Incident Detection and Response Report

Client: Microsoft Windows

Project: Incident Detection and Response on Windows 8.1

Report Date: November 5, 2024

Prepared By: Kennedy Emeanuri

Responder: Kennedy Emeanuri

Contact information:

https://www.linkedin.com/in/kennedyemeanuri/

TABLE OF CONTENTS

INTRODUCTION

Timeline Of Events

On **November 1, 2024**, the following events occurred

9:00 AM: An application named “Putty” was downloaded on to the Windows based system.

9:35 AM: The application was last modified.

EXECUTIVE SUMMARY

This report provides a detailed account of the detection, investigation, and remediation of a suspected security incident involving unauthorized remote access on a Windows-based system. Using a suite of cybersecurity tools, including TCPView, Process Explorer, and Autoruns from Microsoft Sysinternals, the investigation identified an active, unauthorized connection facilitated by a potentially malicious application, *putty.exe*. This application had established a remote connection, potentially compromising the system and posing a threat to the larger network.

The investigation commenced with the detection of an unusual process, *putty.exe*, which was connected to a remote IP address via TCPView. Further analysis with Process Explorer traced this process back to *explorer.exe*, indicating that the unauthorized application was likely downloaded through Internet Explorer and was configured to persist on the system via an autorun entry under the Windows registry. Autoruns confirmed the presence of a suspicious autorun task labelled “secure” in the registry, set to autostart upon system login.

Upon confirming the malicious nature of the application, containment efforts were initiated, including immediate network disconnection to prevent lateral movement and additional potential compromise of connected systems. Remediation involved terminating the active connection, eradicating all traces of the malicious application, and removing registry entries to prevent re-execution on startup. Finally, proactive security measures, including firewall activation, system updates, and antivirus software installation, were implemented to safeguard the system from future threats.

The report concludes with security recommendations focused on preventing similar incidents in the future. These include enhancing network monitoring and alert, restricting administrative privileges, enabling windows defender and firewall policies, and enhancing email and download filtering. These preventive measures aim to enhance security monitoring, reduce the attack surface, and strengthen the overall resilience of the system.

1. TECHNICAL SUMMARY

1.1 Tools Used for Investigation

a. TCPView: Used to list all active TCP and UDP network connections, including remote addresses, port numbers, connection states, and process IDs associated with each connection.

b. Process Explorer: Provided detailed information on running processes, memory usage, threads, and open file handles.

c. Autoruns: Displayed all programs configured to run automatically upon Windows startup, including scheduled tasks, services, drivers, and codecs.

1.2 Evidence Collected

a. Using TCPView revealed that the connection to the remote address was in an "Established" state and identified the specific port number used. It also indicated that the connection was initiated by a downloaded application named *putty.exe*.

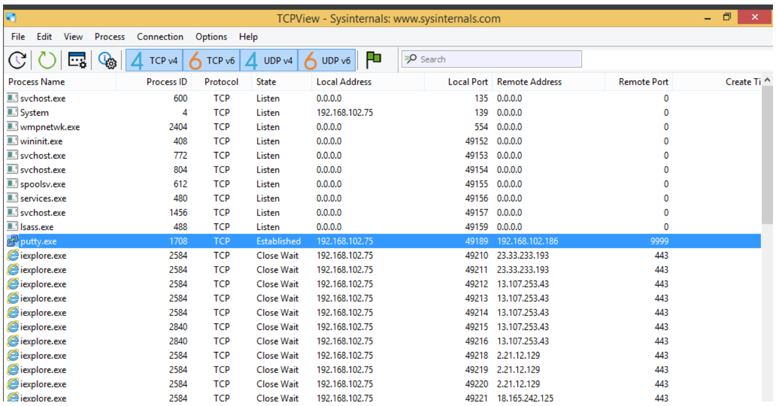
b. Using Process Explorer displayed the process tree, with *explorer.exe* as the parent process and *putty.exe* as a child process.

c. Using Autoruns showed the task "secure" entered in the Windows registry, enabling automatic connection to the remote address upon user login.

