Homeland Security

NCCIC
National Cybersecurity and
Communications Integration Center

Malware Trends

Industrial Control Systems Emergency Response Team (ICS-CERT)
Advanced Analytical Laboratory (AAL)

October 2016

Homeland Security

SUMMARY

This white paper will explore the changes in malware throughout the past several years, with a focus on what the security industry is most likely to see today, how asset owners can harden existing networks against these attacks, and the expected direction of developments and targets in the coming years.

CONTENTS

TABLES

ACRONYMS

BIOS	Basic Input/Output System
DDoS	dedicated denial of service
DLL	dynamic link library
ICS	industrial control system
ICS-CERT	Industrial Control Systems Cyber Emergency Response Team
IT	information technology
MBR	master boot record
OS	operating system
RAT	remote access trojan
UEFI	Unified Extensible Firmware Interface
USB	universal serial bus
VBR	volume boot record

MALWARE TRENDS

1. INTRODUCTION

As technology advances and new devices join the ranks of those connected to the Internet, new vulnerabilities and challenges in the security of information technology (IT) and operational technology (OT) systems come along for the ride. This white paper will explore the changes in malware throughout the past several years, with a focus on what the security industry is most likely to see today; how asset owners can harden existing networks and systems against these attacks; and the expected direction of developments and targets in the coming years.

1.1 State of the Battlefield

As individuals and corporations ramp up the number of connected devices on their networks, the volume of personal and confidential information transmitted around the world has grown to an all-time high. In the pursuit of this information, attackers have taken a renewed interest in exfiltration from both individual and corporate environments.

Compounding the risk, rapid evolution of technology has caused the development rate of software to soar. Increased volume amplifies the potential for exposure, heightening the need for heavy software review and testing. However, vendors sometimes neglect security and validation of software because of the need for rapid development. The National Institute of Standards and Technology (NIST) vulnerability statistics[1] show a significant increase in the number of software vulnerabilities reported over the past 2 years. In 2014, 7,937 vulnerabilities were found in software, as compared to 5,186 in 2013—a 34.6 percent increase. The trend seems to be continuing with 5,619 vulnerabilities reported as of November 1, 2015.

Industrial control systems (ICS) devices are following a similar trend in vulnerability discovery. In 2015, the Industrial Control Systems Cyber Emergency Response Team (ICSCERT) coordinated 177 new vulnerabilities. In 2014, there were 161 reported vulnerabilities and in 2013, 181 reported vulnerabilities.[2] These numbers suggest that the discovery of vulnerabilities in ICS devices is still a growing field and that the number of discoveries is likely to increase as researcher interest expands.

As the malware industry grows, more and more people throughout the world are training to find vulnerabilities in software and system configurations. Software vendors are pushing security patches quicker than ever. [3] However, because of the slow nature of the way patches and upgrades are often handled in personal, corporate, and OT environments, victims may still be susceptible to vulnerabilities that developers had patched months or even years before.

Moving forward, it is likely that these patterns of attack will expand to new devices and include an increasing interest in zero-day vulnerabilities. The struggle between attackers and defenders is destined to be never-ending; however, consistent patching practices and a maintained awareness of new vulnerability reports for systems on the network can help reduce the attack surface. According to the 2015 Verizon Data Breach Investigation Report (DBIR), “99.9% of the exploited vulnerabilities were compromised more than a year after the CVE [Common Vulnerabilities and Exposures] was published.” [4]

[1] https://web.nvd.nist.gov/view/vuln/statistics, NIST vulnerability statistics

[2] https://ics-cert.us-cert.gov/Year-Review-2014, ICS-CERT year end review for 2014

[3] https://secunia.com/?action=fetch&filename=secunia_vulnerability_review_2014.pdf, Secunia review of vulnerability patch availability in 2011 through 2014, page 103

[4] http://www.verizonenterprise.com/resources/reports/rp_data-breach-investigation-report_2015_en_xg.pdf- pg. 15, Discussion on the impact of patching on data breaches

2. ATTACKER TACTIC CHANGES

Historically, most shifts in attacker posture have been gradual, and we see this confirmed in recent incidents. Typically, malware capabilities will vary based on the end goals of the malware author and the targeted entity. However, the overall shift toward robustness, along with a push to specifically attack end user devices of interest, can be seen across all campaigns, which have targeted industrial, corporate, and personal environments.

2.1 Malware as a Service

Malware as a service (MaaS) and related variants[5] are creating a market for malicious software and distributed targeting that have gained a great deal of popularity in the past 4 years. This market provides a customer (the attacker) with access to exploits, use of a botnet, or the creation and distribution of malware. Essentially, attackers can outsource much, if not all the technical load, for a price.

The market for malware is growing rapidly, and while it is not tied to any specific group of threat actors or family of malware, it significantly lowers the technological barrier to entry for would-be criminals. High-end services, such as those sold by firms such as Hacking Team, [6] Vupen,[7] or Zerodium, [8] allow customers to access and utilize advanced exploits, including zero-day attacks, on demand. Many other groups offer a variety of tools such as user-friendly malware customization and simple point-and-click distributed denial-of-service (DDoS) solutions backed by massive botnets. These services operate using robust business models, such as enterprise level support to customers, creating an underground community and marketplace for attackers. [9] This increases the accessibility of intrusion and market disruption, and reduces the overhead for launching a campaign down to a simple, nominal fee, eliminating the need for technical knowledge and resources on the part of the attacker.

As online services and cloud storage become increasingly ubiquitous, some exploit and malware developers are shifting their focus from gathering information and performing breaches on live networks to development of the exploits and tools required to accomplish these operations. This creates a shift toward specialization, allowing third parties to decide what information they want to target and malware developers to hone their craft. It also serves to separate the malware and exploit developers from the crimes being committed. This model continues to yield substantial monetary compensation for all parties and their efforts, while affording the developers a more stable and predictable income and lifestyle, with less risk of apprehension.

This transition mirrors the software development transition the industry has seen in web-based software as a service (SaaS), as well as platforms as a service (PaaS). These product models became popular in 2001 and reached widespread adoption from 2005 to 2008. Overall, this shift has led to higher quality products and more focused attacks. Section 3 will cover these changes in malware development in more detail.

2.2 Destructive Malware

In a major recent trend, attackers have targeted a wide range of systems with destructive malware. This variety targets a user's files, specifically or indiscriminately, and renders them inaccessible. These

[5] Variants include Hacking as a Service (HaaS), Crimeware as a Service (CaaS) and Fraud as a service (FaaS)
[6] www.telegraph.co.uk/technology/8899353/The-spies-behind-your-screen.html - Article discussing the hacking team company and their market for both offensive and defensive utilities
[7] www.forbes.com/sites/andygreenberg/2012/03/21/meet-the-hackers-who-sell-spies-the-tools-to-crack-your-pc-and-get-paid-six-figure-fees/#6cc1ba694483 – Vupen article detailing the goals and products of the exploit developer
[8] https://threatpost.com/vupen-launches-new-zero-day-acquisition-firm-zerodium/113933 - Article detailing the Zerodium acquisition firm, created by the founders of Vupen and focused on acquiring exploits.
[9] https://securelist.com/analysis/publications/72652/breaches-carnivals-and-cybercrime-a-look-inside-the-brazillian-underground/ - Exploration of the criminal sharing of malicious services and applications between cyber criminals in Brazil.

attacks result in the most significant losses when used against corporate entities because of large costs associated with recovery. In the Sony Pictures Entertainment attack in December 2014, where malware wiped victim hard drives, [10] they estimated the total cost of recovery to be $15 million. [11]

Destructive malware can be devastating to an unprepared victim. The best mitigation strategy relies on regular, frequent data backups and network hardening. Potential signs of infection include unusual file corruption discovery, frequent system lockups, and in the case of ransomware, a message demanding monetary compensation in exchange for restoration of the victims files. To raise awareness and provide remediation and mitigation strategies, ICS-CERT has released a document[12] addressing destructive malware as a whole.

