Introduction to Computer Security

Ransomware Project

Sujan Lanka - 11702061

Sai Kumar Reddymalla -11690966

Monica Sai Meghana ghanta - 11798073

Lakshmi Gayatri Donepudi – 11801234

Abstract

The threat from ransomware persists in cybersecurity because it encrypts files to demand money from victims for the decryption keys. The research team established a test environment which brought together Kali Linux as the attacking platform while using Windows-based computers as victims. The Attacker used Metasploit exploit leads to reverse shell creation followed by AES encryption and matching decryption actions. The defense system employs three mechanisms for real-time monitoring through Python Watchdog together with rule-based anomaly detection which automatically terminates malicious processes. The purpose of the research is to explore ransomware behaviors by constructing defense mechanisms through script-based Python implementation. The project showcases an entire ransomware kill chain from infection to remediation through which multiple defensive approaches efficiently counter ransomware operations.

Introduction

The cyber-threat known as ransomware functions as a harmful program which encrypts personal data before requiring monetary ransom for data recovery. The global costs resulting from WannaCry attacks plus many other ransomware incidents reached billions during the past few years. The recent cases show that organizations need to develop efficient detection systems combined with strong mitigation methods immediately. This project performs a ransomware attack simulation which provides an examination of all stages starting from infection

through payload delivery to encryption and detection and finally ending at mitigation.

Through Metasploit we targeted Windows 10 exploit for shell acquisition after which we sent our custom AES-encryption script to the victim machine. The real-time file monitoring occurred under the Watchdog system. A rule-based system monitored system activity for irregular file activity patterns. “ .exe” processes were scanned by a script that took control of them to terminate them. Python automation applications provide an efficient solution to improve the effectiveness of ransomware protection systems.

Related Work

Research on ransomware has increased sharply since the WannaCry incident took advantage of EternalBlue vulnerabilities to infiltrate systems. Multiple reports of ransomware attacks on Windows 7 systems were documented on Microsoft forums because of missing updates. Symmetric AES encryption is available through cryptographic libraries Fernet and PyCryptodome which offer reliable encryption services for security research purposes.

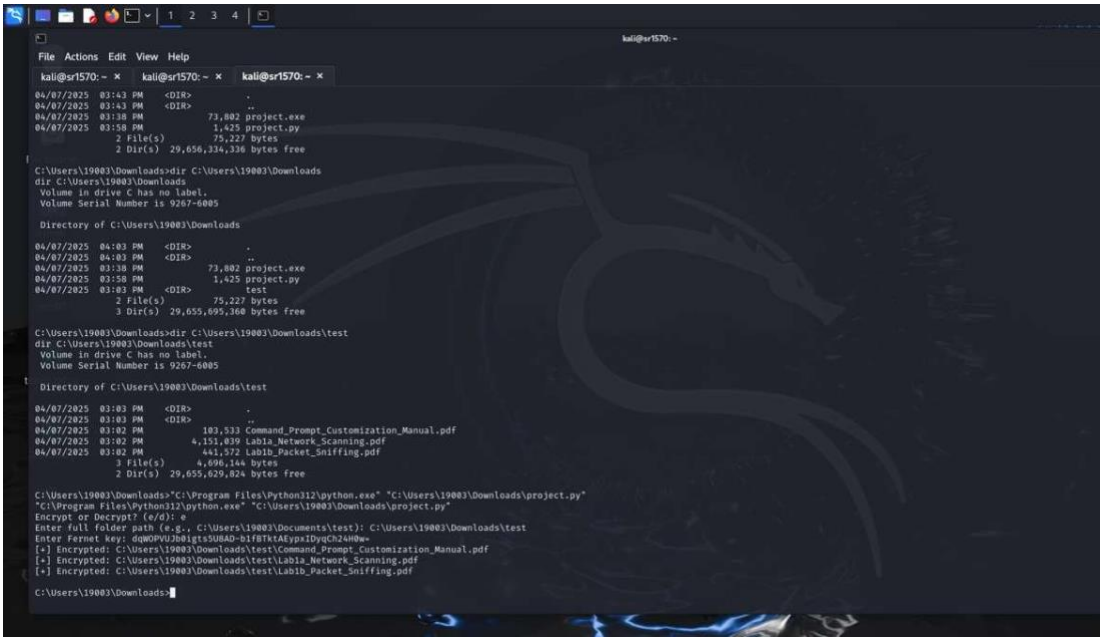
Python Watchdog library lets users monitor file changes in real-time so they can track events and logs through event-driven logging. The literature contains previous research describing rule-based detection systems which track fast file system changes and suspicious access patterns to identify ransomware manifestations. The research findings direct our development of both a practical ransomware test environment along with automatic defensive capabilities.