2. TECHNICAL DETAILS

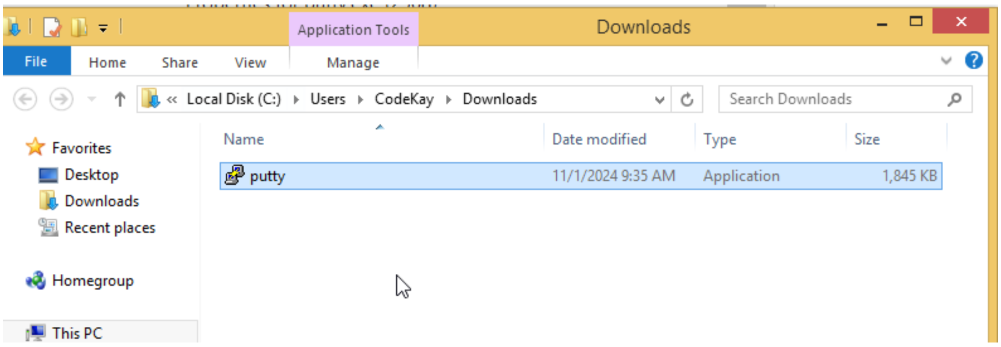
2.1 Incident Detection

While monitoring and auditing a system with the IP address "192.168.102.186" using TCPView, I observed a suspicious process named *putty.exe* with a process ID of 1708 and a local port of 49189. This process had established a connection to the remote address 192.168.102.186 on port 9999.


TCPView - Sysinternals: www.sysinternals.com
File Edit View Process Connection Options Help
4 TCP v4
6 TCP v6
4 UDP v4
6 UDP v6
Search

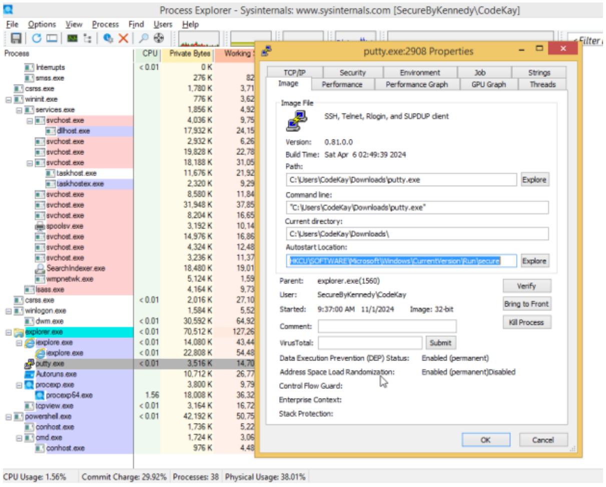
Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Ti
svchost.exe	600	TCP	Listen	0.0.0.0	135	0.0.0.0	0	
System	4	TCP	Listen	192.168.102.75	139	0.0.0.0	0	
wmpnetwk.exe	2404	TCP	Listen	0.0.0.0	554	0.0.0.0	0	
wininit.exe	408	TCP	Listen	0.0.0.0	49152	0.0.0.0	0	
svchost.exe	772	TCP	Listen	0.0.0.0	49153	0.0.0.0	0	
svchost.exe	804	TCP	Listen	0.0.0.0	49154	0.0.0.0	0	
spoolsv.exe	612	TCP	Listen	0.0.0.0	49155	0.0.0.0	0	
services.exe	480	TCP	Listen	0.0.0.0	49156	0.0.0.0	0	
svchost.exe	1456	TCP	Listen	0.0.0.0	49157	0.0.0.0	0	
lsass.exe	488	TCP	Listen	0.0.0.0	49159	0.0.0.0	0	
putty.exe	1708	TCP	Established	192.168.102.75	49189	192.168.102.186	9999	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49210	23.33.233.193	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49211	23.33.233.193	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49212	13.107.253.43	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49213	13.107.253.43	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49214	13.107.253.43	443	
iexplore.exe	2840	TCP	Close Wait	192.168.102.75	49215	13.107.253.43	443	
iexplore.exe	2840	TCP	Close Wait	192.168.102.75	49216	13.107.253.43	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49218	2.21.12.129	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49219	2.21.12.129	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49220	2.21.12.129	443	
iexplore.exe	2584	TCP	Close Wait	192.168.102.75	49221	18.165.242.125	443	

Upon identifying this process, I investigated further and determined it originated from an application named *putty*, which was downloaded and last modified on 11/1/2024 at 9:35 AM.



