**NANYANG TECHNOLOGICAL UNIVERSITY**

**4th YEAR COMPUTER ENGINEERING**

**Final Year Project**

**Windows Management Instrumentation Security**

****

Interim Report

Name: Chew Zhi Jie

Date: 30/1/2017

Table of Contents

[1.1 Introduction 3](#_Toc505768499)

[1.2 Purpose Of Interm Report 4](#_Toc505768500)

[2 Works Done So Far 4](#_Toc505768501)

[3 Future Work 5](#_Toc505768502)

[4 Conclusion 5](#_Toc505768503)

# 1.1 Introduction

Cyber attacks are on the rise, costing companies millions to tens of millions in average. Stolen private information and denial of service also affect individuals extensively. An example is the massive global ransomware cyber attack known as “WannaCrypt” or “WannaCry”. It affected more than 150 countries whereby tens of thousands of computers were infected, including critical infrastructures like hospital which forced them to stop servicing partially.

Malware, also known as file-based attack, has been a norm since the early computer era. In majority of the case, malware attacks can be track down easily by Security Analyst. However, several recent high-profile attacks have adopted “living off the land” technique where it is a file-less attack and leverages heavily on legitimate system management tool such as the Windows Management Instrumentation. The first file-less attack was first discovered in 2010 and it has been on a rise since then. These attacks normally go under the radar of security protection software and is nearly impossible to track as it is usually stored in very unusual places such as the Memory. These legitimate system management tools bring convenience to system administrator, however, the flip side is that it also increases the risk of the system being compromised.

One of those system management tools is Windows Management Instrument, also known as WMI. WMI is preinstalled since the early Windows 2000 Operating System. It is a Web-Based Enterprise Management System for accessing computer information in an enterprise environment. Administrators can utilize Windows Management Instrumentation Command-line to interact with WMI. WMI consists of various privileged functions such as retrieve sensitive data, script automated task, manipulate system registry and many more. This is also where WMI becomes dangerous, attackers cloud use creative ways to abuse WMI. An example is using WMI to inject payload, normally an executable script, into the system registry and bind the payload to a startup event. Hence, the malicious script will be automatically executed upon startup and this technique hijacked the victim to actively listen to Control and Command Server. NotPetya, a cousin of WannaCry, is an example of a cyber attack that utilizes WMI. NotPetya uses PsExec and WMIC to spread its infection. Although NotPetya is not as deadly compared to WannaCry, it also infected over 12,500 machines and further the possibilities of WMI-based attack.

# 1.2 Purpose Of Interm Report

This paper presents the work done so far and future work on the study of offensive component of WMI security and the defenses that are currently available on the market to prevent WMI-based attack. For offensive component, WMI will be used to remotely abuse and attack victim computer without any use of payload. It will then be used to compile into a post exploitation tool for Certis Cisco.

# 2 Works Done So Far

**1. Project initialisation**

* Create a private GitHub repository to store all progress, logs and notes, <https://github.com/Nicholas-Chew/WMI-Security>
* Research on articles and white papers related to WMI Security and log down on

WMI-Security/Readme.md(GitHub) for future study

* Create EndNote to hold white papers for ease of future reference and citation

**2. Understand the basics of WMI**

* Study the basics of WMI such as what is WMI, WMI architecture and its usage
* Study the relation between WMI and CIM
* Study on how to interact with WMI using either WMIC (DCOM protocol) or PSExec (WsMAN protocol)
* Create notes for research studies along the way for ease of future reference under

WMI-Security/Notes/(GitHub)

**3. Learn the coding and scripting skills needed for WMI manipulation**

* Skills needed to manipulate WMI is mainly VBScript and PowerShell Scripting
  + Learn both skills through Lynda
* Generate complex VBScript and PowerShell Script to manipulate system

**4. Research and study how can WMI be misuse**

* Study on WMI temporary and permanent Event Subscription
* Create and delete WMI permanent event on local machine
* Create complex WMI-based query using WQL, WMI Query Language

**5. Create a PowerShell module for WMI Event Subscription**

* WMI Event Subscription creation is complex and time consuming, hence create a WMI Event Subscription Power Shell module

**6. Create two file-less attack using WMI Event Subscription and VBScript for post-exploitation**

* First attack hijacks all the internet browsers such as Firefox or Chrome in victim’s system
  + This attack rewrites the hijack even after the browser resets
* Second attack hijack victim’s system and zombify the machine
  + This attack listens to a C2 server for command; it automatically traverse and duplicate itself in the connected network
* Test both attacks on a default corporate network configuration (Google Cloud Platform)

**5. Test WMI attack against antimalware solution and system logging**

* Test against Bitdefender Total Security 2018 and Windows Defender
  + Result: Both solution did not detect anything after a full scan
* No trace of WMI permanent subscription being created in system log

**7. Compile above knowledge into a WMI-based post exploitation tool**

* Create a command line base post exploitation tool that closely mimic the style of Metasploit using python
* Create the basic structure of the post exploitation tool
  + The tool focuses on file-less attack that is nearly impossible to trace and avoids cyber security protection infrastructure and solution

# 3 Future Work

There will be two major part for the future work:

1. Create more modules for the post exploitation tool
   1. The current tool created does not have any module, hence it is incomplete
2. Test WMI-based attack on more antimalware solution

# 4 Conclusion

There was a delay at the start due to the complexity and skills needed to manipulate WMI. After more research and practice on manipulating WMI, the project gradually catches up with the planned timeline.

However, the post exploitation tool lacks proper professional feedback and testing which might not be suitable for Certis Cisco usage.

Lastly, the project focuses on creating a tool that is suitable and more importantly useful to its industrial attachment company, Certis Cisco.