CIS CSC Controls vs. Ransomware: An Evaluation

Dylan Malloy

CIS CSC Controls vs. Ransomware:

An Evaluation

Author: Dylan J. Malloy, dylanjmalloy@gmail.com

GIAC GSEC Gold Certification

Advisor: *Tanya Baccam*

Accepted: April 19, 2021

Abstract

Cybercriminals continue to develop and enhance both new and existing ransomware variants, exploiting vulnerabilities to compromise computer systems and wreak havoc on individuals and organizations. Ransomware, while everchanging, typically relies heavily on a lack of controls in place for it to be promptly stopped or eradicated; however, many controls set out to reduce the overall impact of ransomware, if not stop it entirely. Organizations often try to protect themselves from ransomware by investing money into their security stack, Anti-virus, Endpoint Detection and Response, and Host Intrusion Prevention System. However, these tools will not be nearly as effective without the proper controls to align their functions. Implementing CIS Critical Security Controls can significantly reduce the impact of ransomware, or even potentially stop it in its tracks, meaning minimal disruptions to operations.

1. Introduction

Our modern society is tuned in, dialed up, loaded in, and inter-connected like never before. The growing ubiquity of the internet has drastically changed the world in the last 20 years, to the point that the pre-digital age is now almost as strange and unknown to youth today as the Iron Age. Thanks to the World Wide Web, we now have the collective knowledge of humanity at our fingertips, and the power to answer almost any question.

However, data does not equal wisdom, and society has not maintained the ideals of the early internet as a place of positivity and collective sharing of knowledge. Criminals have leveraged the digital world to their advantage, developing methods of cyberattack known as ransomware to demand money from individuals and institutions. And in today's ultra-connected digital world, the matter of a ransomware outbreak on a particular system is not a matter of if, but when. As cybercriminals become more sophisticated in their methods of attack, more and more fall victim. Several recent, high-profile ransomware campaigns made global headlines, such as the Hollywood Presbyterian Medical Center (Locky) attack, the WannaCry (WCry/WannaCry) attack, and the Petya Outbreak (Petya).

Fortunately, the bad actors behind these events are not the only ones advancing and developing their methods. Organizations such as the Center for Internet Security (CIS) also work tirelessly to respond to developing threats, evolving and enhancing the defenses of the digital world.

1.1. Overview of Past Ransomware Incidents

It is impossible to truly count or know the total number of ransomware attacks, successful or otherwise, since the advent of the internet; this number is certainly in the millions. Several notable examples stand out in recent memory, however. In February 2016, the Hollywood Presbyterian Medical Center (HPMC) suffered an attack from adversaries who were leveraging the Locky ransomware, delivered through email as a Word document with malicious macros. The adversaries were asking for forty Bitcoins, the value of which was estimated at around seventeen thousand dollars ($17,000) at the time. As a result of this attack, HPMC temporarily took some of its IT systems offline.

After several days, HPMC decided to pay the ransom, and continue to work with authorities (Bisson, 2016).

In May 2017, the WannaCry ransomware outbreak impacted several high-profile targets worldwide, such as the National Health Service (NHS) in the United Kingdom (UK) and Telefonica, an Internet Service Provider (ISP). This targeted attack involved malicious code that aggressively sought out new endpoints to compromise, moving through networks at lightning speed, and providing a chilling reminder for the reason behind ubiquity of epidemiology terminology in cybersecurity: virus, host, infect, outbreak. The cybercriminals behind the WannaCry attack demanded three hundred dollars in Bitcoins, per impacted machine, from thousands of organizations throughout their campaign of aggression (Bisson, 2016).

In April of 2016, a new strain of ransomware called Petya appeared. Petya did not encrypt the files on compromised machines, which made it different from most ransomware seen until then; instead, it simply made the entire hard disk inaccessible. Petya caused endpoints to immediately display the blue screen of death (BSOD), along with the ransom note, before the operating system would be allowed to load by the malicious code (KnowBe4, 2016).

1.2. Brief Overview of CIS CSC

The Center for Internet Security, or CIS, is a community-driven nonprofit organization that has spent years developing cybersecurity controls that prioritize actions that execute a defense-in-depth strategy to reduce and mitigate the impact of potential cyber-attacks. These controls break down into different implementation groups, depending on a particular organization's maturity, resources, and abilities (CIS, 2019). The CIS Critical Security Controls for Effective Cyber Defense (CSC) is a publication of these best practice guidelines for computer security, and is also known as the CIS CSC, CIS 20, CCS CSC, SANS Top 20, or CAG 20.

When implemented correctly, the CIS Controls can significantly mitigate a ransomware's total impact by improving an organization's cyber hygiene through data recovery; limitation and control of ports, protocols, and services; malware defense; and other best practices.

2. CIS Critical Security Controls

The CIS Critical Security Controls are a prioritized, prescriptive set of cybersecurity defensive actions and best practices (ManageEngine, n.d.). There are a total of twenty CIS Critical Security Controls that are each categorized into additional Sub-Controls. The following controls were selected for testing:

- Controlled Use of Administrative Privileges
- Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers
- Malware Defenses; Limitation and Control of Network Ports, Protocols, and Services
- Data Recovery Capabilities.

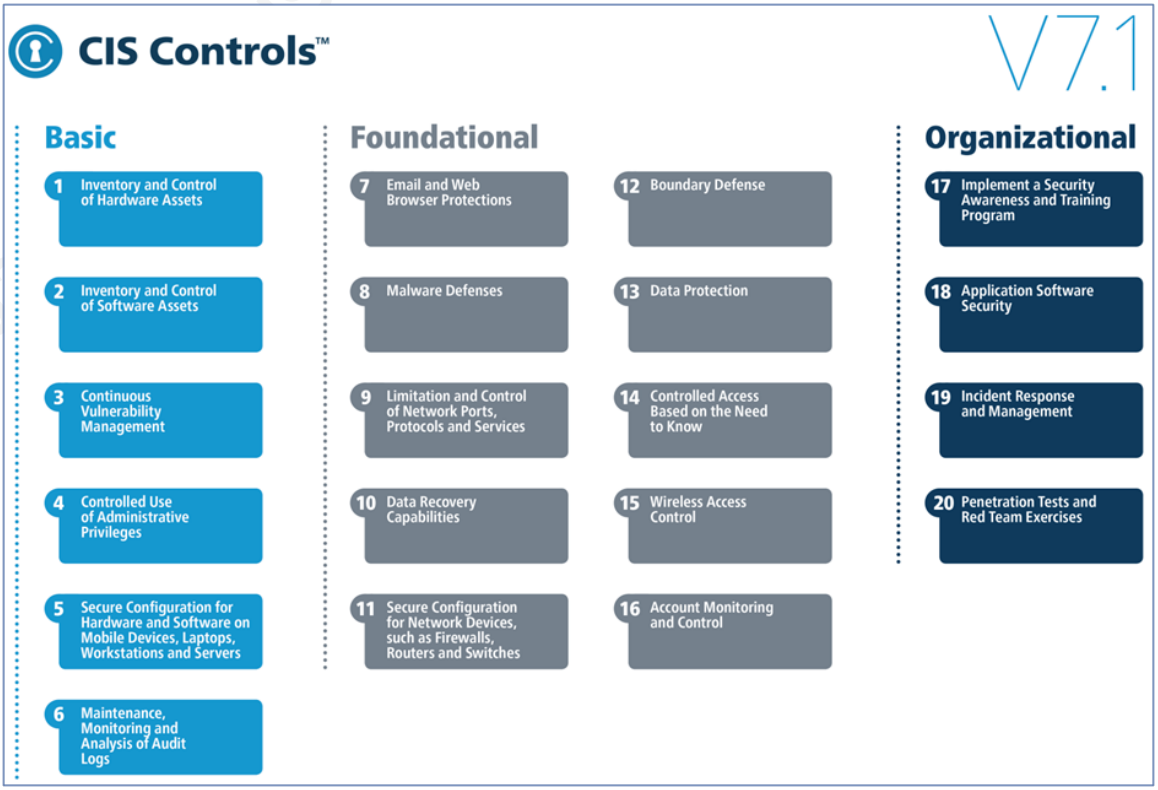

Figure 1. CIS CSC Controls (CIS, 2019)

These controls were chosen because they lined up with the testing environment: a Windows 10 base install, with the latest updates applied, and no additional configuration changes made beyond implementing the CIS controls. It is important to note the additional CIS controls that were attributed to defense against ransomware, but that did not line up with the goals of this experiment:

- Continuous Vulnerability Management
- Inventory and Control of Software Assets
- Email and Web Browser Protections; Boundary Defense
- Implement a Security Awareness and Training Program
- Penetration Tests and Red Team Exercises.

While a case could be made for controls not mentioned above, these controls all have at least one Sub-Control that would work against typical ransomware tactics. Continuous Vulnerability Management was removed from testing, as it ultimately did not line up with the testing scenario and environment.

2.1. Overview & History

The Center for Internet Security (CIS) (established in October 2000) set out to make the connected world a safer place by establishing best practices for securing information technology systems and data against cyber threats. Since CIS is community-driven, the experts who help develop the CIS Controls come from various sectors, such as education, government, healthcare, retail, and more. CIS recognizes that all of the competing technology and solutions at our disposal can cause a "fog of more" for users, overwhelming them with competing technologies, priorities, and solutions (Sornson, 2020). Working as a community, CIS identifies the issues that exist and focuses on those defensive steps that have the most significant value. CIS states its defining value as "knowledge and data–the ability to prevent, alert, and respond to the attacks that are plaguing enterprises today" (CIS, 2019).