2.2.1 Ransomware

Ransomware is a form of malware that has been around for over 25 years, [13] but it did not gain much traction until 2013. Because of the expansion of cryptocurrencies, such as Bitcoin, which allow the untraceable transfers of funds, along with the ease of development and a strong return on investment (ROI), these attacks are seeing rapid growth. In the first quarter of 2015, McAfee Labs observed a 165 percent increase in new ransomware over the fourth quarter of 2014 with approximately 725,000 observed new ransomware samples over the previous quarter's 270,000. [14]

The goal of ransomware involves gaining access to a user's device and iterating through his or her personal files, encrypting everything of interest with a key typically known only by the attacker. The attacker then makes a demand, usually a transfer of funds through a currency such as Bitcoin or Moneypak, in exchange for decryption of the files. The attacker places a time limit on the user to acquiesce to the attacker's demands after which the files will be permanently unrecoverable.

During the last 2 years, a significant amount of growth in the quality and distribution of ransomware indicates that this genus of malware is on the rise and will not be short lived. Famous examples of ransomware include CryptoLocker (which was shut down when the Gameover ZeuS botnet was disrupted), along with CryptoWall, TorrentLocker, and most recently, TeslaCrypt.

Ransomware employs many of the same tactics used by other malware to avoid detection and operate stealthily. Attackers frequently use network-hiding tools such as Tor in such attacks.

It is highly likely that ransomware will continue to grow as a source of concern. They are relatively simple for attackers to create; the potential payouts are massive; and remediation methods are difficult once the attack has compromised the victim. The most effective defense against these attacks, as with purely destructive malware, is regular, frequent backups of systems. Without a backup of a compromised system, the asset owner is at the mercy of the attacker. Even more critical, a piece of ransomware in an ICS environment could cause a long-term disruption of service before administrators can restore the system to working condition.

In its 2015 Global Security Report, Trustwave found that purchasing ransomware yielded a 1,425 percent ROI. These returns are possible due to MaaS groups delivering the crimeware, the necessary exploits, and the distribution methods for attackers to use to target a larger number of victims with increasing effectiveness.

[10] http://blog.trendmicro.com/trendlabs-security-intelligence/an-analysis-of-the-destructive-malware-behind-fbi-warnings/ - analysis of the destructive malware thought to have infected the Sony network.
[11] www.latimes.com/entertainment/envelope/cotown/la-et-ct-sony-hack-cost-20150204-story.html - cost of Sony Pictures Entertainment hack in damage and recovery.
[12] https://ics-cert.us-cert.gov/tips/ICS-TIP-15-022-01 - ICS-CERT mitigation strategies against destructive malware
[13] https://medium.com/un-hackable/the-bizarre-pre-internet-history-of-ransomware-bb480a652b4b - A historical beginning to Ransomware
[14] www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2015.pdf - An analysis of new ransomware seen from Q1 2013 to Q1 2015, page 14

3. MALWARE EVOLUTION

3.1 Design Changes

In the early history of cybersecurity, it was common to see one of two directions taken in an attack involving malware: 1) using many different tools, each for a different purpose; or 2) using a single piece of malware with an extensive feature set. In recent years, the anatomy of an attack has grown more cohesive. Instead of employing many unrelated tools from varying teams or a massive all-purpose application, modern malware has opted to take an approach similar to current large-scale software. In malware such as PlugX, [15] the authors have created a platform providing extensible functionality through a plugin interface. These plugins add new functionality, such as gathering a user's password hashes or monitoring for anti-malware applications, providing attackers with the flexibility to adapt their strategies and tool deployments to new situations and environments.

This kind of modular design hinders the detection and analysis of malware. By extending functionality through plugins, and only deploying the plugins specifically desired to a compromised host, the full arsenal of the attacker remains unknown to researchers. By finding only the loader platform, analysts may have little idea of what occurred on the system or the attacker's goals without also locating plugins or forensic artifacts. Malware families that include a primary backdoor and/or a remote administration capability provide an attacker with arbitrary access to a compromised host. An attacker may take any number of actions with that level of access, which can significantly complicate forensic analysis.

Where it used to be common to see major malware families spread like wildfire without care for who the end victims were, we are now seeing much more deliberately targeted attacks on single victims. On Page 22 of their 2015 DBIR, Verizon says, "70-90% of malware samples are unique to an organization." [16] This is both 1) a result of custom-built malware, designed for one-time use against a single entity; polymorphic malware, which is altered for each distribution to disrupt detection from antivirus applications; and 2) tailored versions of plugin-based malware, which leverage specific functionality to accomplish the attacker's goal at specific victim sites.

3.2 Obstructing the Analyst

As the complexity and distribution of malware rises, techniques used by malicious authors to hide their work have improved. Attackers have spent a large amount of effort making the job of analyzing malware more difficult and time consuming. Increasing numbers of anti-sandbox, anti-debug, and anti-analysis techniques (such as dead code, encryption, and junk code) inhibits both static and dynamic analysis of applications, causing delays in the response time of researchers and analysts.

Significant growth in techniques to evade initial detection for new malware samples is a major issue for those working to protect networks. Observed tactics include anti-emulation techniques, packing, narrow dissemination, and executing and residing only in memory, among others. Each of these techniques can lead analysts down a rabbit hole of misdirection or even completely prevent analysis.

[15] http://labs.lastline.com/an-analysis-of-plugx - Analysis of the PlugX malware that covers the architecture of the base application and plugins

[16] http://www.verizonenterprise.com/resources/reports/rp_data-breach-investigation-report_2015_en_xg.pdf - p22, Verizon discussion of the targeting seen in malware

Table 1. Sample Obstruction Techniques

Technique	Description
Environment Checking	The application checks to see if it is being debugged, emulated, or running under a virtual machine (VM), and then either stops executing or redirects execution flow to an alternate path. Examples of checks may be debug flags, the presence of certain processes or applications, timing checks, or special VM I/O ports.
Packing	The application constructs an executable or extracts additional code during runtime from code stored within itself or a separate file.
Narrow Dissemination	The application targets a specific environment or targets and does not infect other systems. Attempted compromises may occur through spear phishing, IP targeting through malvertising, or other means.
Memory Resident	The application does not exist on the file system and executes only in memory. Once the device is powered down, the application and, potentially, evidence of its execution disappear from the system.

3.3 Shared Code Base

A number of malware tools and exploits have either released their source code or have had it leaked. These releases have become invaluable sources of ideas, attack vectors, and techniques for new and experienced attackers to use as references for their own tools. These attackers have significantly lowered the bar for advanced malware development.

Much of the malware that is publicly available is well known and has been seen in various attacks for a significant period of time, such as Zeus, gh0st, and Carberp. As new catalogs leak from groups such as the 2015 Hacking Team release, new, previously unknown malware and zero-days enter into the public arena. Attackers often redesign or repurpose these kinds of tools, adding new capabilities and changing the potential impact of the malware.

ICS devices are not an exception for this market. While attacking ICS devices is appealing to a much smaller audience, multiple development firms have been in the business of finding, creating, and selling exploits for major ICS vendor products. Currently, businesses are creating penetration tools against supervisory control and data acquisition (SCADA) systems[17] designed for security testing of ICS environments. While these tools have real-world application for testing the resiliency of the network against malicious actors, these resources pose a real danger due to their exposure of live ammunition when in the hands of an attacker.