Approach

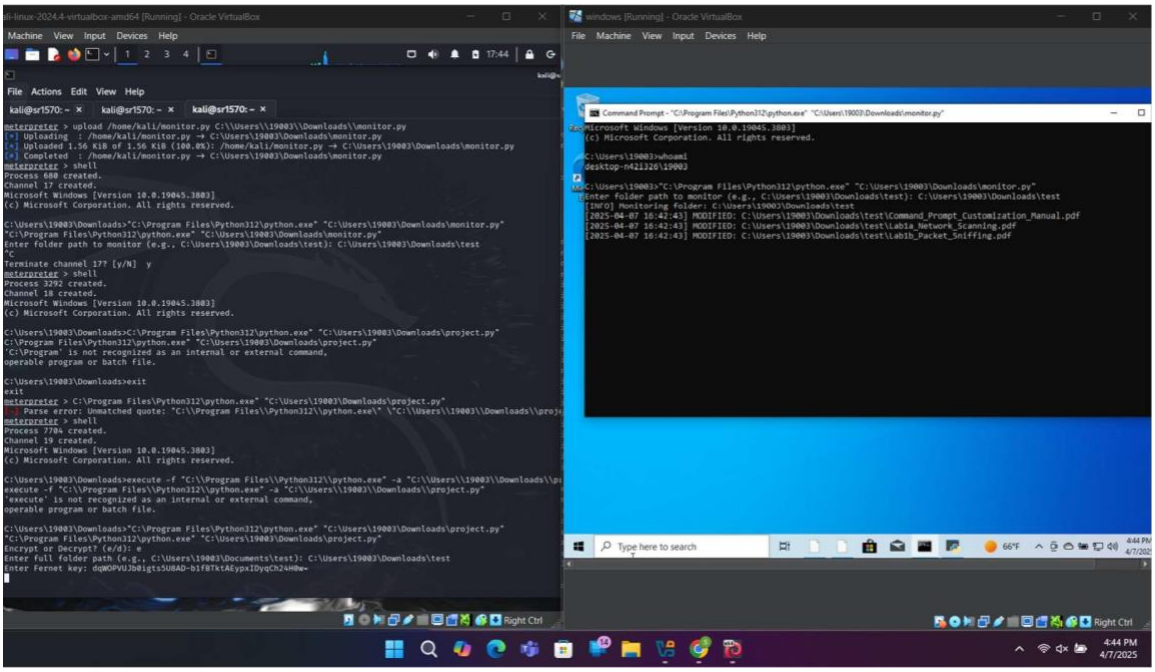
The research follows six essential stages as part of the methodology.

The network devices were discovered through the use of Kali Linux in conjunction with Nmap. The Metasploit to create shell access for the Windows 10 target machine. The operational launch of the reverse shell required manual execution of a .exe file on a Windows 10 machine.

The security application used PyCryptodome for implementing AES-CBC encryption during both encryption and decryption functions. The encryption script needs both a file/folder path and a 32-byte secret key to operate. Each file gains encryption through a 16-byte IV which inserts before the encrypted data. The decryption function executes symmetrically with the exact key used for encryption.