Name	Date modified	Type	Size
putty	11/1/2024 9:35 AM	Application	1,845 KB

To understand the function of this process, I conducted additional analysis using Microsoft Sysinternals' Process Explorer. This revealed that the parent process was *explorer.exe*, indicating that the *putty* application was likely downloaded via Internet Explorer. Furthermore, the application appeared to be running a task named "secure," which was set to autostart on login under the registry key: **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run**.



Process Explorer - Sysinternals: www.sysinternals.com [SecureByKennedy\CodeKay]

Process	CPU	Private Bytes	Working S
Interrupts	< 0.01	0 K	
smss.exe		276 K	82
csrss.exe		1,780 K	3,71
wininit.exe		776 K	3,62
services.exe		1,856 K	4,92
svchost.exe		4,036 K	9,75
dllhost.exe		17,932 K	24,15
svchost.exe		2,932 K	6,26
svchost.exe		19,828 K	22,78
svchost.exe		18,188 K	31,05
taskhost.exe		11,676 K	21,92
taskhostex.exe		2,320 K	9,29
svchost.exe		8,580 K	11,84
svchost.exe		31,948 K	37,85
svchost.exe		8,204 K	16,65
spoolsv.exe		3,192 K	10,14
svchost.exe		14,976 K	16,86
svchost.exe		4,324 K	12,48
svchost.exe		3,236 K	11,37
SearchIndexer.exe		18,480 K	19,01
wmpnetwk.exe		5,124 K	1,59
lsass.exe		4,164 K	9,73
csrss.exe	< 0.01	2,016 K	27,10
winlogon.exe		1,584 K	5,52
dwm.exe	< 0.01	30,592 K	64,92
explorer.exe	< 0.01	70,512 K	127,26
iexplore.exe	< 0.01	14,080 K	43,44
iexplore.exe	< 0.01	22,808 K	54,48
putty.exe	< 0.01	3,516 K	14,70
Autoruns.exe		10,712 K	26,77
procexp.exe		3,800 K	9,79
procexp64.exe	1.56	18,008 K	36,32
tcpview.exe	< 0.01	3,164 K	16,72
powershell.exe	< 0.01	42,192 K	50,75
conhost.exe		1,736 K	5,22
cmd.exe		1,724 K	3,06
conhost.exe		976 K	4,48

CPU Usage: 1.56% | Commit Charge: 29.92% | Processes: 38 | Physical Usage: 38.01%

putty.exe:2908 Properties

TCP/IP Security Environment Job Strings
Image Performance Performance Graph GPU Graph Threads

Image File

SSH, Telnet, Rlogin, and SUPDUP client

Version: 0.81.0.0

Build Time: Sat Apr 6 02:49:39 2024

Path:

C:\Users\CodeKay\Downloads\putty.exe

Command line:

"C:\Users\CodeKay\Downloads\putty.exe"

Current directory:

C:\Users\CodeKay\Downloads\

Autostart Location:

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\secure

Parent: explorer.exe(1560)

User: SecureByKennedy\CodeKay

Started: 9:37:00 AM 11/1/2024 Image: 32-bit

Comment:

VirusTotal:

Data Execution Prevention (DEP) Status: Enabled (permanent)

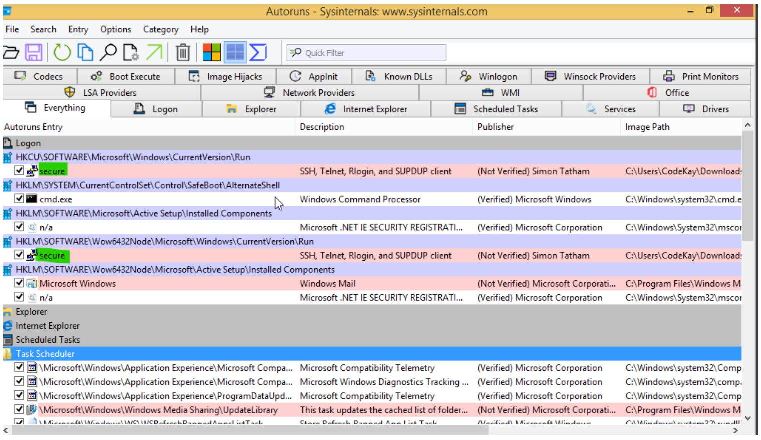
Address Space Load Randomization: Enabled (permanent)Disabled

