

# 오픈소스를 활용한 침해사고 대응

클라우드 네이티브 환경에서의 자동화된 침해 사고 대응 및 악성코드 분석 파이프라인 구축

21일 단기 프로젝트

25. 12. 03 ~ 26. 01. 08

# 목차

프로젝트 개요

시스템 아키텍처

핵심 기술 상세 분석

핵심 기능 단위 테스트 및 시연

프로젝트 회고

# 기획 의도

# 프로젝트 개요

목표 및 핵심 기술

구축 목표

핵심 기술

# 기획 의도

클라우드 네이티브 환경(Kubernetes) 확산.

# 기획 의도

## [2025 SOAR 솔루션 리포트] 보안전문가들이 바라마지 않는 ‘자동화’의 길, SOAR가 걷다

입력: 2025-08-11 10:12

한국인터넷진흥원(KISA)은 ‘2025 사이버 위협 동향 보고서-상반기’를 통해 “바이비트와 위믹스의 해킹사건은 마치 성벽이 아무리 튼튼하더라도 성문이 아닌 보급로를 통해 침투하거나, 성안에 있는 보조 건물의 문이 열려있다면 결국 성 전체가 위험에 처할 수 있다”면서, “기업들은 주요 서비스외에도 다른 연결된 서비스에 대한 상시 보안 취약점 점검 조치와 자산에 대한 위협 모니터링을 통해 가시성 확보를 강화해야할 것”이라고 강조했다.

때문에 보안담당자들은 늘 이러한 보안 업무의 문제를 해결하기 위한 솔루션을 꿈꾸는데, 해결책으로 꼽히는 것 중 하나가 바로 ‘보안 오케스트레이션, 자동화 및 대응(SOAR: Security Orchestration, Automation, and Response)’ 솔루션이다.

<https://www.boannews.com/media/view.asp?idx=138647&page=1&kind=3>

# 기획 의도

클라우드 네이티브 환경(Kubernetes) 확산.

런타임 대응 중심으로의 클라우드 보안 패러다임 전환.

탐지-격리-분석-보고의 전 과정 자동화.



**'자동화된 침해 사고 대응 및 악성코드 분석 파이프라인'**

# 구축 목표

## Runtime & Automation 실시간 자율방어

- 침해 행위 실시간 감지.
- 자율 방어 체계 구축.

## YARA & Cuckoo 지능형 교차 분석

- 시그니처 기반 정적 분석.
- 샌드박스 기반 동적 분석.

## S3 & Dashboard 통합 가시성 확보

- 디지털 포렌식 근거 확보.
- 리포팅과 대시보드 시각화.

# 핵심 기술

클라우드

AWS EKS  
쿠버네티스

AWS Step  
Functions  
자동화 대응

오픈소스

Falco  
런타임 행위 탐지

YARA  
패턴 기반 정적 분석

Cuckoo  
격리 기반 동적 분석

# 시스템 아키텍처

## 전체 구성도

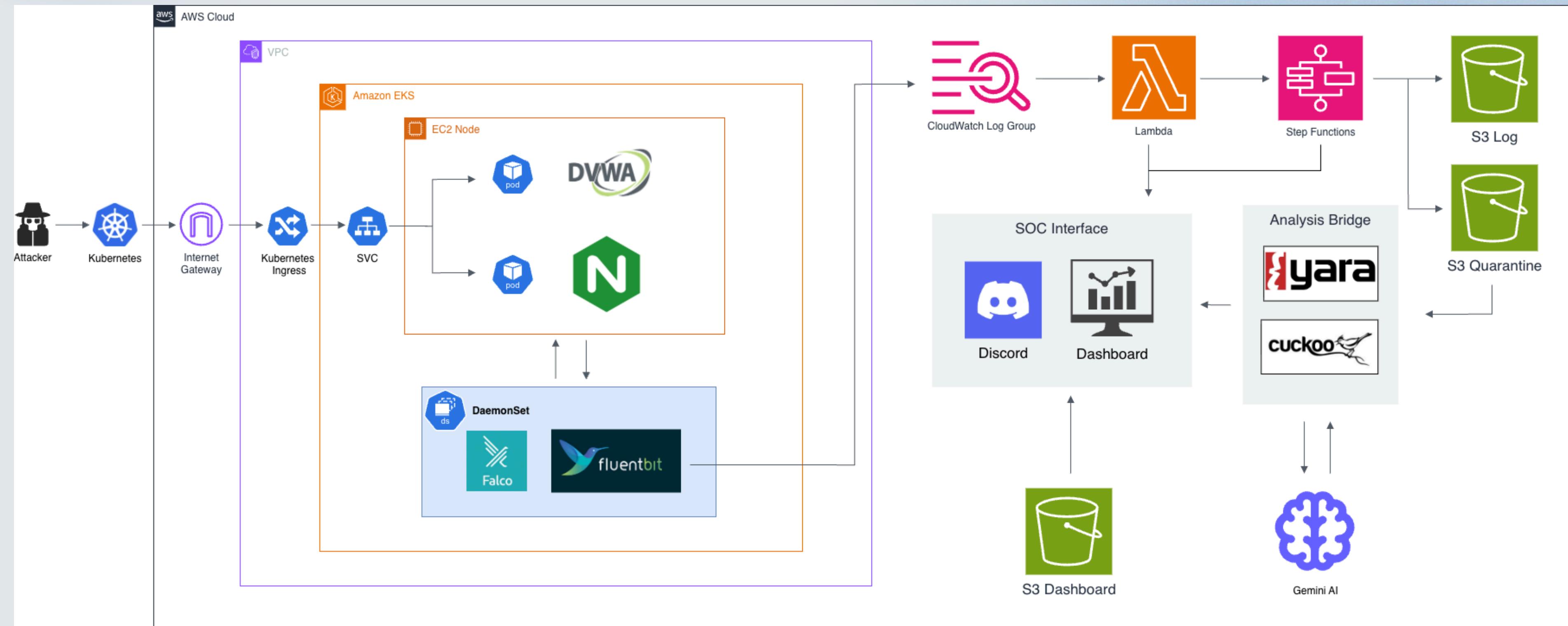
[탐지 및 수집] AWS EKS (Falco) → Fluent Bit / CloudWatch