One concern for analysts with the splitting of a piece of malware into many variants is that it becomes much more difficult to categorize based on previously observed behaviors. This forces malware analysts to find connections between the new varieties and perhaps their creators. It is likely that new teams who have picked up these packages will have different goals for the repurposed frameworks than the original designers. Therefore, they will require reassessment and consideration when found again in the wild. This has led to drastic shifts in previously known samples, such as the evolution of BlackEnergy from a simplistic DDoS application[18] to a plugin driven, multi-platform back door utility in BlackEnergy 2 and 3.

[17] www.forbes.com/sites/thomasbrewster/2015/10/21/scada-zero-day-exploit-sales/#4ffc11dd96c – Forbes article detailing the SCADA device exploit market

[18] http://atlas-public.ec2.arbor.net/docs/BlackEnergy+DDoS+Bot+Analysis.pdf - Analysis of the original BlackEnergy botnet in 2007

4. PERSISTENCE METHODS

Attackers and researchers have developed a number of mechanisms for maintaining persistence and continued access on compromised systems. As antivirus vendors push new signatures with greater speeds and continue to improve heuristics, threat actors have had to adopt new techniques to evade detection and preserve their control on hosts.

A common persistence technique used by malicious applications is to alter entries of the compromised system's registry and file system to ensure execution after the asset owner has rebooted the system. Techniques to do this are numerous and vary depending on the operating system. Common Windows methods include adding an entry to the system's or user's Run registry key, placing the executable or a shortcut in a user's startup folder, scheduling a task to launch the malware directly at certain times or events, or replacing a commonly accessed resource or legitimate application to load a malicious dynamic link library (DLL).

These techniques have proven effective and the level of complexity varies between different families of malware. Avoiding the primary file system is typically ideal, as antivirus programs focus on searching the standard file system for evidence of an infection with less emphasis on memory. This has led to a number of unusual and novel approaches to hiding code in places that today's antivirus applications would not normally scan, such as the file-less trojan Poweliks[19] storing itself in the Windows registry.

4.1 USB Firmware

Many attacks have made use of universal serial bus (USB) memory sticks to spread into devices outside of the traditional network methods, the most infamous of which was Stuxnet.[20] USB attacks are unique due to the potential for infection on devices that are physically isolated from unsecured networks, also known as "air gapping." This is particularly relevant to ICS environments with separate operational and business networks. These environments are out of reach to attackers using more traditional network infection vectors.

USB attacks have become more robust over the years, ranging from simple executables that live on the file system of a portable storage device to the actual firmware of a USB device, which the USB controller chip executes each time the device connects to a computer. This infection vector is particularly dangerous because it puts users of any type of USB device at risk, not just those using storage devices. In addition, the infection could be difficult to detect, silently spreading from device to device as users attach and detach USB components around the company. This could allow malware to propagate and spread until a user attaches it to a device with access to the Internet or the malware reaches its final target.

Because of these risks, it is prudent for companies and individuals to know the hardware they are purchasing and using comes from trusted sources, and never to use their hardware on untrusted machines. Mitigation of this kind of attack relies on segregation of devices: keep all trusted hardware devices off untrusted machines, such as personal laptops or mobile devices, and keep all untrusted devices off trusted machines.

Education of users is paramount in preventing these kinds of attacks, because many users do not understand the risks associated with the technology. A study[21] by CompTIA placed unlabeled USB devices in high traffic areas with links in their root directory prompting users to click them. Of the 200 devices dropped, 17 percent of the links received clicks, indicating that at least 17 percent of the individuals that

[19] https://blog.gdatasoftware.com/blog/article/poweliks-the-persistent-malware-without-a-file.html - Malware injection into registry for extraction and execution at runtime
[20] www.langner.com/en/wp-content/uploads/2013/11/To-kill-a-centrifuge.pdf - Technical analysis of Stuxnet, this paper details the use of USB devices for spreading the malware
[21] https://www.comptia.org/resources/cyber-secure-a-look-at-employee-cybersecurity-habits-in-the-workplace - Study of how the average individual interacts with unknown hardware devices

found the devices plugged them into computers and executed the contained code. Monitoring for devices attached to machines on a corporate network can help to identify potential breaches of this policy, and allow incidents to be contained before an attack has the ability to spread further. However, the greatest reduction from these behavior-based attacks is the education of users.

4.2 BIOS

One of the places to hide low-level persistence mechanisms has been in the master boot record (MBR) or volume boot record (VBR) of the boot disk. [22] The MBR and VBR are low level and include instructions that execute before the OS has booted, and before any antivirus software has loaded. This makes infections at this level potentially untraceable through memory or process scanning performed at the OS level. These attacks will rarely implant any files into the actual OS file system, instead opting for code injection directly into process memory[23] or within sectors of the disk not addressable by the OS. [24] Any malicious payloads that execute in this manner may contain a component that manipulates any reads from the OS to the MBR or VBR to return the original, expected codes rather than the true modified version, making the payload nearly undetectable at the user level.

Basic Input/Output System (BIOS) level malware takes a lower level approach than MBR and VBR bootkits, infecting the actual code running immediately after powering on a system. This form of malware is particularly dangerous, as the BIOS has privileged access to hardware on boot. Any code injected into the BIOS could directly affect any aspect of the system by executing at a lower level than any other code on the system. Researchers saw this kind of attack in the wild as early as 2011 in the Membromi virus,[25] identified as an infection spreading through China, and demonstrated by researchers through proof of concepts such as lighteater[26] in 2015.

What makes this particular form of malware more threatening is that a great deal of remediation techniques—such as reformatting the device, or running entirely off a live disc—do not mitigate the risk. Only fully restoring the BIOS of the motherboard would take it out, a process that IT personnel typically avoid because of a higher potential for costly issues to arise during flashing. Even with this level of remediation, re-flashing assumes that the malware has not modified the internal flashing process during its time on the box, which could make any flashing via software ineffective or destructive and would require specific hardware devices to flash the board directly. Ultimately, this kind of exploitation can easily lead to a full replacement of hardware because of the asset owner's inability to determine if they have truly eradicated the malware.

Legacy BIOS infections have fallen off in the wake of Unified Extensible Firmware Interface (UEFI), a specification for the interface between an operating system and platform firmware. UEFI introduced in 2005 to modernize the boot process and fix the limitations in both security and functionality of legacy BIOS. While the standard has been in place since 2005, implementation details continue to differ between manufacturers and adoption of such a large standard takes time. As a result, research into security issues of UEFI have only begun to gain momentum recently, with research released in 2012 for Mac Extensible

[22] www.symantec.com/connect/blogs/are-mbr-infections-back-fashion-infographic - Highlights the history of MBR and VBR malware as it rose in popularity.

[23] http://malware.dontneedcoffee.com/2014/08/angler-ek-now-capable-of-fileless.html - Malware injection directly into process memory

[24] https://www.f-secure.com/weblog/archives/00001393.html - Analysis of MBR Rootkits for Backdoor.Win32. Sinowal.Y from F-Secure

[25] www.webroot.com/blog/2011/09/13/mebromi-the-first-bios-rootkit-in-the-wild/ - Documentation of the Membromi BIOS attack

[26] www.legbacore.com/Research_files/HowManyMillionBIOSWouldYouLikeToInfect_Full2.pdf - Researcher-developed lighteater is a malware tool for infecting PC BIOS.