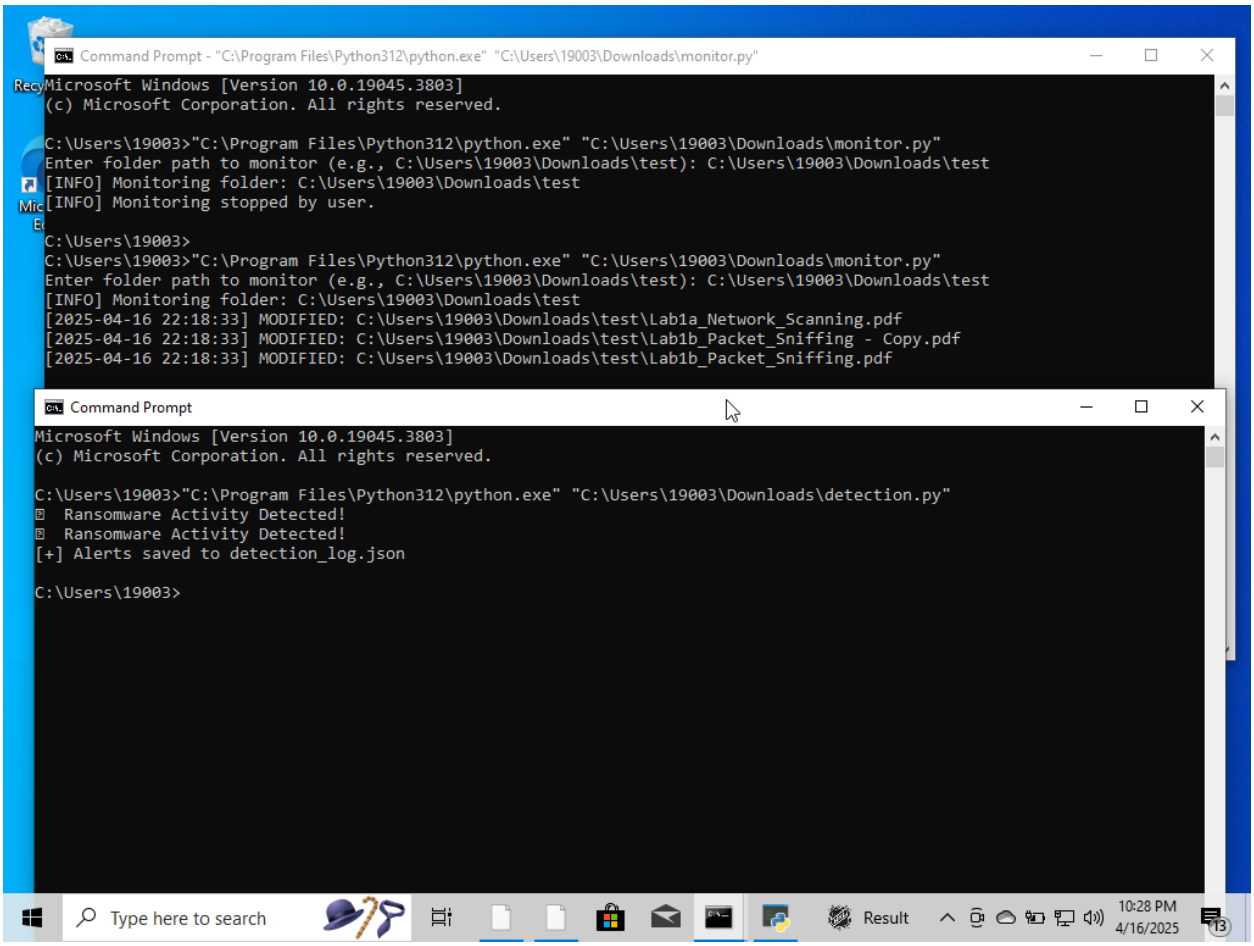


The victim system ran the P,ython Watchdog tool-based monitoring script. The system watches active directories to record every file system operation including timestamp and filename details. The detection logic evaluation received its ground truth from this development.