[대응 및 분석 제어] Lambda → AWS Step Functions (SOAR)

[격리 및 기록] S3 Quarantine / S3 Log

[지능형 심층 분석] YARA / Cuckoo ↔ Gemini AI

# 탐지 및 수집 → 대응 및 분석 제어 → 격리 및 기록 → 지능형 심층 분석 → 시각화 및 알림



# 핵심 기술 상세 분석

4단계 파이프라인 Deep Dive

[탐지 및 수집] 런타임 가시성 극대화.

[대응 및 격리] 서비스 SOAR.

[심층 분석] 하이브리드 포렌식.

[시각화 및 알림] 보안 인텔리전스 가시화.

# [탐지 및 수집] 런타임 가시성 극대화

## 1. DVWA 관련 규칙

**prefix: ARTIFACT\_\* / DVWA\_UPLOAD\_MOVE**

업로드 디렉토리 파일 쓰기

의심 확장자/아카이브 쓰기

chmod

업로드 경로에서 실행/웹서버 쉘 실행

리버스 쉘 실행

wget/curl

tmp→uploads 이동

아웃바운드 연결

**Falco  
Ruleset**

## 2. Container 실행 행위

**prefix: PRJ\_EXEC\_\***

쉘 실행/대화형 쉘

정찰·열거 명령

자격증명·토큰 탐색

다운로드 후 실행

임시 경로 실행

Fetch & Run

아카이브/압축(스테이징)

## [탐지 및 수집] 런타임 가시성 극대화

# Falco Ruleset Example

```

148 - rule: DVWA Webserver Spawns Shell
149   desc: Shell spawned by common webserver/php processes in dvwa namespace (webshell/RCE indicator).
150   condition: >
151     evt.type in (execve, execveat)
152     and dvwa_scope
153     and proc.name in (sh, bash, dash, ash)
154     and proc.pname in (dvwa_webserver_procs)
155   output: >
156     ARTIFACT_WEBHELL reason=webserver_spawns_shell parent=%proc.pname proc=%proc.name cmdline=%proc.cmdline
157     container=%container.name container_id=%container.id
158     pod=%k8s.pod.name ns=%k8s.ns.name node=%evt.hostname user=%user.name
159   priority: WARNING
160   tags: [project, dvwa, webshell, exec]

```

**rule** 규칙의 고유한 이름(식별자)

**desc** 해당 규칙이 무엇을 탐지하는지에 대한 설명

**condition** 이벤트(시스템콜 등)가 규칙에 매칭되는지 판단하는 필터/논리식

**output** 조건이 매칭되면 출력할 알림 메시지 포맷

**priority** 트리거된 이벤트의 심각도, **syslog** 심각도 단계 사용

\* Emergency > Alert > Critical > Error > Warning > Notice > Informational > Debug

## [탐지 및 수집] 런타임 가시성 극대화

# Fluent-Bit / Logs to CloudWatch

```
54 [FILTER]
55   Name grep
56   Match kube.*
57   Regex $kubernetes['namespace_name'] ^falco$ ^falco$
58
59 [FILTER]
60   Name grep
61   Match kube.*
62   Regex $kubernetes['container_name'] ^falco$ ^falco$
```

Fluent-Bit DaemonSet을 사용하여 Falco 규칙에 의해 발생한 로그를 지정된 CloudWatch log group에 저장.

```
66 [FILTER]
67   Name grep
68   Match kube.*
69   Regex log (ARTIFACT_|DVWA_| Terminal shell|Contact K8S|Netcat|System user|Privileged|
```

## [탐지 및 수집] 런타임 가시성 극대화

## CloudWatch log group

```
{  
    "time": "2025-12-17T07:54:01.670868858Z",  
    "stream": "stdout",  
    "_p": "F",  
    "log": "07:53:39.107556628: Notice DVWA_UPLOAD_MOVE src=/tmp/phpEZcDsY dst=/var/www/html/hackable/uploads/test.php proc=apache2 cmdline=apache2 -k start user=w container container_image_repository=docker.io/vulnerables/web-dvwa container_image_tag=latest k8s_pod_name=dvwa-deployment-54885586cd-9k8x6 k8s_ns_name=dvwa",  
    "kubernetes": {  
        "pod_name": "falco-npczm",  
        "namespace_name": "falco",  
        "pod_id": "256e417c-2fcf-45f6-98dd-63be409169b6",  
        "labels": {  
            "app.kubernetes.io/instance": "falco",  
            "app.kubernetes.io/name": "falco",  
            "controller-revision-hash": "954dff78c",  
            "pod-template-generation": "11"  
        },  
        "annotations": {  
            "checksum/certs": "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",  
            "checksum/config": "068724b49b2b91198ca0e440623dd6b5008c352825d2c214388d240174249867",  
            "checksum/rules": "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",  
            "kubectl.kubernetes.io/restartedAt": "2025-12-16T15:15:52+09:00"  
        },  
        "host": "ip-10-0-0-50.ec2.internal",  
        "pod_ip": "10.0.0.29",  
        "container_name": "falco",  
        "docker_id": "1a710caa21f39773926d7d85e71d612710a913403b6613cdb830e1765d8b2c11",  
        "container_hash": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sysdig/main-falco@sha256:85fd8fd43a66382066ff3dd842fe2a520a6a6170f8f2f13192d53b72edfb7dff",  
        "container_image": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sysdig/main-falco:0.41.3-multi"  
    }  
}
```

# 핵심 기술 상세 분석

4단계 파이프라인 Deep Dive

[탐지 및 수집] 런타임 가시성 극대화.

