Offensive Security Incident Responder Exam Report

OSIR Exam Report

student@youremailaddress.com, OSID: OS-XXXXX

2024-12-02

Contents

1 Offensive Security Incident Responder Exam Report

1.1 Introduction

The OffSec Incident Responder exam report contains all efforts that were conducted in order to pass the OffSec certification examination. This report should contain all items that were used to pass the exam and it will be graded from a standpoint of correctness and fullness to all aspects of the exam. The purpose of this report is to ensure that the student has a full understanding of incident response methodologies as well as the technical knowledge to pass the qualifications for the OffSec Incident Responder.

1.2 Objective

The objective of this assessment is to respond to an incident in the Megacorp One environment. The objective in the first phase is to identify all compromised systems and detect if sensitive data was exfiltrated or encrypted. Phase 2 involves performing a forensic analysis on a disk image provided by a colleague from another branch of the Megacorp One's Incident Response team. Based on their initial analysis, the disk image contains a post-exploitation framework binary that contains an encryption key. You must find the binary and obtain the encryption key.

Example pages have already been created for you at the latter portions of this document that should demonstrate the amount of information and detail that is expected in the exam report. Use the sample report as a guideline to get you through the reporting.

1.3 Requirements

The student will be required to fill out this incident response report fully and to include the following sections:

- Executive Summary (All sections)
- Incident Detection and Identification
  - In this section, provide a detailed, story-style walkthrough of Phase 1. Focus on how you identified the answer to each exercise question, and ensure you include the exact Splunk query used in your investigation.
- Incident Detection and Identification - Containment, Eradication, and Recovery
  - In this section, outline the key steps that can be taken to contain and recover compromised systems, as well as eliminate the threat identified in Phase 1. Focus on actions that mitigate the immediate risk, restore system integrity, and remove any remaining traces of the compromise.
- Incident Detection and Identification - Findings
  - In this section, which contains a timetable of your findings, make sure to document the **overall attacker activity or phase** related to each exercise question.
- Forensic Analysis - Disk Image Analysis
  - In this section, provide a detailed, story-style walkthrough of the disk image analysis in Phase 2), focusing on how you identified the malicious binary.
- Forensic Analysis - Malware Analysis
  - In this section, provide a detailed, story-style walkthrough of the malware analysis process in Phase 2), focusing on how you analyzed and identified the encryption key used by the binary.
- Conclusion

The walkthroughs in the **Incident Detection and Identification**, **Disk Image Analysis**, and **Malware Analysis** sections should be clear and thorough, containing enough explanations and screenshots to allow a technically proficient reader to replicate each step. Additionally, ensure that your workflow and decision-making process throughout the analysis are well explained and easily understood.

2 Executive Summary

2.1 Incident Detection and Identification Overview

The SOC team escalated several triggered alerts to the Incident Response team for investigation. The primary objectives were to identify if the triggered alerts contained compromised systems and assess the impact of the attacker's actions, such as determining whether data has been exfiltrated or encrypted.

While investigating the alerts and the recorded data of the incident, we identified three compromised systems in the Megacorp One environment:

- PC1
- PC2
- SRV1

The threat actor accessed and exfiltrated the secret recipe for our chocolate muffins, which could have catastrophic consequences if leaked or sold to competitors.

2.2 High-Level Attack Path

Our investigation revealed the following high-level path the threat actor took to compromise the Megacorp One environment and accessed the sensitive recipe:

1. PC1 was used as the initial entry vector by the threat actor by trying numerous passwords against several user accounts. The threat actor finally succeeded and got access to this machine with administrative privileges.
2. PC2 was configured to use the same password for the local administrator account and the threat actor used it to get access to it. On the machine, the attacker obtained credentials from logged on users by using Mimikatz.
3. SRV1 was accessed using one of the obtained sets of credentials from PC2. The threat actor accessed and exfiltrated the secret chocolate muffin recipe from this machine.

2.3 Forensic Analysis Overview

A disk image was created from a compromised machine in another branch of the Megacorp One enterprise. Analysis of this disk image confirmed that it had been compromised by a threat actor, who had downloaded a password-protected archive containing a malicious binary.

Upon analyzing the binary, we found that it checks whether the system is in a specific state before executing actions to generate a token. By leveraging this token, we were able to obtain an authentication token for the threat actor's Command & Control (C&C) infrastructure, which provided valuable insights into their operations and helped strengthen our security.

3 Incident Detection and Identification

The SOC team escalated several triggered alerts to the Incident Response team as shown in the following screenshot.

Figure 3.1: ImgPlaceholder

One of the escalated alerts is named “Malicious Apps” and monitors the collected events for occurrences of SHA-256 hashes of known malicious applications that are commonly used by threat actors such as Mimikatz and NetExec. The alert triggered only one time for an event recorded at 01/11/2024 1:11:11 AM.

Since events containing information about the usage of applications that are commonly used by attackers may have severe implications, let’s review this event in more detail.

Figure 3.2: ImgPlaceholder

The event provides us several important information that can be leveraged in our incident detection and identification process:

- Username: Administrator
- Filename: Zwetsch.exe
- Directory: `C:\hackingtools\`

Based on the matching SHA-256 hash and the characteristic commandline argument “sekurlsa::logonpasswords”, we can be certain that this is Mimikatz.

[...]