Control Flow Guard:

Enterprise Context:

Stack Protection:

After discovering this autorun task, I used another Sysinternals tool, Autoruns, to further investigate. This analysis confirmed that the "secure" task had instances under both **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run and HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run.**



Autoruns - Sysinternals: www.sysinternals.com

File Search Entry Options Category Help
Quick Filter
Codecs Boot Execute Image Hijacks AppInit Known DLLs Winlogon Winsock Providers Print Monitors
LSA Providers Network Providers WMI Office
Everything Logon Explorer Internet Explorer Scheduled Tasks Services Drivers

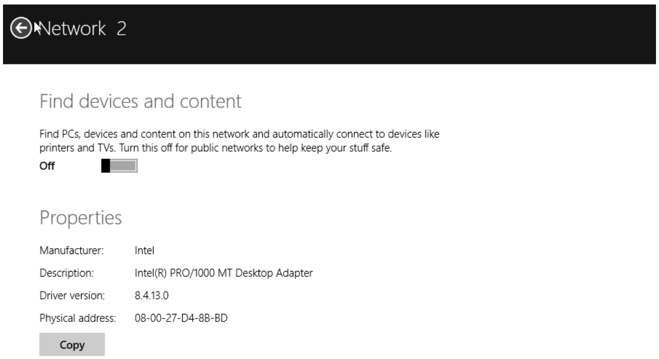
Autoruns Entry	Description	Publisher	Image Path
Logon			
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run			
☑ secure	SSH, Telnet, Rlogin, and SUPDUP client	(Not Verified) Simon Tatham	C:\Users\CodeKay\Download:
HKLM\SYSTEM\CurrentControlSet\Control\SafeBoot\AlternateShell			
☑ cmd.exe	Windows Command Processor	(Verified) Microsoft Windows	C:\Windows\system32\cmd.e
HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components			
☑ n/a	Microsoft .NET IE SECURITY REGISTRATI...	(Verified) Microsoft Corporation	C:\Windows\System32\mscor
HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run			
☑ secure	SSH, Telnet, Rlogin, and SUPDUP client	(Not Verified) Simon Tatham	C:\Users\CodeKay\Download:
HKLM\SOFTWARE\Wow6432Node\Microsoft\Active Setup\Installed Components			
☑ Microsoft Windows	Windows Mail	(Not Verified) Microsoft Corporati...	C:\Program Files\Windows M
☑ n/a	Microsoft .NET IE SECURITY REGISTRATI...	(Verified) Microsoft Corporation	C:\Windows\System32\mscor
Explorer			
Internet Explorer			
Scheduled Tasks			
Task Scheduler			
☑ \Microsoft\Windows\Application Experience\Microsoft Compa...	Microsoft Compatibility Telemetry	(Verified) Microsoft Corporation	C:\Windows\system32\Comp
☑ \Microsoft\Windows\Application Experience\Microsoft Compa...	Microsoft Windows Diagnostics Tracking ...	(Verified) Microsoft Corporation	C:\Windows\system32\comp
☑ \Microsoft\Windows\Application Experience\ProgramDataUpd...	Microsoft Compatibility Telemetry	(Verified) Microsoft Corporation	C:\Windows\system32\Comp
☑ \Microsoft\Windows\Windows Media Sharing\UpdateLibrary	This task updates the cached list of folder...	(Not Verified) Microsoft Corporati...	C:\Program Files\Windows M

Based on these findings, I concluded that *putty.exe* was a malicious application designed to enable unauthorized remote access to the system.

2.2 Incident Response

2.2.1 Containment

Given the system's connection to a wireless network, I promptly disconnected it to contain the incident and prevent potential lateral movement to other connected systems.



Network 2

Find devices and content

Find PCs, devices and content on this network and automatically connect to devices like printers and TVs. Turn this off for public networks to help keep your stuff safe.