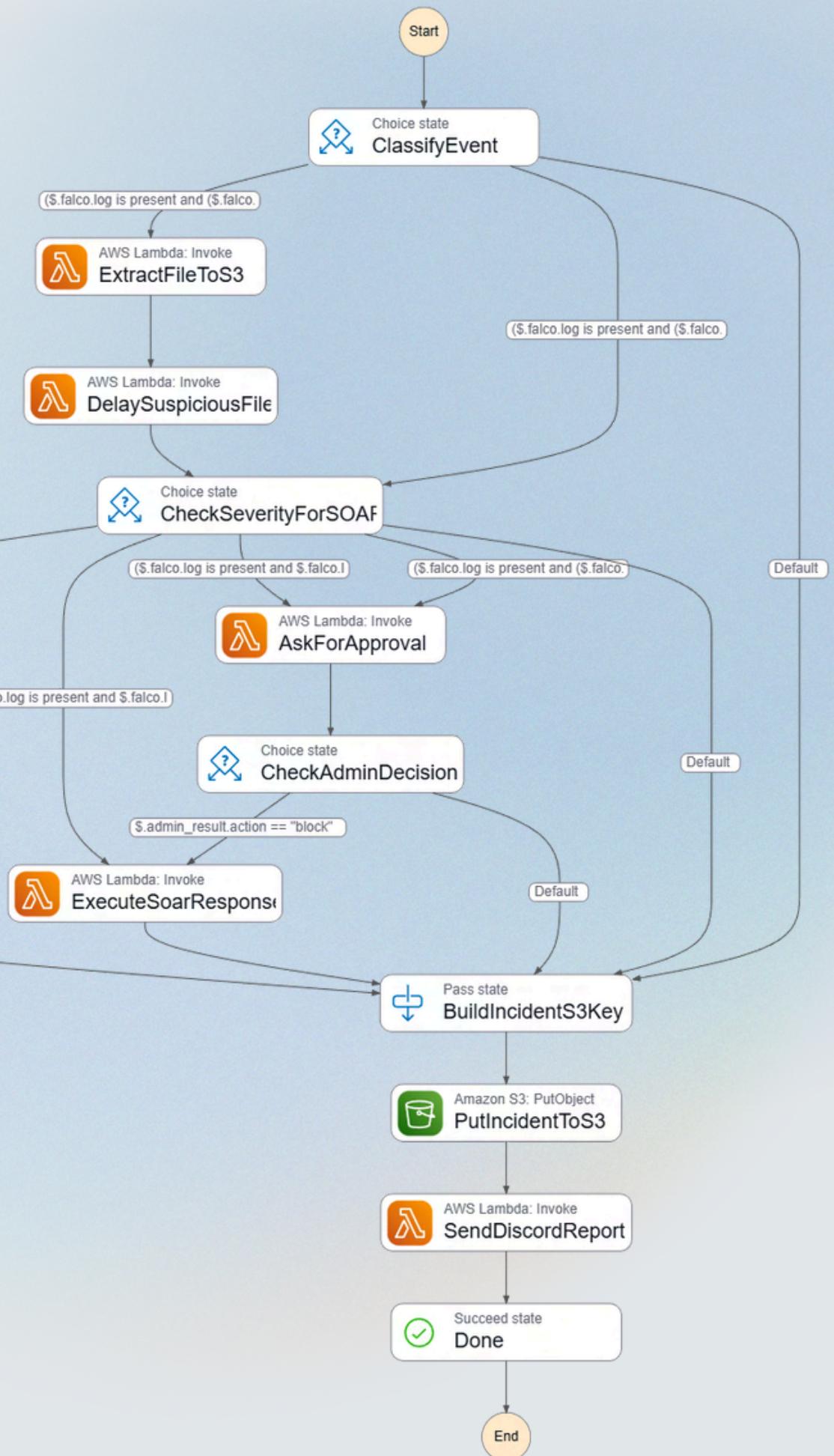
[대응 및 격리] 서비스 SOAR.

[심층 분석] 하이브리드 포렌식.

[시각화 및 알림] 보안 인텔리전스 가시화.