Firmware Interface rootkits[27] and the first proof of concept to defeat the Windows 8 secure boot system coming in 2013 in the form of DreamBoot.[28]

The leaked HackingTeam documents contained details of a partial implementation of a UEFI rootkit with many of the same exploitations seen in prior BIOS malware. Trend Micro[29] and Intel's Advanced Threat Research[30] have dissected the leaked code and provided analysis of the source code, showing that if there is security in place on the Serial Peripheral Interface (SPI) flash of the UEFI, one would have to flash the exploit by physically accessing the PC. However, they point out that this sort of attack may eventually be available in the wild without such requirements. With the leaked code, any malware authors who are interested in recreating the technique will benefit from this head start on development.

Researchers have been on the hunt for issues with UEFI and have found a number of vulnerabilities. However, while UEFI may have some of the same problems that plagued standard BIOS firmware devices, support built into the standard, such as Secure Boot,[31] have improved the security of the BIOS. Intel Security has done a great deal of research on reducing vulnerabilities in UEFI motherboards currently in production, and Kaspersky has released a product aimed directly at scanning at the UEFI level for malware.[32] Other companies, such as VirusTotal,[33] have also been searching for stronger characterization of malicious firmware images through analysis. Despite these efforts, the topic is still a major concern within the field.

Detection of UEFI malware can be time consuming, but there are tools, such as CHIPSEC[34] and UEFITool,[35] to allow for image extraction and exploration. These tools utilize system drivers to allow for direct access to the low-level firmware. However, much like the possibility of the infected BIOS serving an incompatible flashing tool, the returned examination of the UEFI layer by the kernel may also be spoofed. The only certain way to extract the image and be sure you are seeing an accurate representation of the firmware level is to use a tailored hardware tool to read it directly from its chip, which may in turn be encrypted.

There are other low-level attacks, such as manipulating or replacing hard drive firmware as seen by the Equation Group[36] and System Management Mode (SMM) rootkits as demonstrated by the University of Central Florida.[37] While there are many examples of these kinds of attacks, many more will inevitably

[27] http://ho.ax/downloads/De_Mysteriis_Dom_Jobsivs_Black_Hat_Paper.pdf - Black Hat paper on attacking the EFI layer
[28] https://www.virusbtn.com/pdf/conference/vb2014/VB2014-RodionovMatrosov.pdf - A history of BIOS bootkits, through the first proof of concept release of DreamBoot
[29] http://blog.trendmicro.com/trendlabs-security-intelligence/hacking-team-uses-uefi-bios-rootkit-to-keep-rcs-9-agent-in-target-systems/ - Trend Micro analysis of Hacking Team UEFI malware in development
[30] http://www.intelsecurity.com/advanced-threat-research/ht_uefi_rootkit.html_7142015.html - Intel Security analysis of the Hacking Team UEFI malware in development
[31] https://technet.microsoft.com/en-us/library/hh824987.aspx - Secure Boot for UEFI overview
[32] http://media.kaspersky.com/en/business-security/Kaspersky_AV_for_UEFI_1.0.pdf - scanning tool that boots during the BIOS loading procedure to scan for potential malware in the firmware
[33] http://blog.virustotal.com/2016/01/putting-spotlight-on-firmware-malware_27.html – VirusTotal blog entry describing characterization and profiling of malicious firmware images
[34] https://github.com/chipsec/chipsec - a hardware analysis tool that can be used to extract the BIOS of a PC. This tool includes many powerful features for forensics on the device. Designed to be an extensible platform
[35] https://github.com/LongSoft/UEFITool - a UEFI tool that is built exclusively for UEFI. This tool allows the researcher to extract and modify UEFI firmware images
[36] http://www.wired.com/2015/02/nsa-firmware-hacking - Equation Group hard drive firmware rootkit overview
[37] http://www.eecs.ucf.edu/~czou/research/SMM-Rootkits-Securecom08.pdf - Research into SMM rootkits, designed to attack the device at the layer where OEM code would be used to handle power management and USB device communication

come, and all devices with firmware are potential targets. It is increasingly important for all system administrators to keep these kinds of attacks in mind when deploying critical systems. Even if a user's security product says nothing is wrong, assurances cannot be made about these lower levels by common utilities. Asset owners should isolate heavily mission critical devices, such as ICS control systems, from other devices to the highest extent possible, both over networks and via removable devices, as soon as they go into operation. Ensuring absolute security is an impossible task and adding risk only serves to reduce the system's security.

4.3 Remediation for Persistence

While Sections 4.1 and 4.2 touch on a number of persistence methods, many more are available to malware authors. Because OSs are extensible, there are many areas of each system for malware to hide and wait for execution by a common process. Further, the unconventional methods of persistence, such as USB infection and BIOS modification, are single points of vulnerability. As standards in technology evolve to accommodate new types of devices, new vulnerabilities will inevitably begin to arise as well.

Defending against persistence is twofold. Strong security to limit the changes on the file system and flagging suspicious behavior can hold off many malware persistence methods that rely on the file system (see Sections 4.1 and 4.2). However, for unconventional forms of persistence and spreading, it is necessary to enforce policy changes on which devices the asset owner uses for specific tasks, which systems may have connected devices, and careful sourcing and tracking of devices used on trusted systems.

5. INFECTION VECTORS

Attackers use a number of techniques to get malware onto victim devices, and they have been using these same techniques for many years to great effect. Because of the communication-centric nature of the Internet and modern computing in general, the risk of infection through downloaded software, attachments, and web pages will remain high for years to come, because there is no true mitigation against some level of implicit trust of devices on the other end of the communication.

5.1 Trojanized Software

Trojanized software is software that a threat actor has infected with malicious code and redistributed. This attack relies on users either not getting the software from the distributor directly, or worse, on the vendors becoming compromised themselves, allowing the attacker to modify the application at the source. An infected vendor can have a detrimental effect on its users. If an attack compromises a build system or source code repository, the vender may distribute the software before they detect the compromise. This kind of incident occurred in the case of the Havex malware,[38] which attackers distributed through a trojanized download on multiple ICS software vendors' websites.[39] Fortunately, the repackaging and redistribution of packages on third-party hosts is far more common than compromising and altering packages on official vendor channels.

Trojanized software is a rather simple infection vector. The user intentionally provides the means for the application to install and modify the system by installing the trojanized application in the first place. However, without centralized software repositories (such as "app stores"), it can become difficult for users to find the true source of software, particularly those from smaller developers.

Mitigation of trojanized software is, in the general sense, a straightforward problem for corporate environments. By preventing employees from installing their own new software without going through IT, and by having IT source and verify all new applications to be installed for correct distribution sources and

[38] https://ics-cert.us-cert.gov/alerts/ICS-ALERT-14-176-02A - Havex RAT advisory

[39] www.netresec.com/?page=Blog&month=2014-10&post=Full-Disclosure-of-Havex-Trojans - Information on Havex trojanized software

hashes, an organization can avoid the vast majority of trojan issues. The organization may accomplish this verification through a validation of an application's digital signature to help prove that the distributor of the software is legitimate. This combined with hash verification provides further assurance that the executable in question is in fact identical to what the vendor has released. However, digital signing of provided executables and providing hashes has not become standard practice for every vendor.

For software from potentially compromised vendors, the issue becomes much more complex. It is necessary for IT administrators to vet carefully all applications on the network before approving for installation on trusted devices.