Off

Properties

Manufacturer: Intel

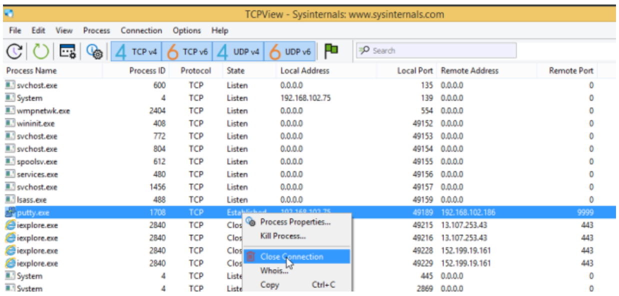
Description: Intel(R) PRO/1000 MT Desktop Adapter

Driver version: 8.4.13.0

Physical address: 08-00-27-D4-8B-BD

2.2.2 Eradication

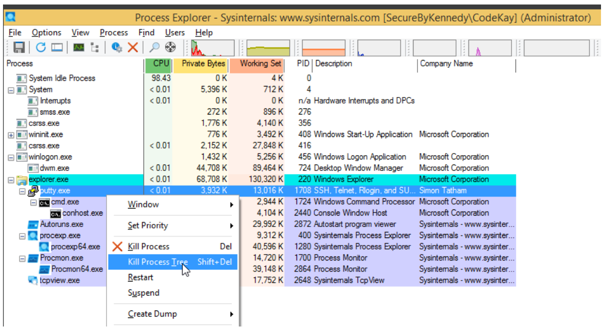
With the system isolated, I proceeded to remove all traces of the application and its associated processes and autorun tasks. First, using TCPView, I terminated the connection to the remote address, effectively halting communication with it.



TCPView - Sysinternals: www.sysinternals.com

Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port
svchost.exe	600	TCP	Listen	0.0.0.0	135	0.0.0.0	0
System	4	TCP	Listen	192.168.102.75	139	0.0.0.0	0
wmpnetwk.exe	2404	TCP	Listen	0.0.0.0	554	0.0.0.0	0
wininit.exe	408	TCP	Listen	0.0.0.0	49152	0.0.0.0	0
svchost.exe	772	TCP	Listen	0.0.0.0	49153	0.0.0.0	0
svchost.exe	804	TCP	Listen	0.0.0.0	49154	0.0.0.0	0
spoolsv.exe	612	TCP	Listen	0.0.0.0	49155	0.0.0.0	0
services.exe	480	TCP	Listen	0.0.0.0	49156	0.0.0.0	0
svchost.exe	1456	TCP	Listen	0.0.0.0	49157	0.0.0.0	0
lsass.exe	488	TCP	Listen	0.0.0.0	49159	0.0.0.0	0
putty.exe	1708	TCP	Established	192.168.102.75	49189	192.168.102.186	9999
iexplore.exe	2840	TCP	Clos		49215	13.107.253.43	443
iexplore.exe	2840	TCP	Clos		49216	13.107.253.43	443
iexplore.exe	2840	TCP	Clos		49228	152.199.19.161	443
iexplore.exe	2840	TCP	Clos		49229	152.199.19.161	443
System	4	TCP	Liste		445	0.0.0.0	0
System	4	TCP	Liste		2869	0.0.0.0	0

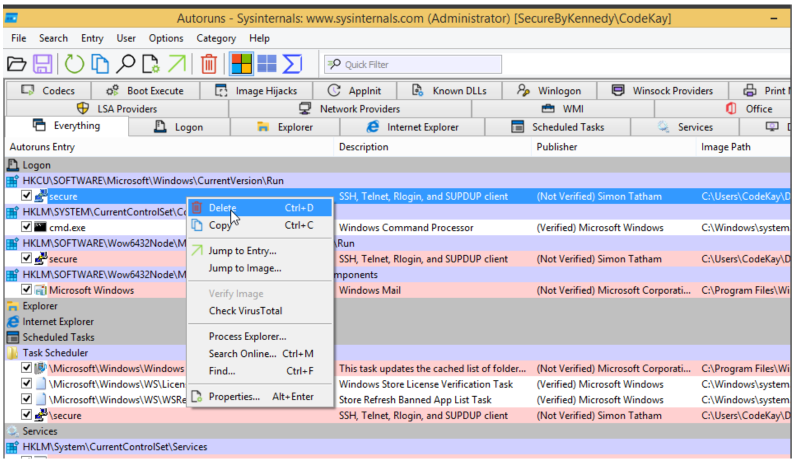
Next, I used Process Explorer to kill the entire process tree initiated by *explorer.exe*, terminating any additional malicious processes potentially created by the *putty* application.