# [대응 및 격리] 서비스 SOAR

## Step Function



# [대응 및 격리] 서비스 SOAR

## Step Function



# [대응 및 격리] 서비스 SOAR

## Step Function

co.log is present and \$.falco.I

관리자 확인을 위한 내용 전송

AWS Lambda: Invoke

Parse

**대응 선택 - 스냅샷, 포드 중지**

\*일부 Shell이나 허가되지 않은 행위는 즉시 차단

AWS Lambda: Invoke

DenyNaclOutboundDenyRule



관리자 선택에 따른 대응 / 통과 여부

Choice state  
CheckAdminDecision

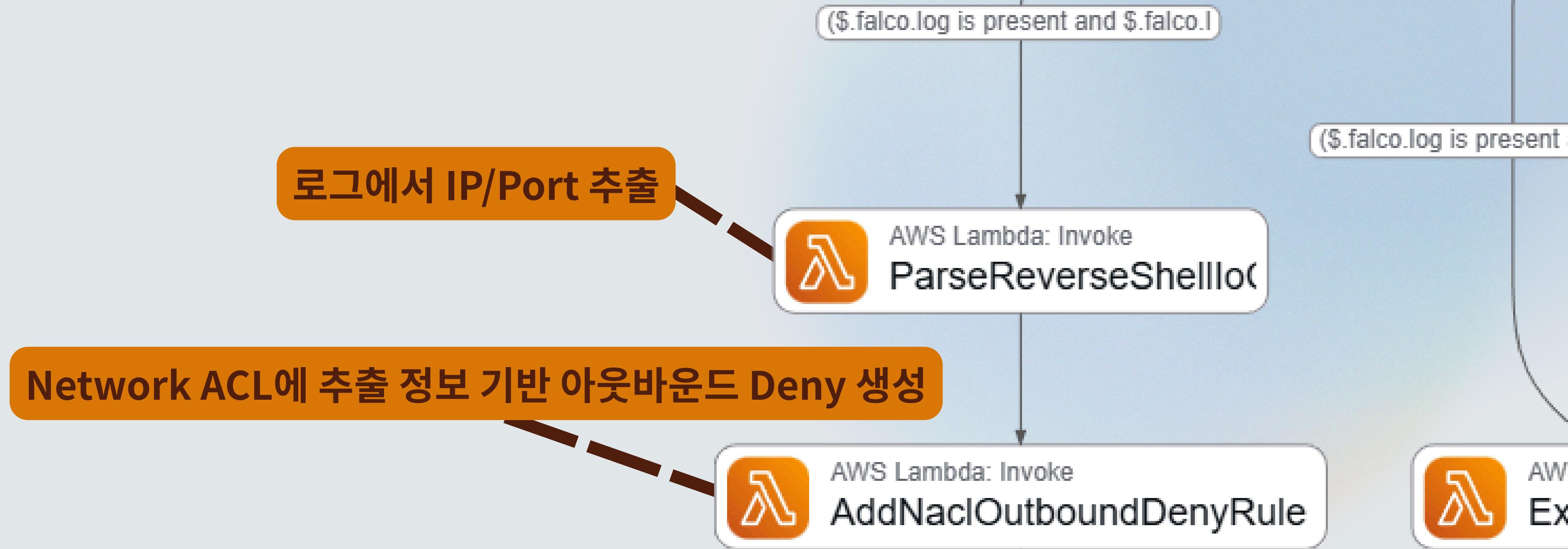
Default

AWS Lambda: Invoke  
ExecuteSoarResponse

Pass state  
BuildIncidentS3Key

## [대응 및 격리] 서비스 SOAR

### Step Function

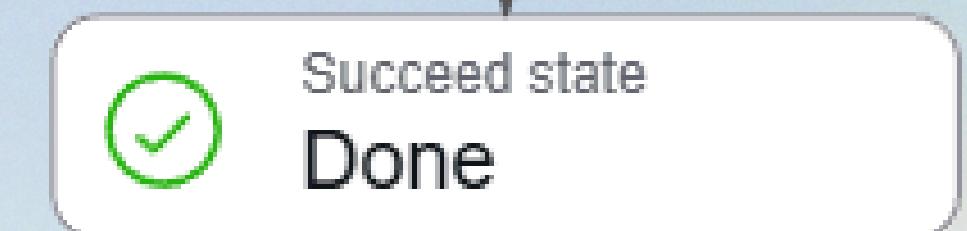
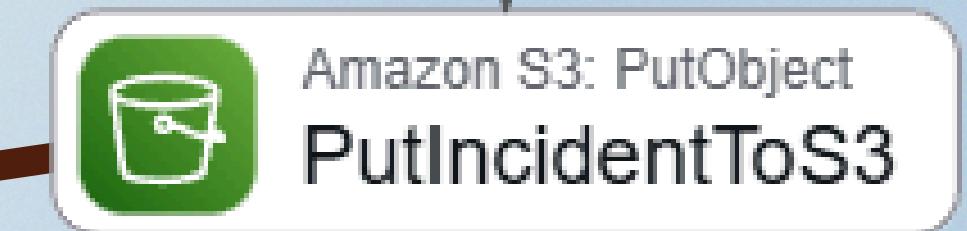


## [대응 및 격리] 서비스 SOAR

# Step Function

전체 row 로그 버킷에 저장(보관용)

최종 보고서 디스코드에 전송



End

## [대응 및 격리] 서비스 SOAR

# Isolate files in S3

Sentinel-security-bucket > file/ > dvwa/ > dvwa-deployment-54885586cd-9k8x6/

### dvwa-deployment-54885586cd-9k8x6/

객체 속성

객체 (2)

객체는 Amazon S3에 저장되어 있는 기본 엔터티입니다. [Amazon S3 인벤토리](#) 를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에

접두사로 객체 찾기

이름	유형	마지막 수정
<a href="#">37f4887ca325_test3.php</a>	php	2025. 12. 17. pm 4:50:42 PM KST
<a href="#">37f4887ca325_test4.php</a>	php	2025. 12. 17. pm 5:05:30 PM KST

## [대응 및 격리] 서비스 SOAR

# Delay Run File & Change Route

**SOAR File Delay - OK**

Falco 파일 이벤트 기반 격리(이동/이름변경) 리포트

Cluster	EventType	Namespace
cluster_1212	DVWA_UPLOAD_MOVE	dvwa

**Pod**  
dvwa-deployment-54885586cd-vl6qk

**Container**  
dvwa-container

**FileName**  
b7b801.js

**SrcPath**  
/var/www/html/hackable/uploads/b7b801.js

**MoveTo**  
/var/www/html/hackable/quarantine/quarantine\_b7b8016b8377\_b7b801.js

**RenameTo**  
quarantine\_b7b8016b8377\_b7b801.js

**QuarantineDir**  
/var/www/html/hackable/quarantine

**SHA256**  
b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311427c4a289

**Sentinel Incident Report - NOTICE - DVWA\_UPLOAD\_MOVE**

**Detection**

**Rule:** DVWA\_UPLOAD\_MOVE  
**Severity:** NOTICE  
**Time:** 2026-01-06T05:33:25.577687449Z  
**Node:** ip-10-0-0-9.ec2.internal  
**K8S:** ns=falco pod=falco-tsvjh