5.2 Phishing

The technique of phishing to either infect a network and/or gather user credentials is one of the most pervasive challenges for both personal and corporate networks today, with phishing accounting for 55.28 percent of email traffic in 2015.[40] Attacks revolve around deceiving the user into either opening a file or link embedded in a message (email, instant message, social media post, etc.).

This form of attack is particularly effective when used in conjunction with personalized information about the user in question, such as their name, interests, accounts, activity, or systems that they regularly use. Attacks using these tailored kinds of messages called as spear phishing. They generally target a single pool of users such as employees working for a specific company or customers of a particular bank. These spear-phishing campaigns can be extremely convincing, and with the rise in information available about individuals on the open web, the potential for carefully crafted fraud is much higher than in the past.

Organizations can best protect against phishing by blocking suspicious communications with spam filters through policy, and by educating end users. Tailored spear-phishing attacks will defeat spam filters, despite the fact that spam filters have become extraordinarily effective in the past decade. These sorts of issues require the education of employees in a corporate environment. Employees must be vigilant and avoid suspicious-looking links or attachments. If anything seems odd about an email, they should confirm with the sender in person or over the phone or forward the email to their information security officer for review.

5.3 Watering Holes

Watering holes are another technique that attackers use to infect users, particularly when targeting a certain industry. The attacker aims to compromise a web site or shared resource frequently used by the employees of a company or sector, injecting malware into web pages to infect employee computers when utilizing the system.

These attacks can be particularly effective for companies with older or unpatched servers used internally, though the concept is the same regardless of how the attacker carries it out. Once the attack compromised the servers, the attacker modifies the web pages and resources to load malicious code. This code then reaches the end users who are accessing the resource and potentially infects their system with malware.

These attacks can be devastating when carried out on a commonly accessed resource such as a homepage or time tracking system on the company network. This is because the attacks can lead to a rapid and pervasive infection. It is critical to keep internal resources patched, perform monitoring on systems regularly, and watch for suspicious modifications to any resources they contain or access. A strong rule to keep in mind is that the asset owner must account for the criticality of the system when considering rules for security.

[40] https://securelist.com/analysis/quarterly-spam-reports/71759/spam-and-phishing-in-q2-of-2015/ - Kaspersky report detailing the state of phishing attacks seen in 2015

5.4 Malvertising

Malvertising in the current landscape of the Internet is a prime target for malware distribution. Much of the Internet is supported through advertisements, which provides a great deal of income to many sites, allowing content creators to monetize otherwise unmarketable content. Web sites often base advertisement sales on features allotted to the ad itself, page space, and the popularity of their host sites.

A primary difficulty comes with adding features to ads, particularly in the form of active scripts and interactive content. The issue here comes in when a threat actor embeds malicious code into advertisements[41] on well-established and reputable sites, compromising users via an exploit kit from an otherwise innocuous page.

The growing concern in the last year is malicious advertisements slipping into large mainstream web pages, such as Forbes, Daily Motion, and MSN. [42] Because of the breadth of these kinds of attacks, they can be difficult to mitigate from the perspective of corporate IT. Whitelisting of websites through the firewall is the most effective means of mitigation for untrusted pages. But trusted pages may still pull content from ad networks compromised with malicious advertisements and manually whitelisting web sites are resource intensive for IT and frustrating for employees with need to access new pages. In some environments, it may be necessary to utilize tools for script blocking, advertisement filtering, or the techniques mentioned for script control and alerts in Section 6.4.

For high security requirements, a stronger approach is to implement complete network separation between trusted and untrusted devices. Systems that are trusted on the network sit behind a firewall using whitelisting, blocking JavaScript along with extensions, such as Java or Flash, and with access only to known good pages. Administrators can use a second set of systems for all general browsing that connects to a firewall with only a blacklist of known malicious domains. This allows for tighter control of what content crosses over the local network, while still allowing employees access to the necessary information to do their job without blockades from IT policy.

6. DEFENSIVE TACTICS

With changes in attacker posture and techniques, defenders must continuously adapt their tools and techniques to stay ahead of breaches. This section presents core methodologies to detect malicious activity on the network and contain infections before they cause extensive damage.

6.1 Monitoring

Network traffic and system real-time monitoring is one of the most impactful lines of defense available to administrators. Unusual IP connections in and out of the network are often the first hints that something out of the ordinary is occurring, while changes on machine file systems outside of the user's typical usage are often the first hard pieces of evidence to indicate a compromise.

Monitoring should be the first active level of defense. Administrators should use tools such as firewalls, real time malware scanning, network packet inspection, and intrusion detection systems (IDS) at ingress/egress points on the network to improve knowledge of what is coming in and going out of the network. This knowledge is more actionable than finding out a compromise occurred months or years in the past.

[41] http://blog.trendmicro.com/trendlabs-security-intelligence/youtube-ads-lead-to-exploit-kits-hit-us-victims/ - Trend Micro article detailing ads embedded in YouTube pages that have been directed to a hijacked address containing malware

[42] http://www.engadget.com/2016/01/08/you-say-advertising-i-say-block-that-malware/ - Malvertising on trusted, large web sites

6.2 Logging

Logging goes hand in hand with monitoring. Ultimately, when an attack occurs and succeeds, it will likely take the victim by surprise, even if the monitoring tools are successful in detecting the malicious activity. Many breaches into networks go undetected for a long period of time before detection. In 2015, Mandiant found a median of 146 days between breach and detection,[43] plenty of time for the attacker to have gathered the information they desired and moved on.

It is common for attackers to evade the initial detection of monitoring tools, slipping into the system through an unmonitored avenue such as malicious email attachments or through a misconfigured firewall rule. These missed detections can leave a network in a state of uncertainty, especially in the case of destructive and information gathering malware. Without logs to determine the time, events, and methodology of the initial infection, administrators are left working blindly, trying to retrace the attacker's steps. Failure to remove all entry points that an attacker originally used may lead to the reinfection of the system and force the asset owner to perform the entire remediation action again, which is an expensive endeavor.

Logs take many forms and should be set up with the same degree of time and care as the detection and monitoring systems were. Network logging at firewalls, file systems, and event logging on servers and clients, logs for antivirus scan results, and logs indicating local or remote user login and logout activities are some of the more useful information repositories from the perspective of a remediation team. Each security incident is unique, but logs provide a solid foundation for investigation and often substantially reduce the time taken to evaluate the malware's trajectory and method of persistence.

Administrators should treat log files in a similar manner to other valuable data on the system. Backups of log files in the event of a data loss may be invaluable for triaging unexpected compromise or system failure, as they allow both analyst and system engineer to troubleshoot and track down the source of both intrusions and failures. The organization should determine the duration of logs. The Health Insurance Portability and Accountability Act (HIPAA) requires that transaction data to be kept for 6 years,[44] whereas Department of Defense (DOD) requires that server transition history be retained for 2 years.[45] Each industry needs to determine its range in the spectrum, keeping in mind that too little will not allow the perspective required to see the full picture if an attack slips into the network too quietly and goes unnoticed.

6.3 Preventing and Reducing Network Contamination

When an asset owner detects an intrusion, the breadth of the information loss is often difficult or impossible to determine. When the asset owner detects an infection, the devices attached to the infected machine are all at risk, including any information servers or services to which they have had access.

Defense-in-depth strategies will prevent or limit the extent of a compromise to a network from any single point of intrusion. Once an attacker has exploited a device, a compromised host can traverse a flat network as if the attacker was the legitimate user of that system, with full access to Active Directory or any other service on the network afforded to them.