3.1 Containment, Eradication, and Recovery

Once the compromise of PC1, PC2, and SRV1 was confirmed, immediate containment measures were implemented. The affected machines were isolated from the network to prevent further lateral movement and to cut off communication with the attacker’s command and control (C&C) infrastructure.

The eradication phase began with a thorough audit of all local Administrator accounts across the environment to ensure that passwords were unique and secure. The shared local Administrator password between PC1 and PC2 was replaced with new, secure passwords on both machines. Other systems using the same original password were identified and remediated as well. User accounts that had

cached credentials on PC2, likely extracted by the attacker using Mimikatz, were reset. Clean backups were restored to PC1, PC2, and SRV1 to remove any residual threats, and their local Administrator account passwords were updated. To further strengthen security, all local Administrator and user account passwords were reset, and a strong password policy, alongside account lockout mechanisms, was enforced to prevent future password attacks.

3.2 Findings

Timestamp	Observation	Affected Assets
01/09/2024 3:25:00 PM	Beginning of Password Spraying with Password Password1!	Host: PC1
01/09/2024 3:58:00 PM	End of Password Spraying.	Host: PC1
01/09/2024 3:58:15 PM	Successful login for local Administrator user	Host: PC1 User: Administrator (local)
01/09/2024 3:59:00 PM	Download of meterpreter.exe from `<IP>` via Browser	Host: PC1 User: Administrator
01/09/2024 3:59:49 PM	Process Creation of meterpreter.exe	Host: PC1 User: Administrator (local)
01/09/2024 4:05:11 PM	Process Creation of PsExec	Host: PC1 User: Administrator (local) Target Machine: PC2 Target User: Administrator (local) Password: Password1!”
[…]	[…]	[…]
01/11/2024 1:11:11 AM	Process Creation of Zwetsch.exe	Host: PC2 User: Administrator (local)
[…]	[…]	[…]

4 Forensic Analysis

4.1 Disk Image Analysis

We began the analysis of the provided disk image by loading it in Autopsy and enabling the plugin “Recent Activities”. Once the analysis of the disk image is finished, we’ll have several options to start our investigation.

Figure 4.1: ImgPlaceholder

Based on the information that were shared by the other incident response team, a download has been recorded. Therefore, let’s begin by analyzing the “Web Downloads” under “Data Artifacts”. One entry catches our attention which states that offer.7z was downloaded from 192.168.48.130:8000.

Figure 4.2: ImgPlaceholder

Let's check if this archive exists in the Downloads directory of the Admin user where it was downloaded to. If yes,let's try to extract it.

Figure 4.3: ImgPlaceholder

Unfortunately, we get prompted for a password. Since we don't have a password, our extraction attempt fails.

Figure 4.4: ImgPlaceholder

At this point, let's think about how we could obtain such a password. One possibility is to assume that the attacker had only access via CLI and therefore might have used PowerShell to extract the archive. In addition, based on the information from the system we know that PowerShell Script Block Logging is enabled. Let's check out the PowerShell Operational Log in Event Viewer and search for "offer.7z".

This reveals the following event:

Figure 4.5: ImgPlaceholder

The event contains the information that the password superpass was used to extract the archive. Let's try this password and extract the files by using 7z. Once the archive is extracted, a new binary appears named "viruz.exe". The lab asks for the file hash of this binary which we can get via the Cmdlet Get-FileHash: 2D51EF5F421E844EC1278CDAAA1830105D1F879A163AF55EA826B428A0A97E68.

Figure 4.6: ImgPlaceholder

4.2 Malware Analysis

The second lab of the Forensic Analysis phase asks to analyze the binary from the previous lab and obtain a token that is used by the threat actor for authenticating to their C&C infrastructure. Let's start by starting an administrative PowerShell session.

Figure 4.7: ImgPlaceholder

Then, let's navigate to the correct directory and execute the binary.

Figure 4.8: ImgPlaceholder

The binary returns the information that a token is missing. At this point, we can either use static or dynamic analysis. We'll use dynamic analysis using ProcMon and start by adding a filter for viruz.exe.

Figure 4.9: ImgPlaceholder

Once we rerun the binary, we'll see the following entries in ProcMon. One entry shows a "NAME NOT FOUND" Result for the file `C:\Windows\token`.

Figure 4.10: ImgPlaceholder

Using the Cmdlet Test-Path, let's check if the file exists:

Figure 4.11: ImgPlaceholder

As expected the file doesn't exist. To resolve this issue, let's create this file as empty file with the Cmdlet New-Item.

Figure 4.12: ImgPlaceholder

Once done, let's rerun the binary.

Figure 4.13: ImgPlaceholder

Now, the binary returns the information "Waiting for token…" instead of the previous error. In addition, it doesn't terminate itself but waits presumably for input of a token of some kind. Let's clear the ProcMon screen and rerun the binary to see what the binary is doing.

[...]

5 Conclusion

Our incident detection and identification process successfully uncovered three compromised systems within Megacorp One's infrastructure, alongside the exfiltration of our confidential chocolate muffin recipe.

During forensic analysis, it was revealed that the threat actor downloaded a password-protected archive on one of the compromised systems, extracting a malicious binary. This binary leveraged temporary tokens to secure an authentication token linked to the threat actor's Command & Control (C&C) infrastructure. By obtaining this authentication token, we gained valuable insights into the threat actor's operations, enhancing our ability to defend against future attacks.

Through swift containment, recovery of the compromised systems, and eradication of the malicious artifacts, we successfully mitigated the threat and prevented further compromise of Megacorp One's environment.