**File Artifact**  
**Status:** Extracted  
**Path:** s3://quarantine-cuckoo/b7b801.js

**SOAR Response**  
Skipped / Not executed

**Raw Log**  
05:33:20.545357633: Notice DVWA\_UPLOAD\_MOVE src=/tmp/phpuALADF dst=/var/www/html/hackable/uploads/b7b801.js proc=apache2 cmdline=apache2 -k start user=www-data container=dvwa-container pod=dvwa-deployment-54885586cd-vl6qk ns=dvwa container\_id=71249ae270b3 container\_name=dvwa-container container\_image\_repository=docker.io/vulnerables/web-dvwa container\_image\_tag=latest k8s\_pod\_name=dvwa-deployment-54885586cd-vl6qk k8s\_ns\_name=dvwa

## [대응 및 격리] 서비스 SOAR

# Respond to Tasks within the Server

**Suspicious Activity Detected! Approval Required.**

**Admin Intervention Needed**

**Event Log:**

```
10:06:05.784220410: Warning PRJ_EXEC_RECON proc=cat
cmdline=cat reverse-shell-final.php user=root
container=dvwa-container container_id=0ec634e1abfe
pod=dvwa-deployment-54885586cd-ffh9c ns=dvwa node=ip-10-0-0-30.ec2.internal container_id=0ec634e1abfe
container_name=dvwa-container
container_image_repository=docker.io/vulnerables/web-dvwa
container_image_tag=latest k8s_pod_name=dvwa-deployment-54885586cd-ffh9c k8s_ns_name=dvwa
```

**5 Sentinel Incident Report - WARNING - PRJ\_EXEC\_RECON**

Admin: allow

**Detection**

**Rule:** PRJ\_EXEC\_RECON

**Severity:** WARNING

**Time:** 2026-01-05T10:07:17.83794268Z

**Node:** ip-10-0-0-30.ec2.internal

**K8S:** ns=falco pod=falco-r428p

**Admin Approval**

**Action:** allow

**Message:** 승인 처리되었습니다 (Access Allowed)

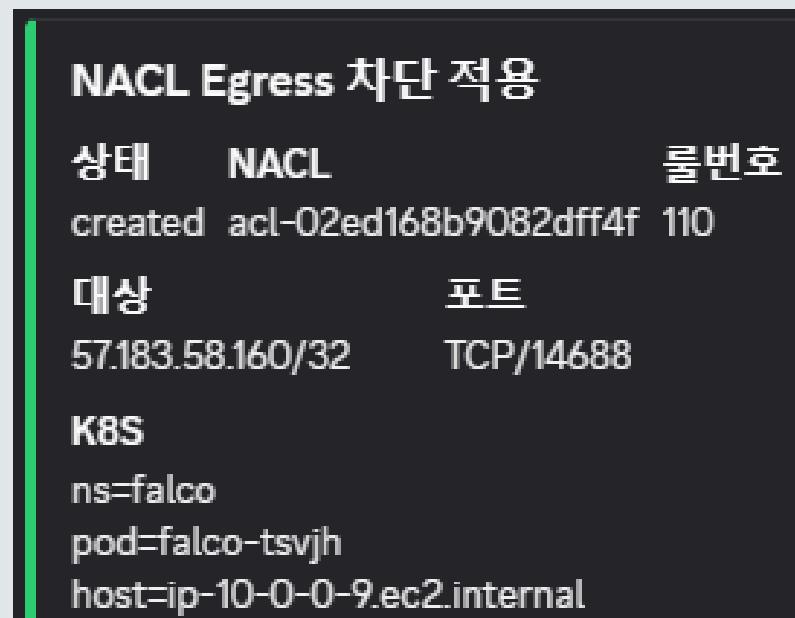
Name	인스턴스 ID	인스턴스 상태
FORENSIC-ISOLATED-i-0263cb193cbf112be	i-0263cb193cbf112be	중지됨
FORENSIC-ISOLATED-i-0bdd8ec3a98d63af3	i-0bdd8ec3a98d63af3	중지됨
	i-05a3abef1ef5fd84c	실행 중

스냅샷 (4) 정보			
Last updated about 3 hours ago			
내 소유	검색	휴지통	작업
스냅샷 ID	전체 스냅샷 크기	볼륨 크기	설명
snap-077c5ba4df0975813	6.7 GiB	20 GiB	[Falco Incident] Forensic snapshot for i-00f285f89b...
snap-0647dc10338319283	6.7 GiB	20 GiB	[Falco Incident] Forensic snapshot for i-00f285f89b...
snap-062b7dab88870773d	6.7 GiB	20 GiB	[Falco Incident] Forensic snapshot for i-00f285f89b...
snap-073ab92975c011b85	6.49 GiB	20 GiB	[Falco Incident] Forensic snapshot for i-0263cb193c...

## [대응 및 격리] 서비스 SOAR

# Reverse Shell Response



아웃바운드 규칙 (4)					
<input type="text"/> Filter outbound rules					
규칙 번호	유형	프로토콜	포트 범위	대상	허용/거부
100	사용자 지정 TCP	TCP(6)	17241	57.181.84.1/32	<span style="color: red;">✖ Deny</span>
110	사용자 지정 TCP	TCP(6)	14688	57.183.58.160/32	<span style="color: red;">✖ Deny</span>

# 핵심 기술 상세 분석

4단계 파이프라인 Deep Dive

[탐지 및 수집] 런타임 가시성 극대화.

[대응 및 격리] 서비스 SOAR.

[심층 분석] 하이브리드 포렌식.