Process Explorer - Sysinternals: www.sysinternals.com [SecureByKennedy\CodeKay] (Administrator)

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
System Idle Process	98.43	0 K	4 K	0		
System	< 0.01	5,396 K	712 K	4		
Interrupts	< 0.01	0 K	0 K	n/a	Hardware Interrupts and DPCs	
smss.exe		272 K	896 K	276		
csrss.exe		1,776 K	4,140 K	356		
wininit.exe		776 K	3,492 K	408	Windows Start-Up Application	Microsoft Corporation
csrss.exe	< 0.01	2,152 K	27,848 K	416		
winlogon.exe		1,432 K	5,256 K	456	Windows Logon Application	Microsoft Corporation
dwm.exe	< 0.01	44,708 K	89,464 K	724	Desktop Window Manager	Microsoft Corporation
explorer.exe	< 0.01	68,708 K	130,320 K	220	Windows Explorer	Microsoft Corporation
putty.exe	< 0.01	3,932 K	13,016 K	1708	SSH, Telnet, Rlogin, and SU...	Simon Tatham
cmd.exe			2,944 K	1724	Windows Command Processor	Microsoft Corporation
conhost.exe			4,104 K	2440	Console Window Host	Microsoft Corporation
Autoruns.exe			29,992 K	2872	Autostart program viewer	Sysinternals - www.sysinter...
procexp.exe			9,312 K	400	Sysinternals Process Explorer	Sysinternals - www.sysinter...
procexp64.exe			40,596 K	1280	Sysinternals Process Explorer	Sysinternals - www.sysinter...
Procmon.exe			14,720 K	1700	Process Monitor	Sysinternals - www.sysinter...
Procmon64.exe			39,148 K	2864	Process Monitor	Sysinternals - www.sysinter...
tcpview.exe			17,752 K	2648	Sysinternals TcpView	Sysinternals - www.sysinter...

Afterwards, I deleted the "secure" autorun task from the Windows registry to prevent it from automatically executing each time a user logged in.



Autoruns - Sysinternals: www.sysinternals.com (Administrator) [SecureByKennedy\CodeKay]

Autoruns Entry	Description	Publisher	Image Path
Logon			
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run			
☑ secure	SSH, Telnet, Rlogin, and SUPDUP client	(Not Verified) Simon Tatham	C:\Users\CodeKay\D
HKLM\SYSTEM\CurrentControlSet\C			
☑ cmd.exe	Windows Command Processor	(Verified) Microsoft Windows	C:\Windows\system
HKLM\SOFTWARE\Wow6432Node\M	\Run		
☑ secure	SSH, Telnet, Rlogin, and SUPDUP client	(Not Verified) Simon Tatham	C:\Users\CodeKay\D
HKLM\SOFTWARE\Wow6432Node\M	mponents		
☑ Microsoft Windows	Windows Mail	(Not Verified) Microsoft Corporati...	C:\Program Files\Wi
Explorer			
Internet Explorer			
Scheduled Tasks			
Task Scheduler			
☑ \Microsoft\Windows\Windows	This task updates the cached list of folder...	(Not Verified) Microsoft Corporati...	C:\Program Files\Wi
☑ \Microsoft\Windows\WS\Licen	Windows Store License Verification Task	(Verified) Microsoft Windows	C:\Windows\system
☑ \Microsoft\Windows\WS\WSRe	Store Refresh Banned App List Task	(Verified) Microsoft Windows	C:\Windows\system
☑ \secure	SSH, Telnet, Rlogin, and SUPDUP client	(Not Verified) Simon Tatham	C:\Users\CodeKay\D
Services			
HKLM\System\CurrentControlSet\Services			

2.2.3 Recovery

Upon successfully eradicating the malicious processes and tasks, I reconnected the system to the network, implementing additional security measures, including enabling the Windows firewall, installing third-party antivirus software, and updating the system.