Designing a defense-in-depth network requires an asset owner to design a specific structure into the network that separates each level of the connected system from one another, keeping the most critical machines as far excluded and protected from the open Internet as possible. ICSCERT recommends achieving this separation through zoning and the implementation of protections to each zone unique to their re-

[43] https://www.fireeye.com/current-threats/annual-threat-report/mtrends.html – Mandiant report on the state of attacks against corporate environments, with 146 days between breach and detection (page 4)
[44] http://csrc.nist.gov/publications/nistpubs/800-66-Rev1/SP-800-66-Revision1.pdf - HIPAA guidelines, log preservation in section 4.22
[45] https://www.stigviewer.com/stig/web_policy/2011-10-03/finding/V-23844 - DISA STIG guidelines for web server access logs

quirements. [46] The design of each network will be slightly different depending on its requirements, but the principles of network topology will follow the same pattern of hindering the lateral movement of attackers within the network.

It is critical that each system has access only to the devices and files that are required for their operation. The concept of a device that can access all elements of the network provides far too much risk to the integrity of the network if as attacker ever compromises the "master" device. It is important for network designers to keep the principle of least privilege access controls[47] in mind with each device added to the network.

6.4 Vulnerability Exploitation

The ramifications of zero-day attacks can be devastating to large networks and are of critical concern to those on sensitive networks; however, the number of attacks that use zero-day exploits is small—less than 0.1 percent of all breaches recorded in 2014. [48]

There are multiple products on the market designed to help those in industry take precautions against zero-day attacks.[49] Exploit mitigation software such as Microsoft's Enhanced Mitigation Experience Toolkit (EMET), MalwareBytes' Anti-Exploit, and HitmanPro.Alert attempt to disrupt or prevent common exploitation techniques. Preventing execution of scripts using tools, such as NoScript and Privoxy, along with enabling click-to-play plugin functionality within browsers, [50] can help prevent exploitation attempts and raise user awareness of the execution of scripts on sites.

Technologies to mitigate these risks include IDS with deep packet inspection, monitoring traffic for unusual beaconing, data-loss prevention systems, regular file system scanning using a tool such as YARA,[51] scanning the system with antivirus software, and monitoring for indications of compromise such as irregular registry changes.

This level of prudence, in tandem with frequent patching, greatly increases the chances that an asset owner can either completely thwart or quickly identify attacks. However, there is no "silver bullet" solution against zero-day exploits, which is part of the reason why attackers value them so highly. Through careful monitoring, however, asset owners can either prevent or identify and expunge many attacks before they cause irrevocable damage.

7. PLATFORM CHALLENGES

Each platform faces its own challenges, making it difficult to respond in the same way for one platform as would be appropriate for another. The discrepancy between platforms causes issues with responders having the background for each potential target involved in an incident and could cause IT administration to overlook a misconfigured device. In essence, each platform is vulnerable to every malware category found on any other platform, but attack vectors are often different depending on the characteristics of the platform in question.

[46] https://ics-cert.us-cert.gov/sites/default/files/recommended_practices/NCCIC_ICS-CERT_Defense_in_Depth_2016_S508C.pdf - ICS-CERT recommendations for defense-in-depth strategies in network design and deployment

[47] https://buildsecurityin.us-cert.gov/articles/knowledge/principles/least-privilege - US-CERT Guidelines for least privilege implementation

[48] https://secunia.com/?action=fetch&filename=secunia_vulnerability_review_2014.pdf - Secunia review of vulnerability patch availability in 2011 through 2014, page 103

[49] https://securelist.com/blog/security-policies/71915/indicators-of-compromise-as-a-way-to-reduce-risk/ - Kaspersky recommendations on monitoring and remediating for compromised systems from both identified and zero-day compromise attacks.

[50] http://howtogeek.com/188059/how-to-enable-click-to-play-plugins-in-every-web-browser/ - Enable click-to-play for scripts inside browsers to prevent users unintentionally executing scripts

[51] http://virustotal.github.io/yara/ - YARA file analysis. A tool used to scan for potentially malicious files on a system

7.1 PC

The personal computer (PC) environment, including Windows, Linux and OS X, has evolved over a long period of time, incorporating a myriad of features for end users over the course of their extensive development. Originally, the designers of all these systems were going for usability rather than security, which massively extended their user base.

With a constantly expanding and evolving collection of needs from users across the personal and corporate computing space, vendors have designed these platforms for extensibility, so asset owners can tweak and tailor them for as many diverse environments as possible. This functionality has permitted many businesses to thrive on using PC OSs in many environments, from users editing files and accessing web pages to direct control of ICS automation tasks. However, this openness has forced the modern OS to expose a great deal of control and resources to all software running on it, as well as preserve compatibility with applications that asset owners have uniquely tailored for any number of environments as the OS evolves.

The flexibility of OSs on PCs today has led to many different persistence mechanisms[52] and attack vectors for malware to target. The diversity and flexibility of these platforms, which has given them such utility, has also resulted in substantial difficulty for the examination of operational systems for compromise or vulnerabilities in configuration.

Because of this immense potential for attack, along with the prevalence of PCs across all types of environments, PCs are by far the most targeted platforms by malware. It is critical to carefully configure and monitor for changes in file system, configuration, and network state for each trusted system on the network.

7.1.1 Windows

While attacks have been on the rise for all OSs, attacks targeting Windows are by far the most widespread. According to Symantec, 317 million new malware variants appeared in 2014, up from 252 million new variants in 2013. [53] The staggering number of malware samples is a direct correlation to the number of users available for the attacker to target. The incentive for the vast majority of attackers is the cash flow generated by infecting victims with crimeware, such as distributing ransomware or serving paid advertisements, to victims through successful infections.

Another core concern for operational environments running Windows is the impact of more common commodity and criminal malware that could disrupt operations. This is of a particular concern to ICS/SCADA systems, because a single piece of untargeted malware, such as CryptoLocker, could put a Windows-powered HMI out of commission just as effectively as a PC.

Windows users must take particular care to keep up with updates both to the OS and individual software installed on their devices. Further, up-to-date antivirus and good system hygiene (avoiding administrator accounts, configuration of software firewall, application whitelisting, etc.) is of particular importance to protecting Windows machines. With the large numbers of attackers eyeing the platform, it is important to keep the system hardened against those seeking to utilize vulnerabilities, both new and old.

[52] https://isc.sans.edu/diary/Wipe+the+drive+Stealthy+Malware+Persistence+Mechanism+-+Part+1/15394 - Examples of persistence establishment in Windows through less common features of the operating system

[53] https://www4.symantec.com/mtginfo/whitepaper/ISTR/21347923_GA-internet-security-threat-report-volume-20-2015-social_v2.pdf - Symantec ISTR 2015, Page 90 discusses new malware variants between 2013 and 2014

7.1.2 OS X

OS X has only recently seen its first substantial number of malware attacks in the past 4 years. [54] This is in part due to both core system differences with Windows devices, as well as what is still a comparatively burgeoning user base. In recent years, the number of users who are changing platforms to OS X has increased the allure of the platform for malware authors.

Driven by the increase in the number of users for the OS X ecosystem, malware authors have adapted the same tactics used against Windows to the OS X ecosystem. While the infection vectors and vulnerabilities are often different from Windows devices, the ultimate goal is the same. OS X users are slowly adapting the same tactics for malware design from botnets[55] to ransomware. [56] Many creators have tailored their new attacks to work along with the same existing infrastructure as the current Windows malware, allowing structures, such as botnets, to work cross platform for larger scale DDoS attacks or bitcoin mining.

OS X users must take care not to underestimate security of their systems. It is currently less common to find OS X malware than Windows malware, and the application model employed by OS X will make porting existing malware to the new platform difficult. However, with increasing numbers of users and a different target audience than the other platforms, there is substantial incentive for attackers to continue their increased focus on OS X going forward.