2.2. Development & Implementation Groups

The CIS Controls were derived from different attacks and defensive strategies leveraged against the law-abiding online community. The goal is not only to block an

attach before its initial stages, but also to provide the mechanisms to detect endpoints that may have already been compromised, and/or slow down the process of attack. The controls reflect five aspects critical to an effective cyber defense model: offense informs defense, prioritization, measurements and metrics, continuous diagnostics, and mitigation and automation. This report will explore how these five aspects are supported by the CIS during its analysis of the effectiveness of the controls against the ransomware-like PowerShell scripts (CIS, 2019).

Historically, CIS controls focused on the order of their controls as a means of implementation, which they dubbed "cyber hygiene." It was found that some of the controls were difficult for organizations to implement if they had limited resources (CIS, 2019). This difficulty showed a need to better distribute the controls and add greater flexibility to their implementation, in order to provide best practices that were relevant to organizations based upon their available capabilities.

CIS was able to meet this need by adding CIS Controls Implementation Groups, abbreviated as "IG#." Based on an organization's size, these groups contain critical sub-controls that indicate the best cyber hygiene practices for small businesses (around ten employees, IG1), regional corporations (IG2), and large corporations with thousands of employees (IG3) (CIS, 2019). Once an organization meets these sub-controls, they can look towards the next implementation group to address the most significant risk(s) specific to their environment. In this experiment, while testing the controls themselves, IG3 was the focus; however, not every sub-control was implemented for every control tested, as some did not add value to the experiment.

CIS divides their twenty controls into three tiers: Basic, Foundational and Organizational. For this experiment, the controls that were selected for testing fell into the Basic and Foundational categories. During this experiment, the following controls were tested: Controlled Use of Administrative Privileges; Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers; Malware Defenses; Limitation and Control of Network Ports, Protocols, and Services; and Data Recovery Capabilities.

While other CIS controls also contribute to the defense against ransomware, the controls selected line up best with the proposed testing environment, because their sub-controls had the greatest potential impact for the prevention or mitigation of ransomware. For example, although continuous variability management has its uses, it does not make sense on a base install of Windows 10. The proposed environment was a Windows 10 base install, with the latest updates applied and no configuration changes made other than the specific CIS controls being tested. Locking down administrative privileges may limit the ransomware's ability to be downloaded, installed, or executed. Through the implementation of a secure configuration, the effectiveness of a CIS hardening baseline and any potential gaps will be observable. Malware Defense is essential for endpoint defense. Measuring the control's ability to prevent malware will be critical for this experiment. Ransomware will often attempt to phone home; limiting and controlling ports and network services may stop its ability to reach back and cause further damage. While data recovery is not a preventative control, testing this control will show the importance of backing up data if it has been deleted, destroyed, or encrypted.

2.3. Tools & Products

Understanding the purpose behind the controls and a recommended approach for implementation is essential. Thankfully, there are tools and products, some free and others paid, which can help attribute to implementing and verifying these controls. Some of these tools were leveraged as part of this experiment: CIS offers CIS-CAT Lite, a free assessment tool that helps organizations implement secure configurations by scanning against CIS Benchmarks. They also offer CIS-CAT Pro, a paid version providing additional features such as automation, reporting, and vulnerability scanning. Windows Defender is an excellent Malware Defense solution that is included with Windows and offers enterprise management. Veeam is a robust backup solution that can allow the backup and recovery of many different workloads, from physical to virtual, offering both a free and paid solution to satisfy Data Recovery and its sub-controls.

3. Ransomware Deep-Dive

3.1. Detailed Description of Ransomware

3.1.1. Overview

Ransomware is a form of malware that falls into one of two categories, based upon distinct attack approaches: crypto ransomware is designed to encrypt a system, rendering the files and data unusable; while locker ransomware will lock the end user out from the system and present them with a ransom note (and, in most cases, will not encrypt the data). The adversary or malicious actor then demands a ransom, usually in the form of a cryptocurrency called Bitcoin, in exchange for the decryption key. Without the encryption key, the only way to recover the data is through a backup. These adversaries will often threaten to sell, leak, or destroy the data they have encrypted and effectively compromised if the ransom is not paid within a specific time range. One particularly scary part of all this is that these cyber criminals do not always uphold their end of the bargain just because their demands have been met. In recent years, ransomware attacks have been on the rise, targeting people, organizations, and governments worldwide (Savage, Coogan, & Lau, 2015).

3.1.2. History

Ransomware has a lengthy history, with the first well-known case dating back to 1989. The first ransomware virus was created by Joseph Popp, an evolutionary biologist from Harvard also known as the father of ransomware (KnowBe4, n.d.). Popp's ransomware virus was called the AIDS Trojan, also known as the PC Cyborg. The AIDS Trojan/PC Cyborg was distributed via floppy disks to attendees of the World Health Organization's (WHO's) international AIDS conference (KnowBe4, n.d.). The ransom demanded the victims to send $189. The AIDS Trojan/PC Cyborg virus was an early generation of ransomware, and therefore used simple symmetric cryptography that was decrypted soon after its release (KnowBe4, n.d.; Savage, Coogan, & Lau, 2015). Around 2006, ransomware started to gain traction as cyber-criminal adversaries began to experiment with crypto-ransomware (Savage, Coogan, & Lau 2015). In 2006, the Archieveus Trojan was the first ransomware virus to use RSA encryption: "The Archieveus Trojan encrypted everything in the /MyDocuments directory and required

victims to purchase items from an online pharmacy to receive the 30-digit password" (KnowBe4, n.d.).

In 2008, Bitcoin was released, ushering in the age of cryptocurrency. This provided a whole new platform on which malicious actors could collect payments in a much more anonymized and difficult-to-trace fashion. As a result of crypto-technology continuing to advance, and the advent of more and more anonymous payment platforms, approximately 60,000 new samples of ransomware were detected by the end of 2011 (KnowBe4, n.d.). Ransomware attacks have continued to wreak havoc on society in recent years, growing ever more commonplace. Cybercriminals, meanwhile, have grown bolder and more daring in their offensives, which can be seen in the Petya, WannaCry and Locky attacks.

3.2. Ransomware Scenarios

While there are many different mechanisms by which cybercriminals can deliver ransomware to an endpoint, this paper will touch on a few of the more common ones, such as phishing, drive-by downloads, and Remote Desktop Protocol (RDP).

Phishing is the most common method for cybercriminals to spread ransomware. It is accomplished by crafting emails to trick a victim into clicking on a link or opening an attachment containing a malicious file. These malicious files can come in different formats, such as ZIP files, PDFs, and Word documents.

A drive-by download is a mechanism wherein malicious files are downloaded without the end users' knowledge, when they visit a website that has been compromised. Cybercriminals will exploit known vulnerabilities in legitimate websites to either embed malicious code, or redirect the victim to another site, where the malicious payload is delivered in the background.

RDP allows for the remote access and configuration of a system. Leveraging a service called Shodan, cybercriminals can find machines exposed to the internet with RDP and attempt to exploit vulnerabilities against them until they can gain administrative access. In 2017, reputable security companies such as Rapid7 and Shodan published data showing millions of endpoints with RDP exposed to the internet (Challita, 2018; Rudis,

Beardsley, Hart, & Sellers, 2017). Although cybercriminals leverage Shodan for their advantage, it provides valuable data for individuals and organizations regarding their devices exposed to the internet, providing them with insights and knowledge they otherwise may not have.

3.3. Impact of Ransomware

Ransomware can have a devastating impact on its target, whether that is a person or a business. Ransomware will typically encrypt or lockout a system or data from being accessed. The malicious actor will then request payment in the form of cryptocurrency, such as Bitcoins. The impact can range from temporary to permanent loss of sensitive data, disruption of operations, financial losses, and/or damage to reputation. These devastating impacts often lead individuals or organizations to pay the ransom if they cannot recover or restore their data or systems. Unfortunately, dealing with criminals means no guarantees, and payment does not necessarily secure the release of encrypted files or the unlocking of a system. Often, paying the ransom can be perceived as a sign of weakness by the cybercriminal, which can leave that individual or organization prone to further cyber-attacks. Furthermore, just because data has been decrypted or systems have been unlocked, does not mean the infection is gone or the malicious actor has been removed (University of California, Berkeley, n.d.).

4. Discussions

Ransomware has historically been successful in bypassing various security controls in order to compromise an endpoint. This experiment sets out to determine if the CSC Controls are, in fact, a useful measure for preventing, mitigating, or remediating ransomware and ransomware-like scripts from successfully executing on a system. Due to the unreliability of available ransomware samples on GitHub, PowerShell was leveraged to create a script that simulates ransomware characteristics such as the mass modification, encryption, and deletion of files. The following CSC Controls were tested in this experiment:

- *Controlled Use of Administrative Privileges*

- *Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers*
- *Malware Defenses*
- *Limitation and Control of Network Ports, Protocols, and Services*
- *Data Recovery Capabilities*

4.1. Experiment Groups

The original proposal was to execute this testing against two experimental groups. Group #1 was to have a Windows 10 base install, with the latest updates and no configuration changes. Group #1 is the control group, as it simulates the configuration level the average end-user or organization would be leveraging, without specifically implementing any one of the CSC Controls. Originally, Group #2 was intended to be a Windows 10 base install, with the latest updates, and the CSC Controls implemented one at a time. However, a problem arose during execution, as Windows 10 comes with Windows Defender enabled, which precisely relates to Control #8. A workaround was devised with the development of a third test group: Group #3 had a Windows 10 base install, with the latest updates, Windows Defender anti-virus and firewall protection disabled, and the CSC Controls implemented one at a time. This adjustment provided more clarity in the analysis for each of the tested controls.

