGate router/firewall, operating a transparent/IP-less firewall. Traffic from the modem is "bridged" over the data link layer (Layer 2) and passed directly to the router/firewall. We have implemented full packet capture via tcpdump on this interface, which rolls into a new file daily. Due to its statelessness, this machine may not attempt network connections of any kind.

4.6 Internal Workstation

The workstation runs XUbuntu[13] and sits on the internal corporate network and serves as the intermediate launchpad for exporting malicious samples to the server.

5 Summary

Due to advanced tactics, techniques, and procedures of today's attackers, it's increasingly difficult to capture live samples due to short-lived hosting domains, cookie verification, and environment checks. The proprietary technology that allows us to detect incoming exploits also allows unprecedented control over the download process. Recognizing that the production environment is not suitable for automated intelligence extraction, the Malware Lab was designed, purchased and deployed, using industry-standard guidelines and the combined, leading industry experience of the Threat Research Lab.

---

[10] The IronKey is a cryptographically-secure, encrypted thumbdrive, utilizing signed firmware and an encrypted data partition. The thumbdrive is compatible with Linux, OSX, and Windows. The IronKey mounts a read-only "Unlocker" partition containing the software to mount and unlock a separate data partition, which may be mounted read-only as well (http://www.ironkey.com).
[11] http://int3.cc/products/facedancer21
[12] OpenBSD is recognized for unparalleled security standards, a thoroughly audited code base, and history for minimal attack surface exposure.
[13] A lightweight version of Linux (http://xubuntu.org).