2.3 Recommendations

Based on the findings of the incident detection and response, the following recommendations can be made to improve the security of the Windows system:

a. **Enhance Network Monitoring and Alerts**
Set up network monitoring tools to detect unusual outbound connections or connections to known malicious IP addresses. Configure alerts for any unexpected port activity or unusual communication patterns, particularly from unfamiliar applications.

b. **Restrict Administrative Privileges**
Limit the number of accounts with administrative privileges. By reducing access to administrative accounts, you can prevent unauthorized software installations and modifications to registry settings, such as those found with the *secure* autorun task.

c. **Enable Windows Defender and Firewall Policies**
Ensure that Windows Defender and firewall policies are active and appropriately configured. Custom rules should be added to block unauthorized access and detect unusual connections to unknown ports or IP addresses.

d. **Enhance Email and Download Filtering**
Block or monitor the downloading of executable files like *putty.exe* through email and web browsing. Deploy email and web filtering solutions that scan for potentially dangerous attachments or links, reducing the chance of malicious downloads.

3. CONCLUSION

This report highlights the critical steps involved in detecting, analysing, and responding to a security incident involving unauthorized access on a Windows-based system. The use of advanced monitoring tools, including TCPView, Process Explorer, and Autoruns, was instrumental in identifying a malicious process (*putty.exe*) that had established unauthorized remote access. By systematically analysing the suspicious process and associated registry entries, it was possible to determine that the application was configured to persist through autorun tasks, posing an ongoing security risk to the networked environment.

Following the identification of the incident, immediate containment measures were employed to isolate the system, preventing potential lateral movement within the network. Eradication efforts focused on removing all traces of the malicious application and its associated autorun tasks, while a series of preventive security measures were enacted to enhance the system's resilience against future attacks. The incident not only underscored the value of proactive monitoring but also highlighted the importance of comprehensive incident response protocols that facilitate swift containment and remediation.

To mitigate the risk of similar incidents in the future, this report outlines several best practices, including the enforcement of application whitelisting, regular audits of autorun entries, and stringent user access controls. These recommendations are designed to reinforce security defences and improve incident readiness across the network.

In conclusion, this report demonstrates the necessity of an organized, well-equipped approach to incident detection and response. By leveraging effective tools and adhering to best practices, organizations can significantly reduce the impact of security incidents and safeguard sensitive systems and data. Continuous improvement of incident response capabilities, coupled with proactive security measures, will be essential in maintaining a robust defence against evolving cybersecurity threats.

4. APPENDICES

4.1 Glossary of Terms

a. **Autoruns**
A tool from the Microsoft Sysinternals Suite that displays programs and services configured to run automatically upon Windows startup, including registry entries and scheduled tasks.

b. **HKCU (HKEY_CURRENT_USER)**
A registry hive in Windows containing configuration information specific to the currently logged-in user. This hive includes information about user-specific settings, such as applications set to run on startup.

c. **HKLM (HKEY_LOCAL_MACHINE)**
A registry hive in Windows that stores configuration information specific to the system and hardware, affecting all users on the device. It includes settings for applications that may run for all users upon startup.

d. **Lateral Movement**
The technique used by attackers to move within a network after gaining initial access, often to reach critical assets or escalate privileges.

e. **Local Port**
A port on the host machine through which applications communicate with network services. In this context, a local port is used by a process to establish connections with remote devices.

f. **Process Explorer**
A tool from the Microsoft Sysinternals Suite that provides detailed information about active processes, including process IDs, memory usage, parent-child process relationships, and open file handles.

g. **Process ID (PID)**
A unique identifier assigned by the operating system to each running process, allowing it to be tracked and managed independently.

h. **Process Tree**
A hierarchical structure showing the parent and child relationships between processes on a system, where child processes are initiated by parent processes.

i. **Registry Key**
A specific location within the Windows registry that stores configuration settings for the operating system and applications. Registry keys can be used to set programs to run automatically on startup.

j. **Remote Address**
The IP address of a device outside the local system with which the system is

communicating over a network. Remote addresses are often used to connect to servers or other external devices.

k. **Sysinternals Suite**
A collection of utilities by Microsoft designed to troubleshoot and manage Windows operating systems, providing tools for system monitoring, process management, and startup configuration.

l. **TCPView**
A tool from the Microsoft Sysinternals Suite that displays all active TCP and UDP connections, along with details such as remote addresses, local ports, and the associated processes.

m. **Windows Registry**
A centralized database in Windows that stores low-level settings for the operating system and installed applications, used to manage startup items, user settings, and system configurations.