Good practice for security of OS X devices on your network is very similar to Windows. IT administrators should use AV for detection of malware on the device, even if it is less common. Kaspersky labs reported that threat actors targeted the average Mac user nine times during the course of 2014,[57] showing that the number of attacks, while not extreme, is nonetheless significant. Keep in mind that only one successful attack is necessary to compromise a device. Keeping systems up to date, scanning for malicious software, and taking care with application installation and machine configuration will help to mitigate the increasing risk of attack.

7.1.3 Linux

Linux has firmly secured its place in a massive variety of locations, from a host of embedded devices to supercomputers and smartphones to servers. Profiling Linux, as such, can be much more difficult, because the number of differing distributions is massive. Because of the nature of Linux as an open source platform, it is impossible to determine definitively the total number of distributions at any given time, but DistroWatch.com tracks 804 distinct distributions as of November 2015, a number that grows constantly. [58]

Attackers going after Linux machines are often targeting devices with specific hardware, software, or configurations rather than seeking out large-scale infection on any machine. Attacks against servers have been seen as extraordinarily effective, such as the New China botnet attack[59]—a piece of malware that infects Linux devices and uses them as large-scale DDoS machines. In this situation, a single-owned server

[54] https://eugene.kaspersky.com/2014/09/29/the-evolution-of-os-x-malware/ - Article detailing the early ages and progress of OS X malware
[55] https://securelist.com/blog/incidents/32661/flashfake-mac-os-x-botnet-confirmed-25 - Botnet targeting OS X
[56] https://blog.malwarebytes.org/fraud-scam/2013/07/fbi-ransomware-now-targeting-apples-mac-os-x-users/ - Ransomware targeting the OS X platform
[57] https://securelist.com/analysis/kaspersky-security-bulletin/68010/kaspersky-security-bulletin-2014-overall-statistics-for-2014/ - Kaspersky 2014 year in review, detailing OS X attacks encountered by observed users
[58] http://distrowatch.com/search.php?status=All - Collection of Linux distributions tracked by DistroWatch.com
[59] http://blog.malwaremustdie.org/2014/09/mmd-0028-2014-fuzzy-reversing-new-china.html - Malware analysis on New China Botnet

of reasonable quality with the malware is able to output massive DDoS attacks because of the server's often high level of network throughput, outclassing the potency of traffic that could be generated by more compromised individual machines. [60,61]

Because the desktop user base of Linux has remained relatively small compared to both Windows and OS X, attacks on the platform can be hard to track down, hard to profile, and come into the spotlight of malware analysts slowly. However, because of a lack of interest from the consumers, many anti-malware companies have focused more on Windows and OS X. It has become more difficult to remediate malware on Linux than other platforms because of the lack of detection mechanisms.

Linux users must take the time to carefully configure their device, take care with the packages they choose to install, and watch for unexpected application or script behavior. While attacks targeting Linux are less common, having a false sense of platform security results in less vigilant security on the part of the users. In recent years, there have been major vulnerabilities in active use for Linux, including Heartbleed[62] and Shellshock.[63] Each of these vulnerabilities was remotely exploitable and posed major risk to users of Linux platforms. The malware risks and techniques employed against Linux targets closely mirror those used against other platforms, such as ransomware[64] and remote access trojans.[65] While uncommon, Linux malware has become more popular in recent years because of the heavy reliance on the platform across a great number of devices, and it is continuing to grow. [66]

7.2 Mobile

Mobile devices and the bring-your-own-device (BYOD) movement as a whole pose a security risk to both individual privacy and organizational integrity. It is has become commonplace to use mobile devices to access a plethora of services, both within company intranet and the public Internet, bringing a wealth of information onto these relatively new platforms. Further, ownership of these devices is quickly becoming ubiquitous across the country, with 64 percent of American adults owning smartphones as of April 2015 compared to 35 percent in 2011. [67] These devices are providing an extensive amount of personal and corporate information conveniently located in one place—for both the user as well as the attacker.

From the perspective of many attackers, compromise of these devices is likely as valuable as PCs, if not more so. The value of a device for many attackers is in what information they can gather from it, as well as the network devices to which it can connect. Many users of smartphones store an incredible amount of personal information on their devices, accessing services such as workplace and personal email, online banking, and social media.

Further muddling the issue surrounding the value of a compromised mobile device, key assets are not always easy to identify from the applications on the device alone. As an example, many services allow users to utilize two-factor authentication by sending a second authentication code via text message or mo-

[60] http://arstechnica.com/security/2015/09/botnet-preying-on-linux-computers-delivers-potent-ddos-attacks - Article detailing the strength of DDoS attacks from Linux servers from the Linux/Xor.DDoS attack

[61] https://bartblaze.blogspot.com/2015/09/notes-on-linuxxorddos.html - article detailing the Linux/Xor malware

[62] http://dl.acm.org/citation.cfm?id=2663755 - The Matter of Heartbleed, ACM - discussion of the Heartbleed vulnerability and consequences

[63] https://www.symantec.com/connect/blogs/shellshock-all-you-need-know-about-bash-bug-vulnerability - Symantec overview of the shellshock vulnerability and consequences

[64] http://blog.checkpoint.com/2015/04/20/analyzing-magento-vulnerability/ - Analysis of magneto ransomware targeting linux users

[65] https://threatpost.com/new-evasion-techniques-help-alienspy-rat-spread-citadel-malware/112064 - Kaspersky analysis of AlienRAT, a remote access trojan targeting a wide variety of platforms

[66] http://www.welivesecurity.com/2015/01/13/really-need-antivirus-software-linux-desktops/ - Evolution of Linux malware

[67] www.pewinternet.org/2015/04/01/us-smartphone-use-in-2015/ - US smartphone usage statistics

bile application.[68] This message, in conjunction with the password, improves the security of the login by taking one factor (something you know—your password) and adding another (something you have—your phone). This increased level of security is useless if the attacker is able to compromise the mobile device because both pieces of information pass through the same medium. This particular case makes the mobile device a more attractive target to an attacker than the victim's PC, because one device yields all the required credentials.

From an operational security perspective, the comparison of mobile to traditional desktop/laptop OSs is a mixed bag. On one hand, modern mobile OSs, such as Android and iPhone OS (iOS), have had a higher degree of security since their inception than have their PC counterparts. Both Google and Apple heavily monitor their application stores for malicious applications and remove developers who repeatedly put their customers at risk. Furthermore, the systems themselves impose heavier constraints on running applications and utilize sandboxing along with other techniques to prevent many forms of malware from easily executing on the system without the user knowing. As a result, exploits are less common, particularly when the user installs only officially sanctioned applications.[69]

However, one core issue with the transition to mobile OSs is that they are still young and evolving ecosystems and not fully tested by the information security and malware research communities. Because of the short time in which these devices have been commonly in use, there has only recently been a large enough user-base to incentivize malware authors to target them and researchers to examine the attack surface.

Just as OS X has started to see a rise in malware through recent years, Android and iOS have also recently seen an increase in attention from malware developers. Security, malware, and forensic analysis on these devices will become an increasing concern for researchers over the coming years as the potential gain for attackers continues to grow.

When employing mobile devices in the corporate setting, administrators need to treat these devices similar to laptops. Asset owners must monitor all devices used between networks and exposed to the open web for malware and should not trust such devices in information-sensitive environments.