4.2. Ransomware Samples

In search of ransomware examples, the following GitHub repositories were selected for sampling: UIM-SEC, theZoo, and Fabrimagic72. Between these three repositories, theZoo proved to be the most valuable, as it provided original files of the ransomwares. The ransomware samples selected from theZoo were WannaCry, Petya, Locky, and Thanos. The WannaCry, Petya, and Locky ransomwares were selected because they are well-known attacks familiar to even laypeople. Thanos was selected as it is a more recent ransomware case.

4.2.1. Ransomware-Like Script

As the reliability of these ransomware samples was unknown, this research proposal included the development of a ransomware-like script in Microsoft's

PowerShell. A PowerShell script, developed by Thomas Rayner, was found on the web that simulates ransomware activity. Although there are many ransomware variants and characteristics of malicious code, Rayner developed his script to simulate characteristics that he found to be common in ransomware attacks (2015). These common characteristics include a user modifying more than one hundred files, a user renaming more than one hundred files, and for these two characteristics to happen in under sixty seconds. Below is the original script from Rayner's website (Rayner, 2015).

```
$strDir = "C:\temp\test1\"

GCI $strDir | Remove-Item -Force

1..200 | % { $strPath = $strDir + $_ + ".txt"; "something" | Out-
File $strPath | Out-Null }

Measure-Command { 1..101 | % { $strPath = $strDir + $_ + ".txt";
$strNewPath = $strPath + ".chng"; "changed" | Out-File -Append
$strPath; Rename-Item -Path $strPath -NewName $strNewPath } }
```

The first three lines set up the environment:

- $strDir is the test directory.
- The second line empties the test directory.
- The third line creates two hundred text files in #strDir 1…200.

For each of those numbers, between one and two hundred, the script creates a file and suppresses the output. The fourth line is the ransomware simulation. For 101 files, the script creates a variable $strPath to an individual file created in line three. It also crafts a new path stored in $strNewPath, which is the same file but with an extension. The contents of the file are then modified by writing "changed" inside of them. The final step is to rename the file. The whole thing is wrapped in a Measure-Command block, to see how long the entire process takes to run (Rayner, 2015).

Rayner's script, while written well, was missing the encryption piece proposed as part of this research. New additions to Rayner's script, shown below, add encryption and create a new directory where the script will effectively execute.

```
Mkdir "c:\temp\ransomware\"

(Get-Childitem -Recurse -Path "c:\temp\ransomware\").Encrypt()
```

4.3. CSC Control #10: Data Recovery Capabilities

When cybercriminals compromise endpoints, they will often make configuration changes and alter data, which can jeopardize organizations' information and systems. It can be challenging for an organization to remove an adversary's presence without reliable data recovery capabilities (CIS, 2019).

4.3.1. Experiment Group #1

For Group #1, the control group, Windows updates were applied to the VM, and no further configuration changes were applied. The ransomware depository was downloaded from theZoo on GitHub and downloaded as a compressed file. Upon extraction, Windows Defender immediately started generating alerts for the artifacts it detected, prompting action for removal. However, it was clear that Windows Defender also started the quarantine process after checking a few folders, as seen in *Figures 2–4*, below.

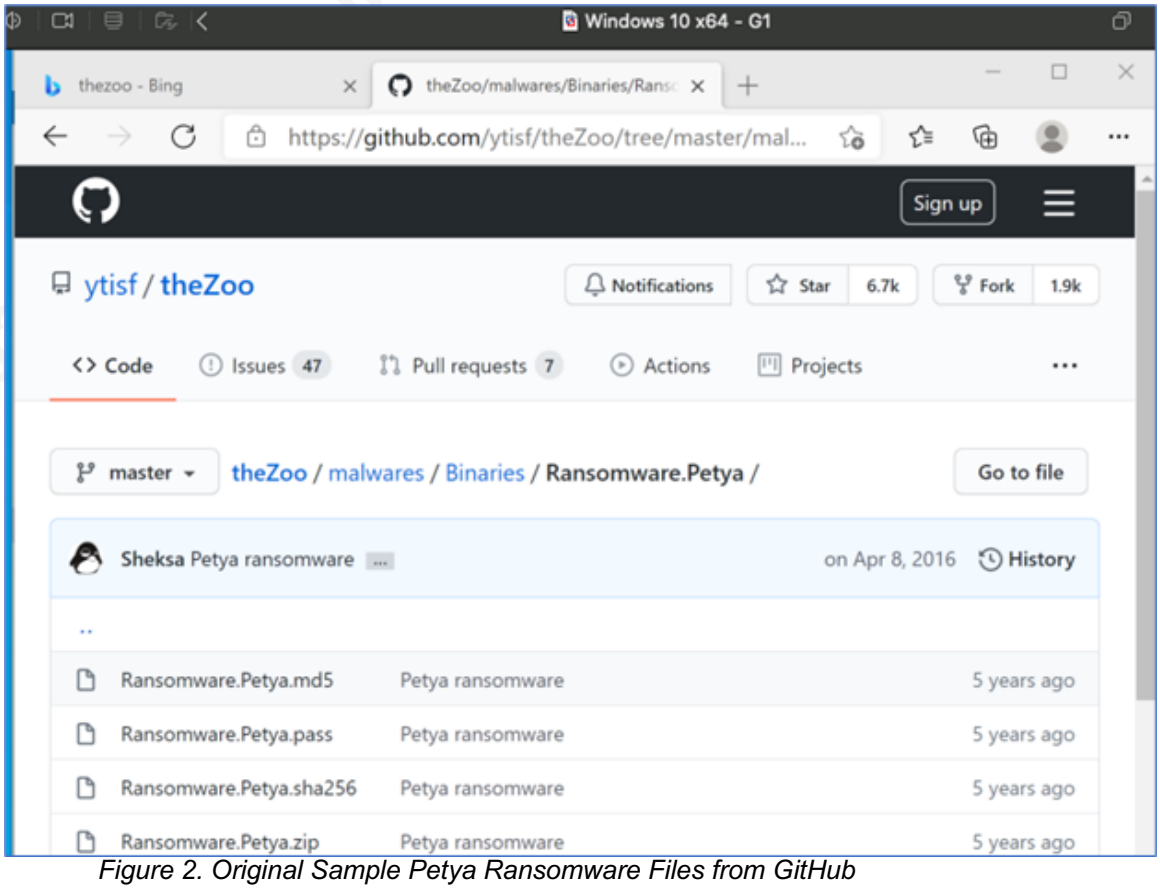

Figure 2. Original Sample Petya Ransomware Files from GitHub

With Windows Defender enabled, these results were expected. They may have differed if the ransomware sample were zero-day-like, but it was a known example, and Windows Defender already had the mechanisms to detect it. The results were subsequently the same for the other ransomware samples sourced from theZoo.

The final step was to test the PowerShell ransomware-like script. As seen in *Figure 4*, below, Windows Defender did not stop the script from running.

