🏥 SIEM Runbook for GenAI Incident Investigation

GenAI System Security Event Examples

☀️ Sandy Dunn

- ➰ Prompt Injection Attempt
- ➰ Sensitive Data Leak via Model Output
- ➰ Abnormal Usage or Model Abuse
- ➰ Toxic/Harmful Content Generation
- ➰ API Key Misuse
- ➰ Model Misconfiguration/Drift Detected

Step-by-Step Guide

1. Initial Triage

Action	Details
Review Alert Details	□ Alert name (eg, Prompt Injection Attempt) □ Time of event - Model version and parametersF □ User/session ID, IP, device info
Retrieve Contextual Logs	□ Prompt submitted □ Model output □ API call metadata □ Access logs (AuthN/AuthZ events)
Assess Risk Level	□ High: Sensitive data leaked, offensive content, successful jailbreak □ Medium: Attempted misuse, blocked outputs □ Low: False positive, benign use
Enrich with Threat Intelligence	□ Check IP reputation, file hash (if uploads involved) or □ known attack patterns (MITRE ATLAS, OWASP GenAI)

2. Containment Actions

If	Then
Sensitive Data Exposure	□ Isolate affected GenAI application □ Revoke user access or API key □ Notify DLP/privacy team
Prompt Injection / Jailbreak	□ Block user session □ Tighten prompt validation rules or input filters

Toxic/Harmful Output	☐ Suspend model endpoint if ongoing ☐ Engage PR/legal teams for potential reputation risk
Anomalous Access	☐ Force reauthentication ☐ Lock user account ☐ Check for credential compromise

3. Investigation

Task	How
Replay Session	☐ Review series of prompts and responses leading to incident ☐ Look for escalation patterns
Model Behavior Check	☐ Check if the issue replicates across different sessions/users ☐ Check for drift or poisoning
User Behavior Analysis	☐ Analyze login location, ☐ Device fingerprint ☐ History of use (first time vs recurring pattern)
Data Exposure Scope	☐ Identify if any customer data, internal documents, PII, or trade secrets were leaked
Cross-Reference Logs	Check adjacent security telemetry ☐ endpoint protection ☐ DLP ☐ identity providers (IdP)

4. Eradication & Recovery

Action	Details
Patch Model/Policies	☐ Update prompt injection defenses (eg, input sanitization, output moderation) ☐ Adjust model parameters if needed (temperature, response length)
Update Access Controls	☐ Rotate API keys ☐ Implement tighter RBAC ☐ Tighter session controls
Reset Affected Systems	☐ Restart/redeploy compromised or suspicious AI instances ☐ clear cache/memory if needed
Notify Stakeholders	☐ Internal: Legal, Compliance, Data Privacy, CISO ☐ External: Regulators of breach/reportable incident

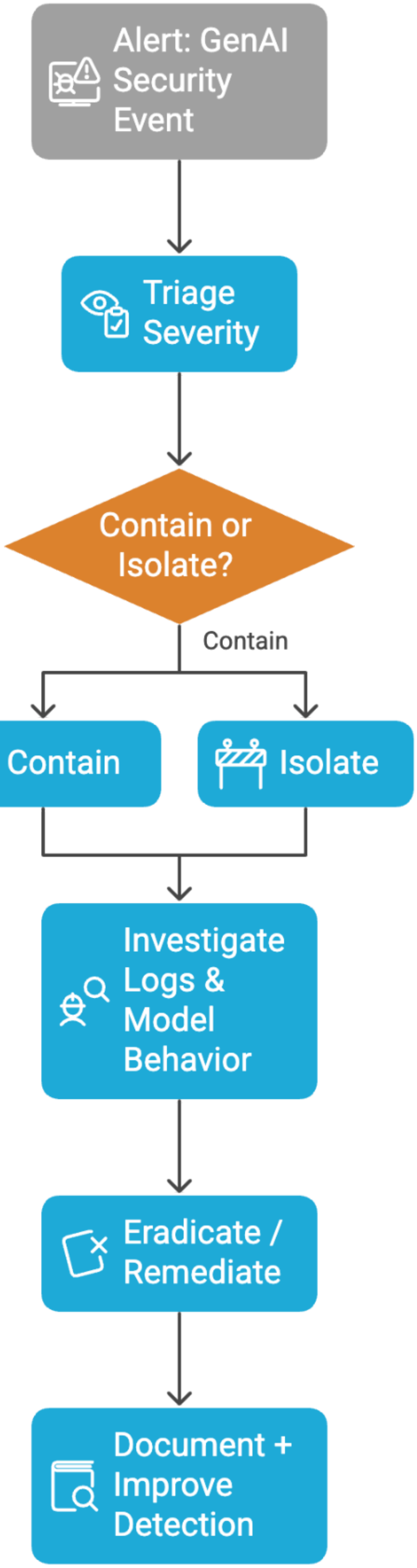
5. Post-Incident Review

Task	Details
Root Cause Analysis	Identify the underlying issue □ Misconfiguration □ Lack of monitoring □ Model vulnerability
Lessons Learned	□ Gaps in detection □ Could earlier alerts have prevented escalation?
Policy/Control Updates	Revise □ Logging □ Filtering □ Access □ Policy
Tabletop Exercise	□ Rehearse a similar GenAI incident scenario quarterly to improve readiness

6. Documentation

Incident report	□ Timeline □ Impact □ Actions taken
Update risk registers	□ If new risks identified
SIEM rule tuning	□ Update detection based on real world events

Runbook Quick Decision Tree


Alert: GenAI Security Event
Triage Severity
Contain or Isolate?
Contain
Contain
Isolate
Investigate Logs & Model Behavior
Eradicate / Remediate
Document + Improve Detection