7.3 Internet of Things

As devices connecting to the Internet become ubiquitous, the concern of the malware community turns to the new risks that the compromise of these devices poses. Advancement in technology is often moving at a faster pace than security, because getting to market is often the primary concern of new technology. This can potentially leave customer data (and, in the case of automated tools, privacy or safety) at risk.

The acceleration of new types of devices connecting to the Internet is higher than ever, increasing the potential for the infection of new and emerging technology and making it a greater concern in areas outside the realm of conventional computing devices. Smart appliances, home security systems, closed-circuit televisions, vehicles, and public transportation are all consistently pushing for further automation and connectivity and are becoming accessible through the Internet. These changes are adding convenience and new features, but are also increasing the attack surface and drastically changing the security perimeter.

[68] https://securelist.com/blog/research/73211/the-asacub-trojan-from-spyware-to-banking-malware - Asacub malware focused on stealing multifactor authentication from Android banking applications
[69] https://www.f-secure.com/documents/996508/1030743/Threat_report_H2_2013.pdf - Mobile malware statistics based on software download location

Recent years have seen multiple proof-of-concept attacks against these devices by researchers, such as taking control of a car while it was in motion on the highway,[70] modifying smart watches,[71] and intercepting baby monitor transmissions.[72] With such a wide array of newly emerging technology, there will need to be significant research taking place on both the defensive and offensive sides before we know how these devices will ultimately fit into our lives and what it will reasonably take to secure them.

Looking outside of the scope of personal devices, point-of-sale devices and ATMs are communicating through the Internet to establish connections to financial institutions and private money-transfer corporations. This mirrors the concerns of home devices with the primary difference being the type of information exfiltrated in a compromise.

7.4 ICS

Since the discovery of Stuxnet, a widespread emergence of interest has come from highly sophisticated actors: intelligence gathering and potential attacks targeting ICS environments. Two malware families, Havex and BlackEnergy 2, were observed specifically targeting ICSs. Both make use of a modular architecture allowing for extended functionality with plugins.

Havex (also known as Energetic Bear, Crouching Yeti, and Dragonfly, among others) is a RAT (remote-access trojan) that initially targeted the Energy Sector in early 2014. Two observed plugins actively collect information about connected ICS devices by scanning for several commonly-used ICS protocol ports and interrogating any available Open Platform Communications servers for information. Havex has been spotted in a variety of critical infrastructure sectors, including Chemical, Energy, and, more recently, the pharmaceutical industry,[73] throughout various countries.

Intelligence gathering on the affected systems appears to be the underlying goal of Havex, rather than directly controlling any connected control systems. This indicates that the current goal for its creators is to steal information about the system's configuration, possibly in anticipation of an attack scenario, or possibly related to corporate intellectual property.

BlackEnergy is an interesting case of malware that has undergone a dramatic change in its design and target depending on the groups that use it. Initially, BlackEnergy was a DDoS bot primarily used by the Russian hacker underground to take down sites. Support for plugins was added in the next major revision (BlackEnergy2),[74] changing the exclusively DDoS box into a powerful multi-tool. Years later, researchers discovered that threat actors utilized zero-day exploits and spear phishing, combined with BlackEnergy 2 and specially-tailored plugins, to target and compromise ICS networks.

Threat actors have also found methods of avoiding detection from firewall services through the exploitation of third parties. BlackEnergy 2 included functionality for fetching configuration from third-party sites that are otherwise not malicious but are exposed to the public (such as Google Plus). This advancement may have helped to keep the malware authors "under the radar" until recently while executing the campaign. In addition, a Linux variant of BlackEnergy2 targeting ARM and MIPS[75] platforms has been seen in the wild—something that is uncommon for malware. New techniques to embed their communication software into internal company routers through backdoors and zero-day exploits has greatly expanded their capability to penetrate and potential to persist in both corporate and ICS networks.

[70] www.wired.com/2015/07/hackers-remotely-kill-jeep-highway - Proof of concept attack against a vehicle's connected systems

[71] https://securelist.com/blog/research/69369/how-i-hacked-my-smart-bracelet - Research into the pairing and authentication mechinisms of smart wearable devices.

[72] https://www.rapid7.com/docs/Hacking-IoT-A-Case-Study-on-Baby-Monitor-Exposures-and-Vulnerabilities.pdf - Case study on Internet of Things devices

[73] www.blog.trendmicro.com/trendlabs-security-intelligence/64-bit-version-of-havex-spotted - Details new variants of Havex, its targeting trajectory, and method of operation

[74] https://ics-cert.us-cert.gov/alerts/ICS-ALERT-14-281-01B - ICS-CERT alert for BlackEnergy 2

[75] https://securelist.com/blog/research/67252/be2-custom-plugins-router-abuse-and-target-profiles/ - Kaspersky report detailing BlackEnergy target platforms

Both Havex and BlackEnergy show a high-degree of sophisticated design and careful construction, leading researchers to believe that they are being maintained by a team of developers, expanding their capabilities and advancement in a way that has drawn a great deal of attention from researchers. Attacks from the BlackEnergy family are still ongoing, with attackers using a variant of BlackEnergy 3 against the corporate network of a power facility as recently as December 2015. [76]

Threats to ICS systems are a growing area of research for both attackers and defenders. With continuing threats and discovery of vulnerabilities in this area, it will remain a high profile topic in the security field going forward due to the potential impact of a breach.

8. CONCLUSION

The malware landscape has grown in parallel with software and emerging technology, adapting new techniques and strategies from industry into their design paradigms and targeting new platforms as they present lucrative opportunity for attackers. The inherent cat and mouse game between malware authors and vendors that has existed for years with no sign of stopping.

Administrators across all industries need to protect their devices from malware attacks, focusing their efforts on keeping device OSs and security software up to date and hardening their infrastructure against open vectors of attack. The threat of data breaches continues to rise, but infrastructure security solutions are evolving and adapting as well. Staying involved with the state of malware, new patches, and security advancements helps to keep network defenders aware of any new challenges they may face. Knowledge of the landscape of the network, along with awareness of the new and emerging threats, allows administrators to take the steps they need when dealing with attacks against their networks.

ICS-CERT publishes vulnerability updates and malware analysis reports for customers in the ICS industry through the Homeland Security Information Network (HSIN).[77] Regular alerts[78] and advisories[79] are available to aid in the protection of critical devices, allowing administrators and users to take timely action when new vulnerabilities arise. ICS-CERT publishes best practices[80] for security configurations to aid in the protection of assets before a compromise has occurred. Each malware family that poses a threat to ICS systems goes through analysis in order to provide remediation techniques and with malware detection signatures. This helps asset owners detect malicious software on their networks and defend against attacks. For updates and analysis on the most current threats facing ICS, visit https://ics-cert.us-cert.gov.

[76] https://ics-cert.us-cert.gov/alerts/IR-ALERT-H-16-056-01 - ICS-CERT alert for BlackEnergy 3 discovery within an ICS environment

[77] https://hsin.dhs.gov/

[78] https://ics-cert.us-cert.gov/alerts - ICS-CERT alert page for newly released alerts on vulnerabilities and malware families

[79] https://ics-cert.us-cert.gov/advisories - ICS-CERT page for newly released advisories on vulnerabilities and malware families

[80] https://ics-cert.us-cert.gov/Recommended-Practices - ICS-CERT best practices page for securing and monitoring systems in ICS environments

Department of Homeland Security

Office of Cybersecurity and Communications

National Cybersecurity and Communications Integration Center

NCCICCustomerService@hq.dhs.gov

1-888-282-0870

Industrial Control Systems Cyber Emergency Response Team

https://ics-cert.us-cert.gov