[시각화 및 알림] 보안 인텔리전스 가시화.

# [심층 분석] 하이브리드 포렌식

1  
파일 유형  
식별 기반  
YARA 정  
적 탐지

```

meta:
  description = "Windows PE 파일 식별 (internal)"
  type = "pe"
strings:
  $mz = { 4D 5A }
condition:
  $mz
}

rule FileType_ELF {
meta:
  description = "Linux ELF 파일 식별 (internal)"
  type = "elf"
strings:
  $elf = { 7F 45 4C 46 }
condition:
  $elf
}

rule FileType_Text {
meta:
  description = "일반 텍스트 파일 식별 (internal)"
  type = "text"
condition:
  filesize < 5MB and uint8(0) != 0
}

```

# 정적 분석 (YARA Ruleset)

```

rule Linux_Reverse_Shell
{
meta:
  description = "리눅스 Reverse Shell 패턴 탐지"
  severity = "critical"

strings:
  $bash_tcp    = "/dev/tcp/"
  $nc_rev     = /nc\s+-e\s+\bin\bash/
  $python_r   = "python -c"
  $bash_i     = "bash -i >& /dev/tcp"
  $bash_c     = "bash -c"

condition:
  FileType_Text and any of them
}

```

2 리눅스 원격 쉘 공격 패턴 탐지

## [심층 분석] 하이브리드 포렌식

```
rule Malicious_Base64_Encoded
{
    meta:
        description = "난독화 Base64 + 실행 코드"
        severity = "medium"

    strings:
        $eval      = /eval\s*\(/i
        $exec      = /exec\s*\(/i
        $decode    = /base64(i|_decode)?/i
        $b64_payload = /[A-Za-z0-9+\//]{40,}{0,2}/

    condition:
        FileType_Text and $b64_payload and any of ($eval, $exec, $decode)
}
```

### ③ Base64 인코딩 + eval/exec 실행 시도 탐지

## 정적 분석 (YARA Ruleset)

```
rule Suspicious_PowerShell_Command
{
    meta:
        description = "PowerShell 스크립트 의심 명령"
        severity = "high"

    strings:
        $bypass    = /ExecutionPolicy\s+Bypass/i
        $hidden    = /WindowStyle\s+Hidden/i
        $encoded   = /-enc\s+[A-Za-z0-9+\//]{10,}/i
        $base64    = /[A-Za-z0-9+\//]{20,}{0,2}/

    condition:
        FileType_Text and any of them
}
```

### ④ Powershell 기반 악성 스크립트 우회 및 난독화 패턴 탐지

# [심층 분석] 하이브리드 포렌식

## 정적 분석 (YARA Ruleset)

AI Sentinel 앱 오후 12:09 이동하기

분석 보고: reverse-shell-final5.php

위험도 점수  
6.6 / 10.0

YARA 탐지

- php\_reverse\_shell (Sev: Warning)
- php\_reverse\_shell\_2 (Sev: Warning)
- WebShell\_\_findsock\_php\_findsock\_shell\_
- php\_reverse\_shell (Sev: Critical)

PHP 웹쉘 3종 (서버 제어권 탈취, 데이터 유출 )

AI Sentinel 앱 2025-12-18 오전 9:54 이동하기

분석 보고:  
[b7b8016b837766fc9a8d6cfeec6239c](#)  
[05778eec6525bc61327b6311427c4a2](#)  
[89.js](#)

위험도 점수  
6.2 / 10.0

YARA 탐지

- FileType\_Text (Sev: Informational | Tags: static\_analysis)
- Suspicious\_PowerShell\_Command (Sev: High | Tags: static\_analysis)

Powershell 명령 실행 악성 스크립트

# [심층 분석] 하이브리드 포렌식

# 동적 분석 (Cuckoo Sandbox)

```

Ubuntu 64-bit Arm Server 20.04.5
Dec 18 00:38
yooju@yooju: ~
yooju@yooju: ~ x yooju@yooju: ~ x yooju@yooju: ~ x yooju@yooju: ~ x
yooju@yooju:~$ cuckoo -d
[...]
Cuckoo Sandbox 2.0.7
www.cuckoosandbox.org
Copyright (c) 2010-2018

2025-12-18 00:37:36,269 [cuckoo] DEBUG: Increasing resource limit for number of open files to 104
2025-12-18 00:37:36,274 [cuckoo.core.database] DEBUG: Using database-wide lock for sqlite
2025-12-18 00:37:36,350 [cuckoo.core.startup] DEBUG: Imported modules...
2025-12-18 00:37:36,364 [cuckoo.core.startup] DEBUG: Imported "auxiliary" modules:
2025-12-18 00:37:36,364 [cuckoo.core.startup] DEBUG: |-- MITM
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Reboot
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Replay
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Services
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Sniffer
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: Imported "machinery" modules:
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- vSphere
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- KVM
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- ESX
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- XenServer
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- VirtualBox
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Avd
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- QEMU
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- VMware
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- Physical
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: Imported "processing" modules:
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- AnalysisInfo
2025-12-18 00:37:36,365 [cuckoo.core.startup] DEBUG: |-- ApkInfo

```

이름	유형	마지막 수정	크기	스토리지 클래스
b7b8016b837766fc	js	2025. 12. 18. am 9:39:15 AM KST	3.8MB	Standard
9a8d6cfeec6239c05				
778eec6525bc6132				
7b6311427c4a289.j				

① Cuckoo 실행 화면

② S3 버킷 파일 업로드

# [심층 분석] 하이브리드 포렌식

YARA 룰 로딩 완료: rules.yar  
AI+Cuckoo 통합 감시 시작 (Local & Remote YARA Merge Mode)

