# Incident Response Technical Report — Week 3

**Containment, Eradication & Recovery**

**Incident Type:** Ransomware Simulation

**Environment:** Windows 10 VM

---

## 1.0 Incident Overview

This report provides a detailed account of the technical analysis and response actions performed during Week 3 of the incident response process, following a ransomware simulation executed in Week 2. The primary objective of this phase is to document the containment, eradication, and recovery procedures in a forensically sound manner.

The scope of this report covers the following key activities:

- **Root Cause Analysis:** Identifying the origin and mechanism of the simulated attack.
- **Containment:** Isolating the affected system to prevent further impact.
- **Eradication:** Removing all artifacts and components related to the simulated threat.
- **Recovery:** Restoring the system to its operational, pre-incident state.

The incident was initiated through a controlled, non-destructive ransomware simulation script, `safe-ransom.ps1`, to mimic the behavior of a real-world ransomware attack within a secure laboratory environment.

## 2.0 Root Cause Analysis (RCA)

### 2.1 Cause of the Incident

The root cause of the simulated ransomware activity was the manual execution of the PowerShell script `C:\Evidence\safe-ransom.ps1`. The script was designed to simulate ransomware behavior by performing the following actions:

- Creating copies of files with a `.locked` extension.
- Generating a ransom note (`README_RECOVER.txt`).
- Logging its actions to `sim_log.txt`.
- Confining its operations to the user's `Documents` folder to prevent system-wide impact.

## 2.2 Evidence Supporting the Root Cause

The following evidentiary artifacts were collected and analyzed to confirm the root cause:

| Artifact | Description |
| --- | --- |
| **Simulation Log** | `sim_log.txt` provided a clear timestamp and sequence of the file "encryption" operations. |
| **Locked Files** | The instantaneous appearance of file copies with the `.locked` extension correlated with the script's execution. |
| **Ransom Note** | The file `README_RECOVER.txt` was automatically generated by the script as designed. |
| **Script File** | The source script, `safe-ransom.ps1`, was located in the `C:\Evidence` directory. |
| **File Hashes** | The integrity of the script and related files was confirmed via SHA256 hashes recorded in `hashes.txt`. |

## 2.3 Impact Assessment

The impact of the simulation was localized to the `C:\Users\whoami\Documents\` directory. No system-level files or directories were affected. The specific items impacted were:

- `notes.txt` and its `.locked` copy
- `password_list.txt` and its `.locked` copy
- `report.txt` and its `.locked` copy
- `README_RECOVER.txt` and its `.locked` copy

# 3.0 Containment Actions

Containment procedures were implemented to prevent the incident from escalating and to preserve the integrity of the evidence.

## 3.1 System Isolation

Given the offline and localized nature of the simulation, the following isolation steps were sufficient:

- The execution of the `safe-ransom.ps1` script was terminated.

- The virtual machine was logically isolated, with no network activity occurring post-incident.
- All evidentiary files were secured in the `C:\Evidence` directory to prevent modification.

These actions satisfied the isolation requirements for a Digital Forensics and Incident Response (DFIR) laboratory setting.

### 3.2 Evidence Preservation

All relevant artifacts were collected and preserved to maintain a verifiable chain of custody. The collected evidence included:

- Original documents
- `.locked` file copies
- The ransom note
- The simulation log
- The simulation script
- A memory dump of the `explorer.exe` process
- A file containing SHA256 hashes of all evidence
- A documented chain of custody and incident timeline.

No modifications were made to the collected evidence to ensure its forensic integrity.

### 3.3 Memory Dump Acquisition

A memory image of the `explorer.exe` process was captured to preserve volatile evidence. The details of the memory dump are as follows:

- **File:** `explorer.DMP`
- **Path:** `C:\Evidence\memory\explorer.DMP`
- **SHA256:** `7BFB7D4148974E81EF33F7973E9CD72FC6E34F1AFF5D37D0FF1AAA38B51E539E`

## 4.0 Eradication Procedures

The eradication phase focused on the complete removal of all malicious components from the system.

### 4.1 Removal of the Attack Source

The `safe-ransom.ps1` script, identified as the source of the attack, was preserved as evidence and then securely removed from any directory in the system's execution path.

### 4.2 Removal of Malicious Artifacts

After verifying the integrity of the original files, all `.locked` files created by the script were deleted. The removed artifacts include:

- `notes.txt.locked`
- `password_list.txt.locked`
- `report.txt.locked`
- `README_RECOVER.txt.locked`

### 4.3 Persistence Check

A thorough investigation was conducted to ensure no persistence mechanisms were established. The following areas were inspected:

- Startup folders
- Windows Registry (e.g., `Run` keys)
- Scheduled Tasks
- System Services
- Active Processes

No persistence mechanisms were discovered. The analysis of the memory dump further confirmed the absence of any hidden or malicious processes.

## 5.0 Recovery Steps

The recovery phase aimed to restore the system to its clean, pre-incident operational state.

### 5.1 File Restoration

Since the simulation script only created copies, the original files remained intact. The following files were verified and confirmed as restored:

- `notes.txt`
- `password_list.txt`
- `report.txt`
- `README_RECOVER.txt`

Their SHA256 hash (as empty test files) was verified as `E3B0C44298FC1C149AFBF4C8996FB92427AE41E4649B934CA495991B7852B855`.

### 5.2 System Validation

Post-recovery validation confirmed the following:

- No malicious scripts were executing.
- No signs of persistence were present.
- No unexpected network behavior was observed.
- No unauthorized modifications were made to system files.

The system was confirmed to be functioning normally.

### 5.3 Environment Reset

The `Documents` and `Evidence` folders were cleared of all temporary and non-evidential files. The system was deemed stable and fully recovered.

## 6.0 Lessons Learned

This simulation reinforced several key principles of incident response:

- **Forensic Discipline:** Even controlled simulations require strict adherence to forensic procedures to ensure the integrity of the process.
- **Importance of Volatile Data:** Memory evidence is critical for effective containment and analysis, allowing for the capture of transient artifacts.
- **Analysis Over Assumption:** The presence of `.locked` files does not definitively indicate encryption; thorough analysis is essential to determine the true nature of an attack.
- **Reconstruction from Logs:** Timelines and logs are fundamental to accurately reconstructing the sequence of events during an incident.
- **Chain of Custody:** Maintaining a verifiable chain of custody is paramount for ensuring the trustworthiness of collected evidence.

## 7.0 Final Deliverables

The following deliverables were completed as part of the Week 3 incident response activities:

- Root Cause Analysis
- Documentation of Containment Actions
- Documentation of Eradication Steps
- Recovery Plan and Execution Record
- Acquisition of Memory Dump

- This formal technical report.

All designated tasks for Week 3 have been fully completed.