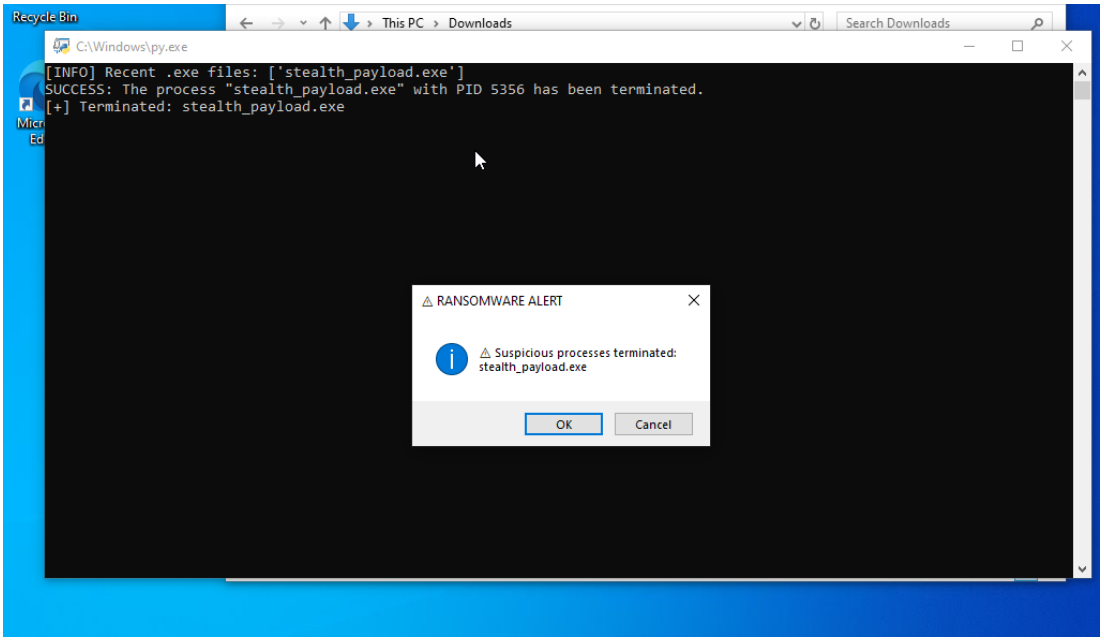


The detection system est1ablishes ransomware behaviors when modifications to three or more PDF files occur within three seconds. Watchdog logs undergo

parsing to recover file activity patterns before suitable alerts get activated.



Through the mitigation script the framework stopped the malicious process to stop the ransomware attack.



Proactive system monitoring together with automatic ransomware detection and response mechanisms proved their defense capabilities according to the results obtained.

Conclusion

The project develops an extensive simulation of ransomware operations while building an automation framework based on Python which detects and fights these threats. Open-source tools and scripting knowled]ge allow users to develop their own effective anti-ransomware protection systems when these tools are paired with publicly available commercial solutions. Research into the future development of the system should include machine learning capabilities for anomaly detection and rule engine improvements for adaptive threat handling methods.

References

[1] SEED Labs, *Ransomware Lab*, [Online]. Available: https://seedsecuritylabs.org/Labs_20.04/PDF/Ransomware_Lab.pdf

[2] Rapid7, *Metasploit Framework Documentation*, [Online]. Available: https://docs.rapid7.com/metasploit/

[3] Python Cryptography Authority, *Fernet Encryption Documentation*, [Online]. Available: https://cryptography.io/en/latest/fernet/

[4] Python Software Foundation, *Watchdog – File Monitoring Library*, [Online]. Available: https://python-watchdog.readthedocs.io/en/latest/

[5] Python Software Foundation, *Python Standard Library Documentation*, [Online]. Available: https://docs.python.org/3/library/index.html