발견: b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311427c4a289.js  
다운로드: b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311427c4a289.js  
로컬 YARA 탐지: ['FileType\_Text', 'Suspicious\_PowerShell\_Command']  
분석 시작...  
대시보드 업데이트 (yara\_complete)  
대시보드 업데이트 (analyzing)  
대시보드 업데이트 (completed)  
Cuckoo 분석 완료.  
분석 끝! 점수: 6.2  
최종 YARA 정보 (로컬 2 + Cuckoo 0): 총 2건  
Gemini 요약 중...  
디스코드 전송 결과: 204  
대시보드 업데이트 (WARNING)

## ③ 업로드된 파일 감지

# 동적 분석 (Cuckoo Sandbox)

The screenshot shows the Cuckoo Sandbox analysis interface. At the top, it says "Ubuntu 64-bit Arm Server 20.04.5" and "Dec 18 00:43". The main window title is "Cuckoo Sandbox". The URL in the address bar is "localhost:8000/analysis/1/summary/". The interface includes a sidebar with various icons for file operations like upload, download, and analysis. The main content area is titled "Summary" and contains the following information:

- File:** b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311427c4a289.js
- Summary:**
  - Size: 3.8MB
  - Type: ASCII text, with very long lines, with CRLF line terminators
  - MD5: 934519a23b703d07355c354b8f0ef991
  - SHA1: 8a5f456d2e379a75c20ed74bb4651e21427195b1
  - SHA256: b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311427c4a289
  - SHA512: Show SHA512
  - CRC32: 7B1F26E8
  - ssdeep: None
  - Yara: None matched
- Score:** This file is **very suspicious**, with a score of **6.2 out of 10!**

## ④ 파일 분석 리포트

# 핵심 기술 상세 분석

4단계 파이프라인 Deep Dive

[탐지 및 수집] 런타임 가시성 극대화.

[대응 및 격리] 서비스 SOAR.

[심층 분석] 하이브리드 포렌식.

[시각화 및 알림] 보안 인텔리전스 가시화.

# [시각화 및 알림] 보안 인텔리전스 가시화

## 대시보드 (Dashboard)

INTELLIGENCE ANALYSIS

GEMINI AI SUMMARY

1. 식별: 시스템 정보를 수집하고 가상 환경 탐지 및 분석을 회피하며, 자체 압축 해제 후 악성 스크립트 실행, 의심스러운 프로세스 생성 및 지속성을 확보하려 함.  
 2. 위험: 시스템 제어권 탈취, 중요 정보 유출 및 추가 악성코드 감염을 유발할 수 있는 중간 위험도 악성코드임.  
 3. 대응: 감염된 시스템을 네트워크에서 격리하고, 관련 프로세스 및 파일을 즉시 제거한 뒤 백신 등으로 정밀 검사 및 치료를 진행해야 함.

### ① 파일별 Gemini 위협 분석 요약

AI 5sentinel 앱 오전 9:54

분석 보고:  
[b7b8016b837766fc9a8d6cfeec6239c05778eec6525bc61327b6311](#)  
[427c4a289.js](#)

위험도 점수  
**6.2 / 10.0**

YARA 탐지  
**FileType\_Text** (Sev: Informational | Tags: static\_analysis)  
**Suspicious\_PowerShell\_Command** (Sev: High | Tags: static\_analysis)

AI 요약 및 상태

1. 식별: 시스템 정보를 수집하고 가상 환경 탐지 및 분석을 회피하며, 자체 압축 해제 후 악성 스크립트 실행, 의심스러운 프로세스 생성 및 지속성을 확보하려 함.  
 2. 위험: 시스템 제어권 탈취, 중요 정보 유출 및 추가 악성코드 감염을 유발할 수 있는 중간 위험도 악성코드임.  
 3. 대응: 감염된 시스템을 네트워크에서 격리하고, 관련 프로세스 및 파일을 즉시 제거한 뒤 백신 등으로 정밀 검사 및 치료를 진행해야 함.

Task ID: 1

### ② 위협 분석 보고서 디스코드 알림 전송

# [시각화 및 알림] 보안 인텔리전스 가시화

```

<html lang="ko">
<body class="h-screen w-screen relative">
<script type="text/babel">
    const RadarPanel = ({ data, isAnalyzing, isBlocked }) => [
        const getThreatStatus = (score) => {
            if (!data || score === undefined || score === null) {
                return { status: 'UNKNOWN', colorClass: 'text-slate-400', glowClass: '' };
            }

            let status, color, glow;

            if (score >= 9.0) { status = 'CRITICAL'; color = 'text-[var(--alert))'; glow = 'shadow-[0_0_15px_var(--alert))'; }
            else if (score >= 7.0) { status = 'HIGH'; color = 'text-red-400'; glow = 'shadow-[0_0_15px_rgba(255,0,0,0.7))'; }
            else if (score >= 4.0) { status = 'MEDIUM'; color = 'text-amber-400'; glow = 'shadow-[0_0_10px_rgba(255,193,7,0.7))'; }
            else if (score > 0.0) { status = 'LOW'; color = 'text-[var(--primary))'; glow = 'shadow-[0_0_10px_var(--primary))'; }
            else { status = 'CLEAN'; color = 'text-[var(--success))'; glow = 'shadow-[0_0_10px_var(--success))'; }

            return { status, colorClass: color, glowClass: glow, rawScore: score };
        };

        const threat = getThreatStatus(data?.score);
        const isYaraComplete = data?.status === 'yara_complete';

        const displayStatus = isYaraComplete ? 'YARA COMPLETE (Cuckoo PENDING)' :
            isAnalyzing && data?.status === 'SCANNING' ? 'ANALYZING' : threat.status;
        const displayScore = isYaraComplete || (isAnalyzing && data?.status === 'SCANNING') ? '...' : data?.score;
        const displayColorClass = isYaraComplete || (isAnalyzing && data?.status === 'SCANNING') ? 'text-[var(--primary))' : threat.colorClass;

        return (
            <div className="glass-box rounded-2xl p-0 flex flex-col relative overflow-hidden h-full">
                <div className="p-5 border-b border-white/10 flex justify-between items-center bg-black/40">
                    <span className="text-xl font-bold text-[var(--primary)] tracking-widest flex items-center">
                        <i className="fas fa-satellite-dish mr-3"></i> THREAT RADAR
                    </span>
                    {(isAnalyzing || isYaraComplete) && <span className="text-xs text-red-500 animate-pulse font-bold tracking-wider">TRACKING...</span>}
                </div>

                <div className="flex-1 flex flex-col items-center justify-center p-8 relative">
                    <div className="w-64 h-64 border border-slate-600 rounded-full relative flex items-center justify-center bg-[radial-gradient(white 50%, transparent 50%)]">
                        <div className="absolute inset-0 rounded-full border border-slate-700 scale-75"></div>
                        <div className="absolute inset-0 rounded-full border border-slate-700 scale-50"></div>
                        <div className="absolute w-[1px] h-full bg-slate-700"></div>
                        <div className="absolute h-[1px] w-full bg-slate-700"></div>
                    </div>
                    {isAnalyzing && <div className="absolute inset-0 radar-sweep"></div>}
                    {data && !isAnalyzing && (
                        <div className={`absolute w-4 h-4 rounded-full animate-ping ${threat.colorClass} ${threat.glowClass}`}
                            style={{ top: '30%', right: '30%' }}></div>
                    )}
                </div>
            </div>
        );
    ];
}