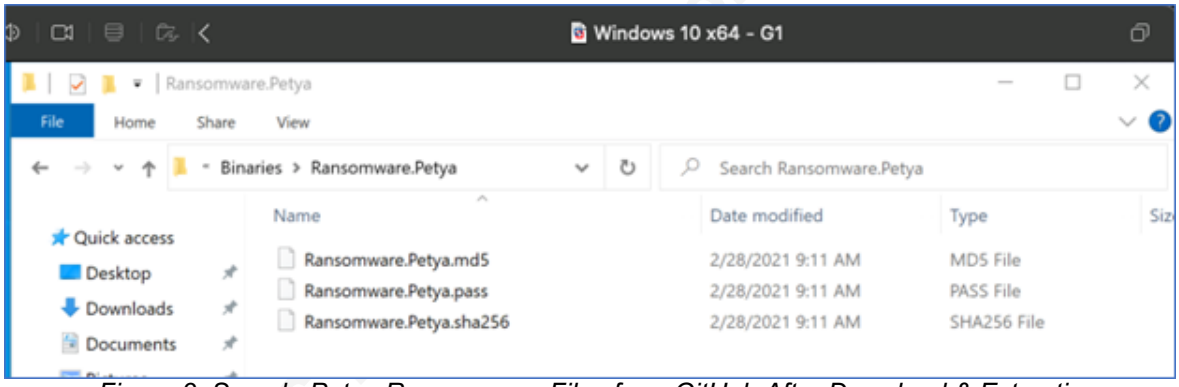

Figure 3. Sample Petya Ransomware Files from GitHub After Download & Extraction

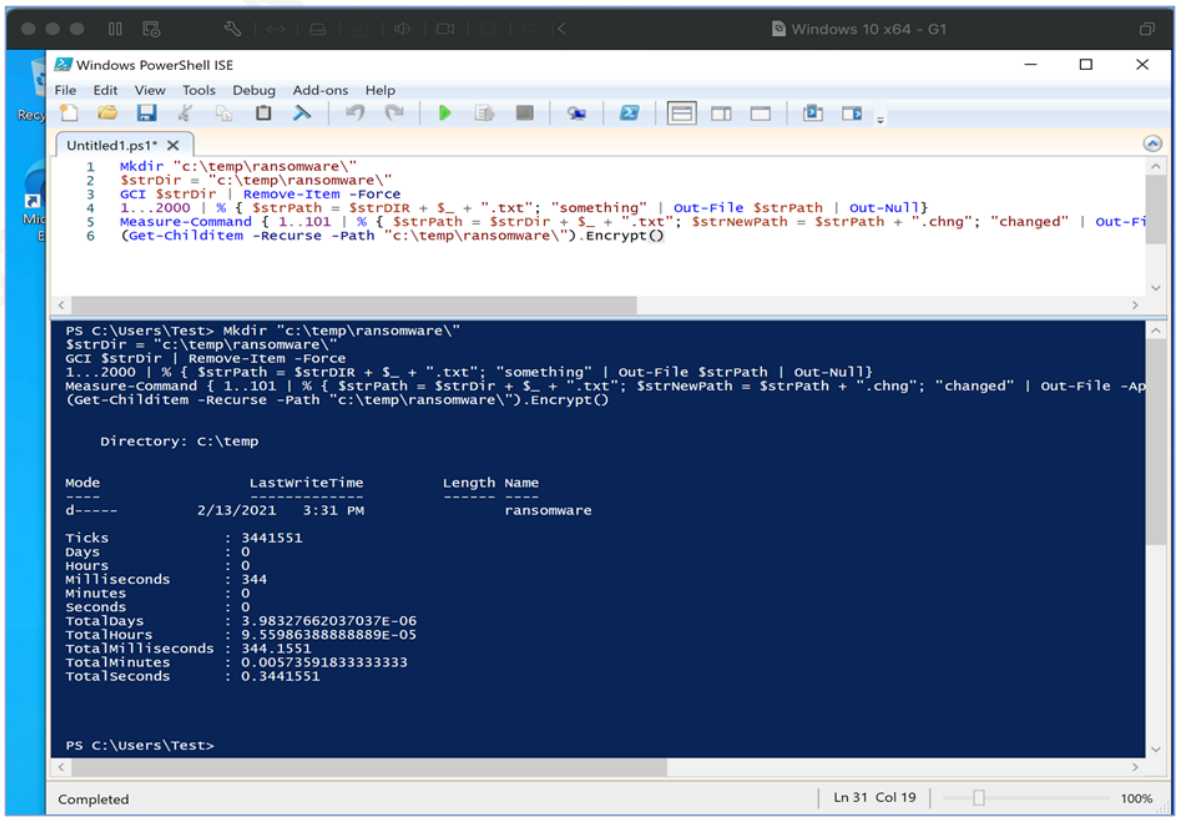

Figure 4. PowerShell Script Executing Successfully

4.3.2. Experiment Group #2

For Group #2, Windows updates were applied (*Figure 5*, with Sub-Controls highlighted in green). In search of a tool to adequately cover these sub-controls, a software called Veeam Backup & Replication was utilized. After installing the Veeam agent, a full backup was created and scheduled for 7:00 a.m. Pacific Time daily to an external hard drive. This step covered Sub-Controls #1, #4, and #5. After capturing a full backup, the next step was to see if the endpoint could successfully restore it. This restore completed successfully, proving the backup taken was in a good state, covering Sub-Control #3. For Sub-Control #2, leveraging VMware Fusion Pro, a Full Clone was created, which would allow for quick recovery of the entire system if needed.

With Control #10 and its respective sub-controls in place, the virtual machine was ready for testing. The ransomware was downloaded from theZoo repository on GitHub in a compressed file format. Upon extraction of the ZIP file, Windows Defender proceeded to generate alerts for each artifact identified and prompted for removal as experienced with the first group. The next test involved the execution of the ransomware-like PowerShell script, which ran successfully, as shown in *Figure 6*, below.

Sub-Control	Asset Type	Security Function	Control Title	Control Descriptions	Implementation Groups 1	2	3
10.1	Data	Protect	Ensure Regular Automated Backups	Ensure that all system data is automatically backed up on a regular basis.	x	x	x
10.2	Data	Protect	Perform Complete System Backups	Ensure that all of the organization's key systems are backed up as a complete system, through processes such as imaging, to enable the quick recovery of an entire system.	x	x	x
10.3	Data	Protect	Test Data on Backup Media	Test data integrity on backup media on a regular basis by performing a data restoration process to ensure that the backup is properly working.		x	x
10.4	Data	Protect	Protect Backups	Ensure that backups are properly protected via physical security or encryption when they are stored, as well as when they are moved across the network. This includes remote backups and cloud services.	x	x	x
10.5	Data	Protect	Ensure All Backups Have at Least One Offline Backup Destination	Ensure that all backups have at least one offline (i.e., not accessible via a network connection) backup destination.	x	x	x

Figure 5. Sub-Controls Implemented for Testing are Highlighted Green (CIS, 2019)

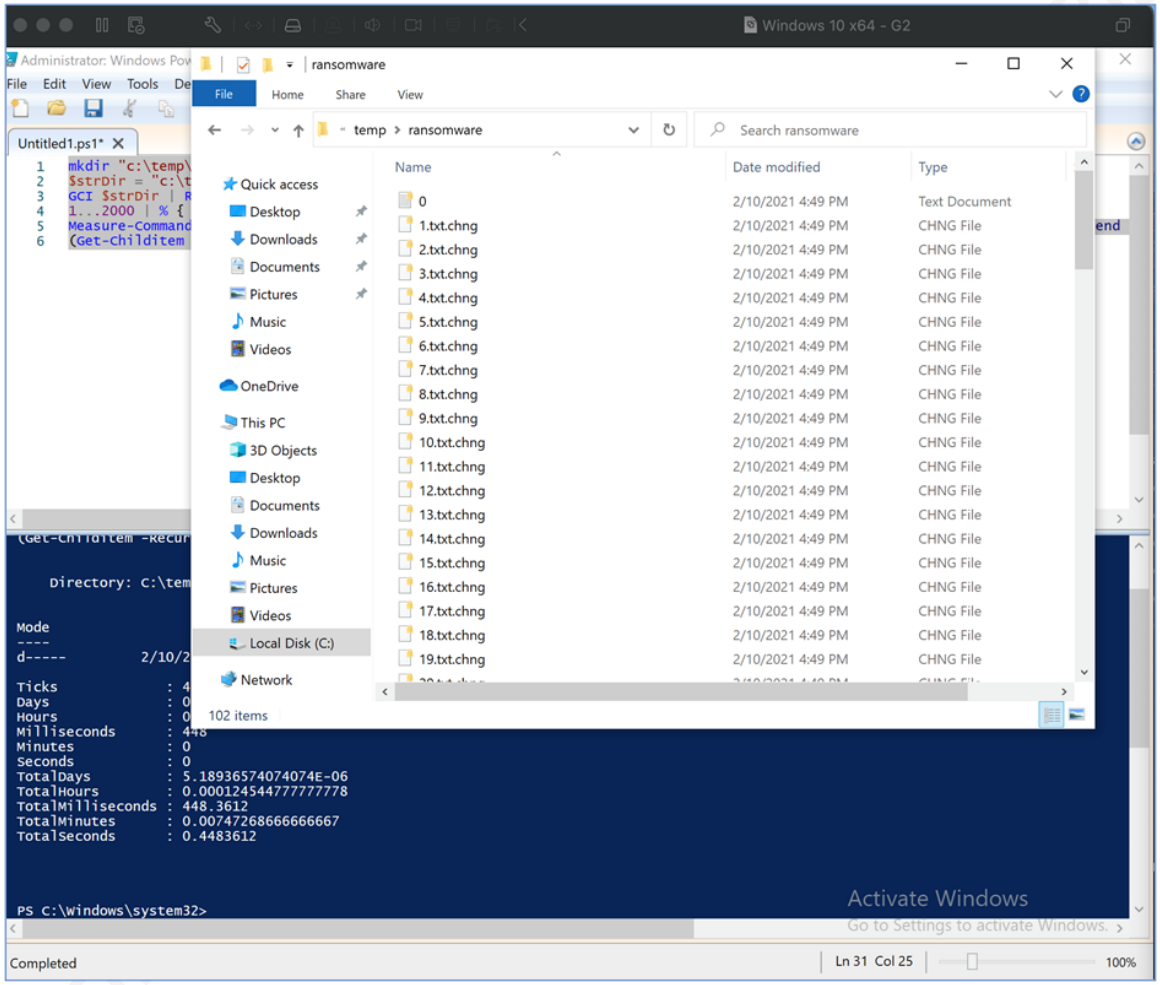

Figure 6. PowerShell Script Executing Successfully

After successful execution of the PowerShell script, there was a new directory on this virtual machine named Ransomware, with files that had been encrypted. The next step was to test the restoration of the endpoint using the full backup that had been taken. After plugging in the external hard drive where the backup was saved, the restore was executed by right-clicking on the Veeam icon in the system tray, as in ***Error! Reference source not found.***, on the next page.

The restore process took approximately ten minutes to complete. Upon completion, the malicious files were no longer present on the endpoint, as seen in ***Error! Reference source not found.***, on the page after next.

