Incident Response Playbook: Suspicious DLL/Process Injection

Team AnubisX

Version 1.0
September 23, 2025

Document Control

Attribute	Value
Version	1.0
Status	Draft
Owner	AnubisX Security Team
Review Cycle	Every Quarter

Contents

1 Introduction

1.1 Purpose

This playbook defines incident response procedures for handling ”Suspicious DLL/Process Injection”. It provides roles, responsibilities, detection indicators, containment steps, and recovery guidance to minimize impact and restore services.

1.2 Scope

This playbook applies to systems, network components, cloud services, and personnel. It is intended for use by incident responders, SOC analysts, IT operations, legal, and leadership.

2 Overview of the Attack

Process injection and DLL side-loading allow attackers to run code in the context of legitimate processes, evade detection, and escalate privileges. Key risks include:

- Stealthy execution and persistence
- In-memory credential theft
- Privilege escalation using trusted process context

3 Incident Response Phases

This playbook follows the NIST Incident Response lifecycle framework.

3.1 Phase 1: Preparation

Goal: To ensure the team is equipped and ready to respond to a process injection incident before it occurs.

- **Roles and Responsibilities:** Define roles: Incident Commander, Lead Analyst, Forensics, IT, Communications.
- **Logging Auditing:** Ensure logging and centralized authentication audits are enabled.
- **Tools Resources:** Deploy specialized detection rules and maintain playbooks for the specific alert type.
- **Training:** Regular backups and least-privilege access models.

3.2 Phase 2: Identification & Analysis

Goal: Confirm the activity and determine scope and severity.

1. **Initial Analysis and IOC Evaluation:** Analyze logs and alerts to identify Indicators of Compromise (IOCs). Common IOCs include:
   - Unexpected DLLs loaded into trusted processes
   - Processes exhibiting code injection patterns (CreateRemoteThread, NtCreateSection)
   - Signed-but-modified DLLs or side-loaded binaries

2. **Severity Level Assessment:** Classify the incident to ensure appropriate allocation of resources. Severity is based on: Operational Impact, Criticality of affected systems/data, Scope of attack, and Detection/Recovery timelines (MTTD/MTTR).

Level	**Description**	**Example**	**MTTD**	**MTTR**
Low	Single process shows unusual DLL load.	Developer tool loading plugin unexpectedly.	<4 hrs	<24 hrs
Medium	Multiple instances of injection on several hosts.	Malicious DLL injected into userland processes across hosts.	4-12 hrs	1-3 days
High	Credential theft and lateral movement resulting from injection.	In-memory tools extract credentials and attackers move laterally.	12-24 hrs	3-7 days
Critical	Widespread injection enabling enterprise compromise.	Injection used to establish C2 and deploy ransomware broadly.	24+ hrs	7-21 days

Table 1: Incident Severity Matrix

3.3 Phase 3: Containment

Goal: To limit attacker actions and preserve evidence.

- Suspend affected processes after memory capture, detect child/winapi usage patterns.
- Quarantine host and block further DLL loads from suspicious paths.

3.4 Phase 4: Eradication

Goal: To remove malicious components and prevent reinfection.

- Remove malicious DLLs, replace with signed binaries from trusted sources, reimage if necessary.
- Enable binary whitelisting and code integrity checks.

3.5 Phase 5: Recovery

Goal: To safely restore systems and business operations.

- Validate system integrity and certificate chains.
- Resume services with monitoring in place.

3.6 Phase 6: Post-Incident Activities (Lessons Learned)

Goal: To strengthen resilience and prevent recurrence.

- Conduct a blameless post-mortem and update playbooks.
- Produce final incident report and recommended mitigations.
- Implement controls to reduce recurrence.

4 MITRE ATT&CK Framework Mapping

Suspicious DLL/Process Injection ATT&CK Mapping

- **Tactic: Defense Evasion**
  - *T1055 – Process Injection*
  - *T1218 – Signed Binary Proxy Execution*
- **Tactic: Credential Access**
  - *T1003 – OS Credential Dumping*