```

## ③ 실시간 관제 대시보드 UI/UX 구현

# 대시보드 (Dashboard)

선택	이름	유형	마지막 수정	크기	스토리지 클래스
<input checked="" type="checkbox"/>	index.html	html	2025. 12. 17. pm 5:48:33 PM KST	42.9KB	Standard
<input type="checkbox"/>	latest.json	json	2025. 12. 16. pm 4:59:08 PM KST	1.8KB	Standard
<input type="checkbox"/>	reports/	폴더	-	-	-

## 4 S3 버킷 정적 웹 호스팅으로 배포

# 핵심 기능 단위 테스트 및 시연

영상 시연

unit\_test — falco\_demo\_recorder.py      Open Agent Manager

falco\_demo\_recorder.py ●

```

240     class CommandInjectionScenario(TestScenario):
241         """Command Injection 공격 시나리오"""
242         def __init__(self, name: str, description: str, payload: str, cmd_desc: str):
243             super().__init__(name, description)
244             self.payload = payload
245             self.cmd_desc = cmd_desc
246
247             def execute(self, browser: BrowserService) -> bool:
248                 print(f"\n[{self.description}] 실행 중...")
249                 print(f"    Payload: {self.cmd_desc}")
250                 try:
251                     browser.driver.get(f"{browser.config.dvwa_url}/vulnerabilities/exec/")
252                     time.sleep(1)
253                     browser.take_screenshot(f"{self.name}_before")
254
255                     input_field = browser.wait.until(EC.presence_of_element_located((By.NAME, "ip")))
256                     input_field.clear()
257
258                     # 보안 레벨에 따른 페이로드 조립
259                     separator = ";" if browser.config.security_level == "low" else "&&"
260                     if browser.config.security_level not in ["low", "medium"]:
261                         separator = "|"
262                     full_payload = f"127.0.0.1{separator} {self.payload}"

```

문제   출력   디버그 콘솔   터미널   포트

(.venv) glory1994@GloryLeeui-MacBookAir unit\_test % ./run\_demo.sh

🔧 TEST CONFIGURATION (Default values in brackets)

🌐 DVWA URL [http://54.160.7.208:31100]: █

×

✖ 0 △ 0   ✎   화면 읽기 프로그램이 최적화됨   줄 6, 열 17   공백: 4   UTF-8   LF   Python 3.13.11 ('.venv': venv)   Pyrefly (error-off)   Antigravity - Settings   🔍

# 프로젝트 회고

## 좋았던 점

**Step Function**을 활용하여 공격을 파악한 후  
상황에 맞게 대응할 수 있었음.

S3 저장소를 분리하여 악성파일들을 분석할 수 있었음.

고도화된 공격에 대응하기 위해 규칙 설정  
이 필요하면, 커스텀 규칙을 수정하여 바로  
적용시키고 대응할 수 있었음.

Falco 초기 연결 시 잖은 오류가 발생하였는데,  
잘 극복하였음.

오탐 발생 원인을 분석하며 룰 설계의 중요성을  
인식하게 되었음.

성격이 다른 보안 도구들을 하나의 파이프라인으로  
엮어 통합 플랫폼을 구축해 볼 수 있어서  
의미 있었음.

# 프로젝트 회고

## 아쉬운 점

시간 제약으로 인해 모든 시나리오를 충분히 확장하지 못한 부분이 아쉬움.

아웃바운드 대응을 하더라도 IP 변경 후 공격이 가능.  
IP를 계속 변경하는 공격을 시도하면 대응 방안이 무색해짐.

악성파일이 업로드 된 직후 바로 대응이 되지 않아 일부 공격은 잠시 실행되었음.

Falco 커스텀 규칙 별 심각도 기준을 명확하게 세우지 못함.

로드 밸런서...ㅠㅠ

Cuckoo Sandbox 가상머신의 잖은 오류가 많이 아쉬웠음. AWS의 베어메탈 서버를 활용하거나 CAPE Sandbox를 사용했다면 어땠을지 궁금함.

모두 고생하셨습니다.

감사합니다 .

Q&A  
질문 답변

피드백