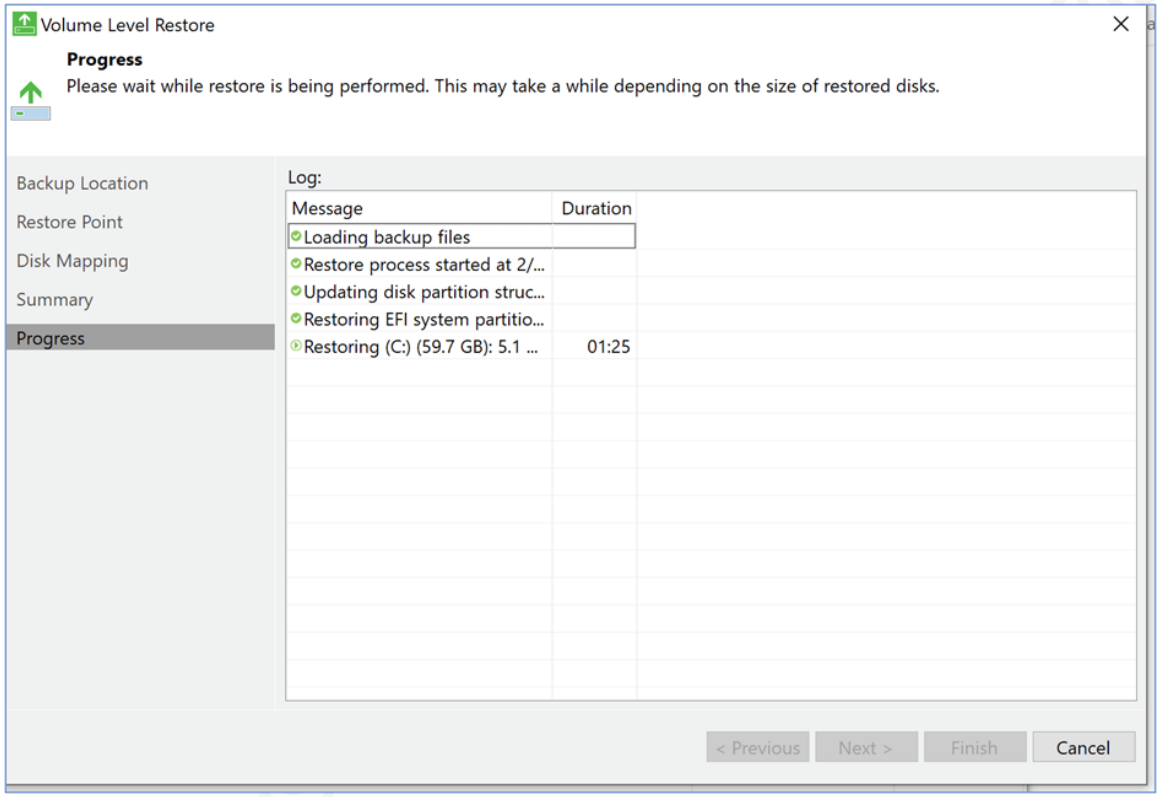

Figure 7. Restoring from Backup Using Veeam

4.3.3. Experiment Group #3

After the first two groups, it became clear that an out-of-box configuration would impact testing, as Windows 10 by default comes with some configuration settings related to CIS controls. To test the full capabilities of the Control #10, Windows Defender and Firewall were disabled. After disabling the anti-virus, Veaam backup software was installed, followed by capturing a full-back up to an external hard drive. For redundancy, a snapshot of the virtual machine was also taken. With two backups taken, it was time to test them to ensure they were captured successfully. Performing a volume restore from Veeam and reverting to the snapshot demonstrated a clean restore point. With these few steps completed, the sub-controls for Data Recovery Capabilities had been satisfied. Now ready to test the control, an example copy of ransomware was obtained from theZoo. The first detonation of ransomware was WannaCry and executed successfully. After a few moments, it was clear the virtual machine was infected with WannaCry, presenting its ransom note requesting payment for decryption, as shown below in ***Error! Reference source not found.***.

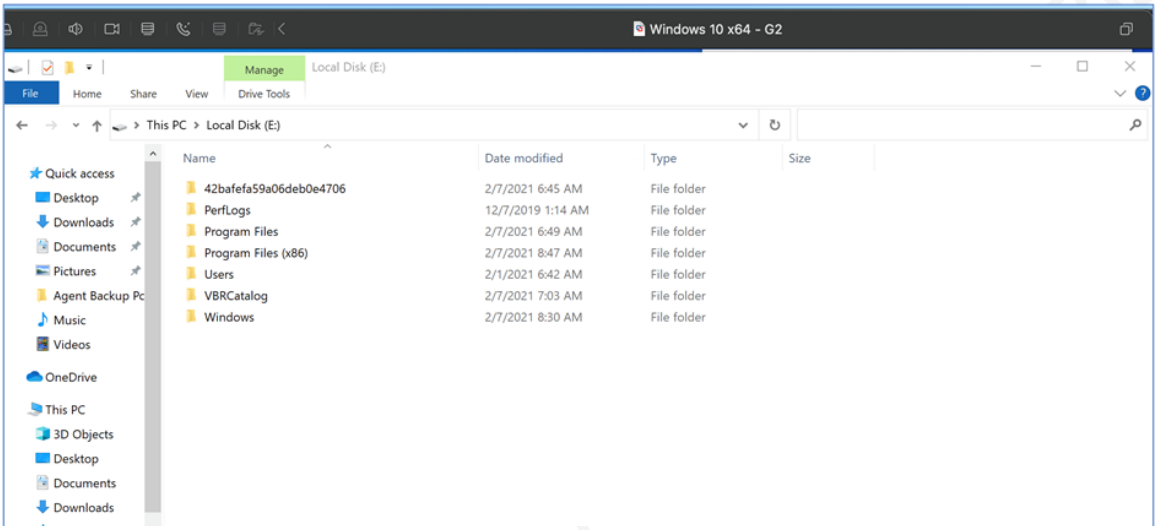

Figure 8. Successful Restore from Veeam Backup

The backup taken from Veeam allowed for a volume restore; however, the ransomware was still present as a restore of the files and structure took place, but did not restore to a point prior to the operating system being infected. Veeam would have allowed for the volume to be restored to a new virtual machine. This step was not taken, as it was demonstrated that a clean copy of the data was present. Reverting back to the virtual machine's snapshot and the PowerShell ransomware script, it was clear the recovery process was repeatable.

Figure 9. Sample WannaCry Ransomware Successfully Executing

4.3.4. Effectiveness of Control #10

The following table, shown in ***Error! Reference source not found.***, is the recorded results for the tests performed against the CIS Control Data Recovery Capabilities. In cases where the ransomware and ransomware-like scripts were able to run successfully, data was able to be recovered and restored through the process of backups that were taken and stored on an external hard drive.

Group	Ransomware	Source	Download permitted?	Execution permitted without interruption?	Did it run slower?	Was Partial, Full or No Execution allowed?	Did CSC Control allow for recovery if execution was permitted?	Notes
G1	WannaCry	theZoo GitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G1	Cerber	theZoo GitHub	Yes	No	N/A	No Execution	N/A	See notes above
G1	PowerShell Ransomware	Thomas Rayner	Yes	Yes	N/A	Full Execution	N/A	Successful Execution
G2	WannaCry	theZoo GitHub	Yes	No	No	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G2	Cerber	theZoo GitHub	Yes	No	No	No Execution	N/A	See notes above
G2	PowerShell Ransomware	Thomas Rayner	Yes	Yes	No	Full Execution	Yes	Successful Execution, Control allowed for recovery of data with a clean backup
G3	WannaCry	theZoo GitHub	Yes	Yes	No	Full Execution	Yes	Successful Execution, Control allowed for recovery of data with a clean backup
G3	Cerber	theZoo GitHub	Yes	Yes	No	Full Execution	Yes	See notes above
G3	PowerShell Ransomware	Thomas Rayner	Yes	Yes	No	Full Execution	Yes	See notes above

Figure 10. Experimental Results for CSC Control Data Capabilities

4.4. CSC Control #9: Limitation & Control of Network Ports, Protocols, & Services

CSC Control #9 covers managing and documenting the use of required ports, protocols and services on network devices is crucial in minimizing vulnerabilities available to adversaries within the Windows environment (CIS, 2019).

4.4.1. Experiment Group #1

Results for Experiment Group #1 did not differ from the previous test. Upon downloading and extracting Ransomware samples from theZoo on GitHub, Windows Defender removed the malicious artifacts as they landed on the endpoint. The ransomware-like PowerShell script was able to successfully run without being interrupted.

4.4.2. Experiment Group #2

The two sub-controls involved in this test were Sub-Controls #2 and #4, as depicted in ***Error! Reference source not found.***, below. This process involved going through the list of default Windows firewall rules, disabling everything that was not essential, and adding explicit deny rules. Upon successfully configuring the Windows firewall rules, the ransomware from GitHub was downloaded. Windows Defender again removed the malicious artifacts as they came through to the endpoint.

Sub-Control	Asset Type	Security Function	Control Title	Control Descriptions	Implementation Groups 1	2	3
9.1	Devices	Identify	Associate Active Ports, Services, and Protocols to Asset Inventory	Associate active ports, services, and protocols to the hardware assets in the asset inventory.		x	x
9.2	Devices	Protect	Ensure Only Approved Ports, Protocols, and Services Are Running	Ensure that only network ports, protocols, and services listening on a system with validated business needs are running on each system.		x	x
9.3	Devices	Detect	Perform Regular Automated Port Scans	Perform automated port scans on a regular basis against all systems and alert if unauthorized ports are detected on a system.		x	x
9.4	Devices	Protect	Apply Host-Based Firewalls or Port-Filtering	Apply host-based firewalls or port-filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	x	x	x
9.5	Devices	Protect	Implement Application Firewalls	Place application firewalls in front of any critical servers to verify and validate the traffic going to the server. Any unauthorized traffic should be blocked and logged.		x	x

Figure 11. Sub-Controls Implemented for Testing Highlighted in Green (CIS, 2019)

4.4.3. Experiment Group #3

Following the same approach as experiment Group #2, all unnecessary firewall rules and ports were disabled with an explicit deny rule added to catch anything not being implicitly allowed. After going through this lengthy process, Windows Defender was disabled to determine how this control responds to the ransomware samples. The firewall rules did not prevent the download of the ransomware. From here, the execution of WannaCry, Cerber, and PowerShell was performed. Although execution was permitted, the ransomware was not able to 'phone home,' due to the services and ports being locked down. It is worth noting that, even if WannaCry had been able to communicate, it would have failed upon reaching a registered domain, as this would have triggered its 'kill switch.'

4.4.4. Effectiveness of Control #9

Although Control #9 was not effective in stopping the ransomware's execution, it prevented its ability to communicate back to its home base. Windows comes with many ports and services allowed and open. Closing these down indeed narrows the attack surface leaving the system less vulnerable. Historically, ransomware preys on specific ports and services being open for exploitation. With proper implementation, the ransomware would not be able to effectively call back to its home base, which would reduce the total impact.

Group	Ransomware	Source	Download permitted?	Execution permitted without interruption?	Did it run slower?	Was Partial, Full or No Execution allowed?	Did CSC Control allow for recovery if execution was permitted?	Notes
G1	WannaCry	theZooGitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G1	Cerber	theZooGitHub	Yes	No	N/A	No Execution	N/A	See notes above
G1	PowerShell Ransomware	Thomas Rayner	Yes	Yes	N/A	Full Execution	N/A	Successful Execution
G2	WannaCry	theZooGitHub	Yes	No	No	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G2	Cerber	theZooGitHub	Yes	No	No	No Execution	N/A	See notes above
G2	PowerShell Ransomware	Thomas Rayner	Yes	Yes	No	Full Execution	No	Successful Execution
G3	WannaCry	theZooGitHub	Yes	Yes	No	Full Execution	No	Successful Execution
G3	Cerber	theZooGitHub	Yes	Yes	No	Full Execution	No	Successful Execution
G3	PowerShell Ransomware	Thomas Rayner	Yes	Yes	No	Full Execution	No	Successful Execution

Figure 12. Experimental Results for Limitation & Control of Network Ports, Protocols, & Services

4.5. CSC Control #8 Malware Defenses

CSC Control #8 addresses controlling the installation, execution, and spread of malicious code on endpoints within the enterprise is critical. Automation can be leveraged to update defenses quickly, gather data, and take corrective action (CIS, 2019).

4.5.1. Experiment Group #1

There was no variation in results from the first tests with experiment Group #1.

4.5.2. Experiment Group #2

Windows Defender comes enabled as part of the base install for Windows 10. A few configuration changes needed to occur to line up with the sub-controls highlighted in green in ***Error! Reference source not found.***. Sub-Controls #1 and #6 were left out of

testing, as Windows Defender is managed locally, and those would not make a difference for this experiment's purpose. Sub-Controls #4 and #5, USB and Removable Drives, though very important in an enterprise setting, were not part of the proposal and were not tested. Checking for the latest protection updates through Windows Security in the control panel ensured the virtual machine was compliant with Sub-Control #2. Moving to Sub-Control #3, Data Execution Prevention (DEP) and Address Space Layout Randomization (ASLR) both come enabled by default.

In most cases, ransomware attempts to phone home back to a command-and-control server, and this is where Sub-control #7 is crucial— the enabling of DNS Query Logging, as shown in ***Error! Reference source not found.***, below. Lastly, Sub-Control #8, that enables command-line audit logging, is often overlooked. Enabling this control requires three separate modifications in the Windows event viewer: enabling Module Logging, PowerShell Script Block Logging, and PowerShell Transcription. These three configurations are in the event viewer under Windows Logs and Security. While all three of these vary slightly in the level of detail they capture, they are important to use in combination. It is worth noting that Script Block Logging will provide the most detail and should be used at a minimum to help identify the execution of malicious code and commands.

Sub-Control	Asset Type	Security Function	Control Title	Control Descriptions	Implementation Groups 1	2	3
8.1	Devices	Protect	Utilize Centrally Managed Anti-Malware Software	Utilize centrally managed anti-malware software to continuously monitor and defend each of the organization's workstations and servers.		x	x
8.2	Devices	Protect	Ensure Anti-Malware Software and Signatures Are Updated	Ensure that the organization's anti-malware software updates its scanning engine and signature database on a regular basis.	x	x	x
8.3	Devices	Detect	Enable Operating System Anti-Exploitation Features/ Deploy Anti-Exploit Technologies	Enable anti-exploitation features such as Data Execution Prevention (DEP) and Address Space Layout Randomization (ASLR) that are available in an operating system or deploy appropriate toolkits that can be configured to apply protection to a broader set of applications and executables.		x	x
8.4	Devices	Detect	Configure Anti-Malware Scanning of Removable Media	Configure devices so that they automatically conduct an anti-malware scan of removable media when inserted or connected.	x	x	x
8.5	Devices	Protect	Configure Devices to Not Auto-Run Content	Configure devices to not auto-run content from removable media.	x	x	x
8.6	Devices	Detect	Centralize Anti-Malware Logging	Send all malware detection events to enterprise anti-malware administration tools and event log servers for analysis and alerting.		x	x
8.7	Network	Detect	Enable DNS Query Logging	Enable Domain Name System (DNS) query logging to detect hostname lookups for known malicious domains.		x	x
8.8	Devices	Detect	Enable Command-Line Audit Logging	Enable command-line audit logging for command shells, such as Microsoft PowerShell and Bash		x	x

Figure 13. Sub-controls implemented for testing highlighted in green (CIS, 2019)

4.5.3. Experiment Group #3

This group was not leveraged for testing this control, since the purpose of this group was to disable Windows Defender to test the other controls.

4.5.4. Effectiveness of Control #8

Malware Defense proved to be quite useful, stopping and pulling apart the ransomware samples at each download and extraction. Part of its success was due to outdated ransomware samples and their signatures being known. Windows Defender was unable to stop the PowerShell ransomware script from executing. However, Malware Defense sub-controls called for additional logging of the command-line, which provided valuable insight into the script's commands. In an enterprise setting with centralized logging and reporting, this sub-control would allow the defensive side to follow the kill chain and allow for a smoother mitigation and eradication process. As shown in ***Error! Reference source not found.*** and ***Error! Reference source not found.***, the organization leveraging Group #1 would not see what was happening, compared to Group #2, an organization that is leveraging this control and its sub-controls.

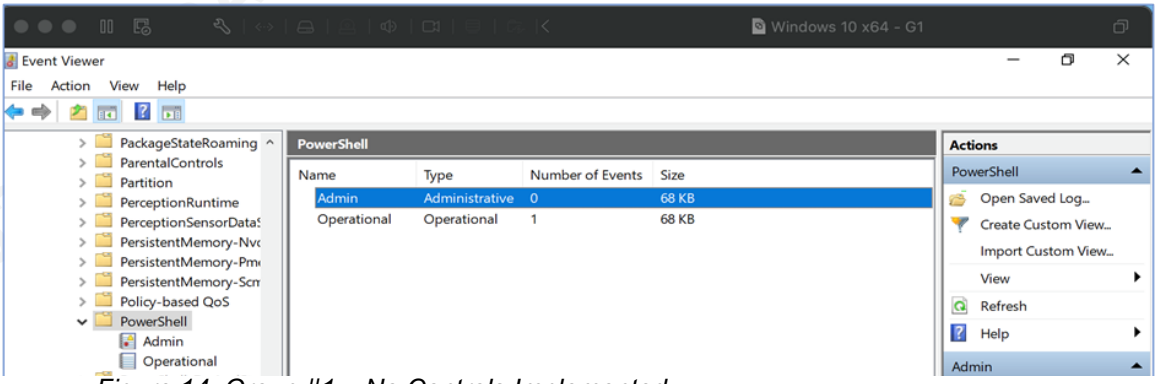

Figure 14. Group #1 – No Controls Implemented

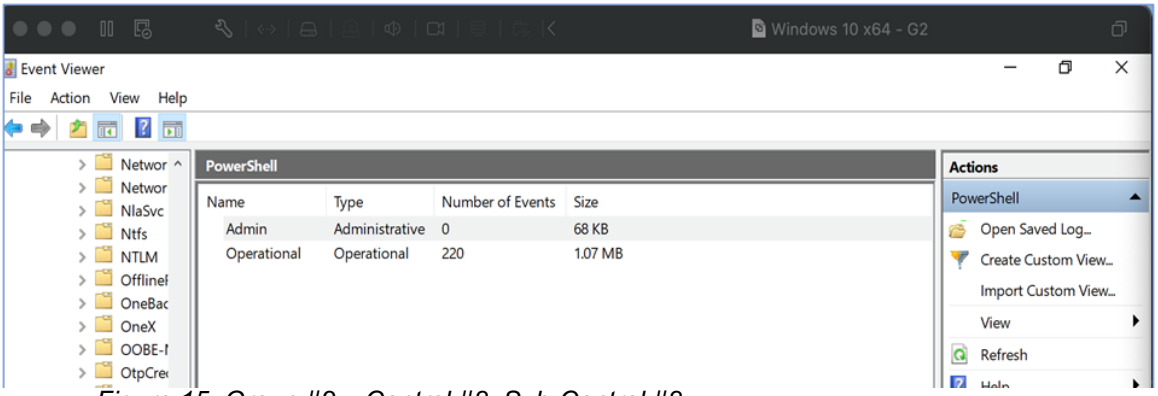

Figure 15. Group #2 – Control #8, Sub-Control #8

4.6. CSC Control #5 Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers

CSC Control #5 states that organizations should establish, implement, and manage the security configuration of laptops, workstations, mobile devices, and servers using rigorous configuration management and change control processes to prevent adversaries from exploiting vulnerable configurations (CIS, 2019).

4.6.1. Experiment Group #1

There was no variation in results from the first tests with experiment Group #1.

4.6.2. Experiment Group #2

In testing this control, the first sub-control implemented was Establish Secure Configurations. PowerShell script to assist with the CIS baseline hardening of this Windows virtual machine was obtained from a scipags page on GitHub called HardeningKitty. Upon following the setup process, it was time to execute the script. The first execution went through and configured most of the settings required to be CIS compliant. The second execution of the script went through and evaluated the settings to validate if anything was missing. Upon completion, the virtual machine was rated at six or excellent, confirming the system was now hardened. After hardening, the ransomware was obtained from theZoo online repository on Github. No issues were encountered downloading the source files. Upon attempting to extract the ransomware from its zip files to execute it, Windows prevented the zip file from being opened. At the same time, Windows Defender went through and started generating alerts for the malicious artifacts. Interestingly, in previous tests, Windows allowed for the opening of the zip file before Windows Defender started firing off alerts. The PowerShell script mimicking ransomware was able to execute without being stopped.

Sub-Control	Asset Type	Security Function	Control Title	Control Descriptions	Implementation Groups 1	2	3
5.1	Applications	Protect	Establish Secure Configurations	Associate active ports, services, and protocols to the hardware assets in the asset inventory.	X	X	X
5.2	Applications	Protect	Maintain Secure Images	Ensure that only network ports, protocols, and services listening on a system with validated business needs are running on each system.		X	X
5.3	Applications	Protect	Securely Store Master Images	Perform automated port scans on a regular basis against all systems and alert if unauthorized ports are detected on a system.		X	X
5.4	Applications	Protect	Deploy System Configuration Management Tools	Apply host-based firewalls or port-filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.		X	X
5.5	Applications	Detect	Implement Automated Configuration Monitoring Systems	Place application firewalls in front of any critical servers to verify and validate the traffic going to the server. Any unauthorized traffic should be blocked and logged.		X	X

Figure 16. Group #2 - Control #5, Sub-Control #1 (CIS, 2019)

4.6.3. Experiment Group #3

This group was not leveraged for testing this control since this group's purpose was to disable Windows Defender to test the other controls; however, Windows Defender is a part of this control.

4.6.4. Effectiveness of Control #5

Overall, this control was very effective in stopping the ransomware from executing on the system. Although the download itself was permitted, the zip file's opening and execution of the malicious process were not allowed. These baseline configurations provide a significant amount of hardening towards an endpoint that ultimately closes off many openings that come with an out-of-box installation of Windows. It was not successful in stopping the PowerShell script from executing. This is expected, as the script itself is not malicious enough to be detected.

Group	Ransomware	Source	Download permitted ?	Execution permitted without interruption?	Did it run slower?	Was Partial, Full or No Execution allowed?	Did CSC Control allow for recovery if execution was permitted?	Notes
G1	WannaCry	theZoo GitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G1	Cerber	theZoo GitHub	Yes	No	N/A	No Execution	N/A	See notes above
G1	PowerShell Ransomware	Thomas Rayner	Yes	Yes	N/A	Full Execution	N/A	Successful Execution
G2	WannaCry	theZoo GitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents, extraction of zip was not permitted.
G2	Cerber	theZoo GitHub	Yes	No	N/A	No Execution	N/A	See notes above
G2	PowerShell Ransomware	Thomas Rayner	Yes	Yes	No	Full Execution	Yes	Successful Execution
G3	WannaCry	theZoo GitHub	N/A	N/A	N/A	N/A	N/A	Not utilized for this control as the purpose of this group was to disable Windows Defender
G3	Cerber	theZoo GitHub	N/A	N/A	N/A	N/A	N/A	See notes above
G3	PowerShell Ransomware	Thomas Rayner	N/A	N/A	N/A	N/A	N/A	See notes above

Figure 17. Experimental Results for Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers

4.7. CSC Control #4 Controlled Use of Administrative Privileges

CSC Control #4 includes the processes and tools used to manage the use, assignment, and configuration of administrative privileges on computers, applications, and networks (CIS, 2019).

4.7.1. Experiment Group #1

There was no variation in results from the first tests with experiment Group #1.

4.7.2. Experiment Group #2

CIS's Control #4 deals with controlling the use of Administrative privileges. For this test, Sub-Controls #3 and #7 were implemented for testing. Implementing sub-control three involved the creation of a local account without administrative privileges. For Sub-Control #7, PowerShell and command prompted were limited to administrative users only through the local group policy. After finalizing the configuration changes, theZoo ransomware was downloaded from the standard user account. Upon downloading and extracting the zip, Windows Defender kicked in and started identifying and removing

Sub-Control	Asset Type	Security Function	Control Title	Control Descriptions	Implementation Groups 1	2	3
4.1	Users	Detect	Maintain Inventory of Administrative Accounts	Use automated tools to inventory all administrative accounts, including domain and local accounts, to ensure that only authorized individuals have elevated privileges.		X	X
4.2	Users	Protect	Change Default Passwords	Before deploying any new asset, change all default passwords to have values consistent with administrative level accounts.	X	X	X
4.3	Users	Protect	Ensure the Use of Dedicated Administrative Accounts	Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not Internet browsing, email, or similar activities.	X	X	X
4.4	Users	Protect	Use Unique Passwords	Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		X	X
4.5	Users	Protect	Use Multi-Factor Authentication for All Administrative Access	Use multi-factor authentication and encrypted channels for all administrative account access.		X	X
4.6	Users	Protect	Use Dedicated Workstations For All Administrative Tasks	Ensure administrators use a dedicated machine for all administrative tasks or tasks requiring administrative access. This machine will be segmented from the organization's primary network and not be allowed Internet access. This machine will not be used for reading email, composing documents, or browsing the Internet.			X
4.7	Users	Protect	Limit Access to Scripting Tools	Limit access to scripting tools (such as Microsoft® PowerShell and Python) to only administrative or development users with the need to access those capabilities.		X	X
4.8	Users	Detect	Log and Alert on Changes to Administrative Group Membership	Configure systems to issue a log entry and alert when an account is added to or removed from any group assigned administrative privileges.		X	X
4.9	Users	Detect	Log and Alert on Unsuccessful Administrative Account Login	Configure systems to issue a log entry and alert on unsuccessful logins to an administrative account.		X	X

Figure 18. Group #2 - Control #4, Sub-Control #3 & #7

malicious artifacts. This time around, the PowerShell script's execution was not successful as it was blocked for standard users.

4.7.3. Experiment Group #3

Applying the same configuration steps mentioned in experiment Group #2 above, Windows Defender was disabled this time. Upon disabling Windows Defender, the next step was to switch back over to the standard user account and download ransomware from theZoo. Extraction and execution of the ransomware was permitted from the perspective of a standard user allowing the ransomware to compromise the endpoint. PowerShell was again stopped due to the configuration change put in place by Sub-Control #7.

4.7.4. Effectiveness of Control #4

Control #4 was effective in stopping the PowerShell script from executing. Downloading the ransomware and executing it as a standard user did not seem to stop or prevent successful compromise. Though this did not prove to be effective for this specific test and use case, these controls are still very important. Administrative tasks should always be performed from an elevated account and not from the perspective of an account with standard access. Limiting access to administrative accounts will reduce the potential impact when credentials or endpoints are compromised.

4.8. CSC Control #10, 9, 8, 5, & 4

In this test, the five controls covered above will be combined to see if the results vary.

4.8.1. Experiment Group #1

There was no variation in results from the first tests against experiment Group #1.

Group	Ransomware	Source	Download permitted ?	Execution permitted without interruption?	Did it run slower?	Was Partial, Full or No Execution allowed?	Did CSC Control allow for recovery if execution was permitted?	Notes
G1	WannaCry	theZoo GitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G1	Cerber	theZoo GitHub	Yes	No	N/A	No Execution	N/A	See notes above
G1	PowerShell Ransomware	Thomas Rayner	Yes	Yes	N/A	Full Execution	N/A	Successful Execution
G2	WannaCry	theZoo GitHub	Yes	No	N/A	No Execution	N/A	Windows Defender removed contents on download and extraction of zip
G2	Cerber	theZoo GitHub	Yes	No	N/A	No Execution	N/A	See notes above
G2	PowerShell Ransomware	Thomas Rayner	Yes	No	No	No Execution	Yes	No Execution
G3	WannaCry	theZoo GitHub	Yes	Yes	No	Full Execution	N/A	Download and Execution permitted
G3	Cerber	theZoo GitHub	Yes	Yes	No	Full Execution	N/A	See notes above
G3	PowerShell Ransomware	Thomas Rayner	Yes	No	N/A	No Execution	N/A	No Execution

Figure 19. Experimental Results for Controlled Use of Administrative Privileges

4.8.2. Experiment Group #2

All five CIS controls and sub-controls covered above were implemented together on one virtual machine for this experiment group. To re-cap, this involved configuring Veaam as a backup solution, Windows Defender for malware defense, Windows Firewall for limiting ports and services, HardenKitty for CIS Baselining, and implementing a standard user account for controlling administrative privileges. Upon successful implementation of these controls, the ransomware samples were downloaded. The samples were permitted for download as previously seen, Windows Defender started scanning the malicious artifacts and removing them. Extraction of the ransomware samples from the zip file was not permitted, as previously seen. The PowerShell script was not permitted as PowerShell access was limited to administrative users only.

4.8.3. Experiment Group #3

This group was not leveraged for testing, as Windows Defender was included for this test.

4.8.4. Effectiveness of Controls #10, 9, 8, 5, & 4 Implemented Together

Implementing all five of the controls together proved to be successful in stopping the virtual machine from becoming infected even though the downloads were permitted. The PowerShell script was unsuccessful in execution. If the execution was permitted in either scenario, recovery would have been successful through backups. No individual one

of these controls is a silver bullet; the more of them that are implemented together, the less likely it is that a system will be compromised.

5. Recommendations & Implications

5.1. Recommendations for Practice

The use of VMware to build out a lab to test CSC Controls' effectiveness against ransomware samples proved to be a valuable resource throughout this experiment. It allowed for easy replication between each control and sub-control that was tested. The applications used as part of this experiment worked well; however, it would be most useful to test the tools applicable to the specific entity looking to implement these specific controls or others not covered in this research.

Minimal configuration changes of these virtual machines were implemented to simulate a base Windows install. A base install was used to avoid the complexities of locking down the endpoint, potentially lessening the effectiveness of the controls if it was locked down more or less. Windows Defender comes enabled by default, which interfered with testing, resulting in the creation of an additional testing group. Future researchers should include this configuration change from the start to avoid this extra step.

While theZoo had some excellent ransomware samples available, they were outdated. It certainly was a worthy repository to pull from, compared to others found during the research process. Leveraging PowerShell ransomware-like scripts proved to be valuable, as Windows Defender did not consistently stop them.

Two controls were identified as the most effective between the five implemented and tested as part of this research. The first control was Control #4, Controlled Use of Administrative Privileges, and the reason is that it was the only control that prevented the execution of the PowerShell script. PowerShell and command prompt can be very dangerous utilities in a Windows environment and, in most cases, should be limited to administrative users and activities. Though Malware Defenses proved extremely valuable in preventing the ransomware from executing, there is one control that took a few additional steps, Secure Configurations for Hardware and Software on Mobile Devices,

Laptops, Workstations, and Servers. This control ties back to CIS benchmarks, which involve hardening a system. In this hardening, antivirus is included, along with closing off ports, services, and many other settings within the Windows environment, which ultimately creates some overlap. While Data Recovery Capabilities was not effective in preventing the ransomware from executing, it did provide valuable insight into the importance of backing up data.

5.2. Implications for Future Research

The CIS Controls proved themselves useful and effective best practices for the systems and conditions in this study. One limitation, however, was the lack of zero-day (or newly-released) ransomware available, and the effectiveness of these controls against a novel attack remains unknown. Nonetheless, the CIS Controls proved useful in the tests performed. Zero-day ransomware samples that have been altered are needed to determine how specific controls, such as Malware Defense, would hold up. Data Recovery Capabilities proved that backing up data is invaluable, and no matter how compromised an endpoint is, with a clean backup, it is possible to revert to that data.

Additional research into available ransomware samples is needed, as the samples leveraged were dated. Finding ransomware samples on the web proved to be difficult while searching credible websites. It is possible this research missed a repository with samples that may have been more effective for testing. Further research into PowerShell scripting may have proven to be more useful for an individual who is more skilled in those scripting capabilities. It would be highly beneficial for future researchers to explore this, as it is a safer option.

VMware Fusion pro was useful for building virtual machines, cloning them, and capturing screenshots. Creating full clones of a base image and creating snapshots proved valuable to test each sample and revert. While VMware allowed for the isolation of the VM, it would be beneficial for future researchers to take a more cautious approach, and have a more dedicated testing environment, as opposed to their primary workstation.

5.3. Potential Gaps

Among the controls tested above, one gap identified was that the malicious files were permitted in all cases. Windows Defender caught some artifacts during the download, but a good portion of them went undetected until extraction of the zip file was attempted. However, it is possible this gap is a result of the controls and sub-controls covered in this research and not a gap across the set of CIS controls. Combining all of the controls for a final test showed the value of having multiple controls in place working together. This research's main potential gap is that not all of the CIS controls were tested and implemented together. Implementing some controls, and not others, will result in coverage gaps. Malware Defense provided good protection against the ransomware but not the PowerShell script that is where controlled use of administrative privileges came in to provide additional protection. Without backups in place, recovering data would not have been possible in the event malware defense failed.

6. Conclusion

The proposed test environment presented in this paper is a viable solution for testing CIS critical security controls. Though the ransomware samples were dated, that did not stop the implementation and testing of the controls. Weighing the controls' effectiveness was a bit tricky, as zero-day ransomware is often not as forgiving as these samples were, but that did not take away from the core principles behind the controls and sub-controls tested. Whether a ransomware is dated, or zero-day, a clean backup of data or a system will allow for its safe recovery. Implementing Malware Defense can be valuable for preventing malicious software from executing. Or, if execution is allowed, Malware Defense can provide valuable insight into exactly what occurred against that system allowing for remediation and further tuning. Although closing out ports and adding deny firewall rules did not prevent the ransomware from executing, it is still valuable to lock down a system as much as possible, narrowing the potential attack surface. Without the CIS Controls in place, there would have been significantly less visibility into what occurred, and the recovery process would not have been feasible. Future research using this approach could involve additional CIS controls not covered, testing them individually and simultaneously, and further evaluating their effectiveness.

References

Bisson, D. (2016, February 18). *Hollywood Hospital Pays $17,000 to Ransomware Attackers.* Trip Wire: The State of Security. https://www.tripwire.com/state-of-security/latest-security-news/hollywood-hospital-pays-17000-to-ransomware-attackers/

Bisson, D. (2019, September 9). *10 of the Most Significant Ransomware Attacks of All Time.* Trip Wire: The State of Security. https://www.tripwire.com/state-of-security/security-awareness/10-most-significant-ransomware-attacks/

Boddy, M., Jones, B., & Stockley, M. (2019, July). *RDP Exposed – The Threat That's Already at Your Door.* Sophos. https://www.sophos.com/en-us/medialibrary/PDFs/technical-papers/sophos-rdp-exposed-the-threats-thats-already-at-your-door-wp.pdf

Challita, A. (2018, August 9). *The four most popular methods hackers use to spread ransomware.* ITProPortal. https://www.itproportal.com/features/the-four-most-popular-methods-hackers-use-to-spread-ransomware/

CIS (2019, April). *The 20 CIS Controls & Resources.* Center for Internet Security. https://www.cisecurity.org/controls/cis-controls-list/

KnowBe4. (n.d.). *Ransomware Timeline*. What Is Ransomware? https://www.knowbe4.com/ransomware#ransomwaretimeline

ManageEngine. (n.d.). *Implementing the CIS Controls. https://www.manageengine.com/cis-critical-security-controls/*

Rayner, T. (2015, November 11). *Using PowerShell To Simulate A Ransomware Attack.* Writing code & automating IT. https://thomasrayner.ca/using-powershell-to-simulate-a-ransomware-attack/

Richardson, R. & North, M. (2017, January 1). *Ransomware: Evolution, Mitigation, & Prevention.* Kennesaw State University Faculty Publications. https://digitalcommons.kennesaw.edu/cgi/viewcontent.cgi?article=5312&context=facpubs

Rudis, B., Beardsley, T., Hart, J., & Sellers, T. (2017, June 14). *National Exposure Index 2017.* Rapid 7 Global Assets. https://www.rapid7.com/globalassets/_pdfs/research/rapid7-national-exposure-index-2017.pdf

Savage, K., Coogan, P. & Lau, H. (2015, August 6). *Security Response: The evolution of ransomware.* Symantec Threat Intel. https://its.fsu.edu/sites/g/files/imported/storage/images/information-security-and-privacy-office/the-evolution-of-ransomware.pdf

Sornson, P. (2020, July 16). *Center for Information Security (CIS) Offers Cybersecurity Solutions.* Sonrai Security Blog. https://sonraisecurity.com/blog/center-for-information-security-cis-offers-cybersecurity-solutions/

University of California, Berkeley. (n.d.). *How does a computer become infected with Ransomware?* Berkeley Information Security Office. https://security.berkeley.edu/faq/ransomware/how-does-computer-become-infected-ransomware

Verizon. (2020). *Data Breach Investigations Report.* https://enterprise.verizon.com/resources/reports/2020-data-breach-investigations-report.pdf