네트워크 인프라 구축

서버 관리, 방화벽, 데이터베이스 운영, 네트워크 분석을 아우르는 종합 관리 현황 및 향후 계획

이름 : 송지연

작성일자 : 2025.11.03

시스템 구성요소 및 인프라 요구사항

1. 주요 시스템 구성요소 및 요구사항

구분	주요 구성	요구사항 및 보안 대책
웹 서버	Ubuntu Linux 기반 Apache 서비스 DMZ 영역에 구축	고가용성 클러스터 유형 SSL/TLS 암호화 적용 , WAF (웹 애플리케이션 방화벽) 연동.
데이터베이스 서버	MariaDB (관계형 DB) 내부망에 설치	데이터 암호화 (저장/전송 시) , 최소한의 접근 권한 제어 (TCP 3306 포트).
네트워크 스위치	Cisco Catalyst C3650 모델	VLAN 10/20 설정 및 인터-VLAN 라우팅 지원 , 포트 보안 MAC 주소 필터링 기능 적용.

시스템 구성요소 및 인프라 요구사항

2. 핵심 보안 인프라 및 기능 요구사항

🛡️ 보안 인프라 (Firewall, Log, Analysis)

방화벽 (AhnLab TrusGuard) INT(내부), DMZ, EXT(외부) 존 기반 정책을 적용하고
인바운드/아웃바운드 트래픽 및 애플리케이션 레벨 필터링을 제어

시스템 로그 서버
Ubuntu Linux 환경에서 실시간 보안 로그를 수집 및 분석하며, 로그의 무결성을 보장

패킷 분석 도구 (Wireshark)
DMZ와 웹 서버 간의 트래픽을 분석하고 이상 징후를 탐지하여 네트워크 모니터링을 강화

🔒 기능 요구사항 및 보안 정책

영역	주요 보안 기능
인증/권한 관리	다중 인증(MFA), 역할 기반 접근 제어(RBAC), 세션 관리 및 타임아웃 적용.
암호화	데이터 전송 암호화 (TLS 1.3 이상) 및 저장 암호화 (AES-256) 구현.
네트워크 정책	정적 NAT (웹 서버용) 및 동적 NAT (내부 사용자용) 설정 , 최소 권한 원칙 기반의 존(Zone) 정책 운영.
서버 보안	SSH root 직접 로그인 금지 및 일반 사용자에게 sudo 권한을 위임하여 관리.

시스템 구성요소 및 인프라 요구사항

3. 구축 목표 및 관리체계

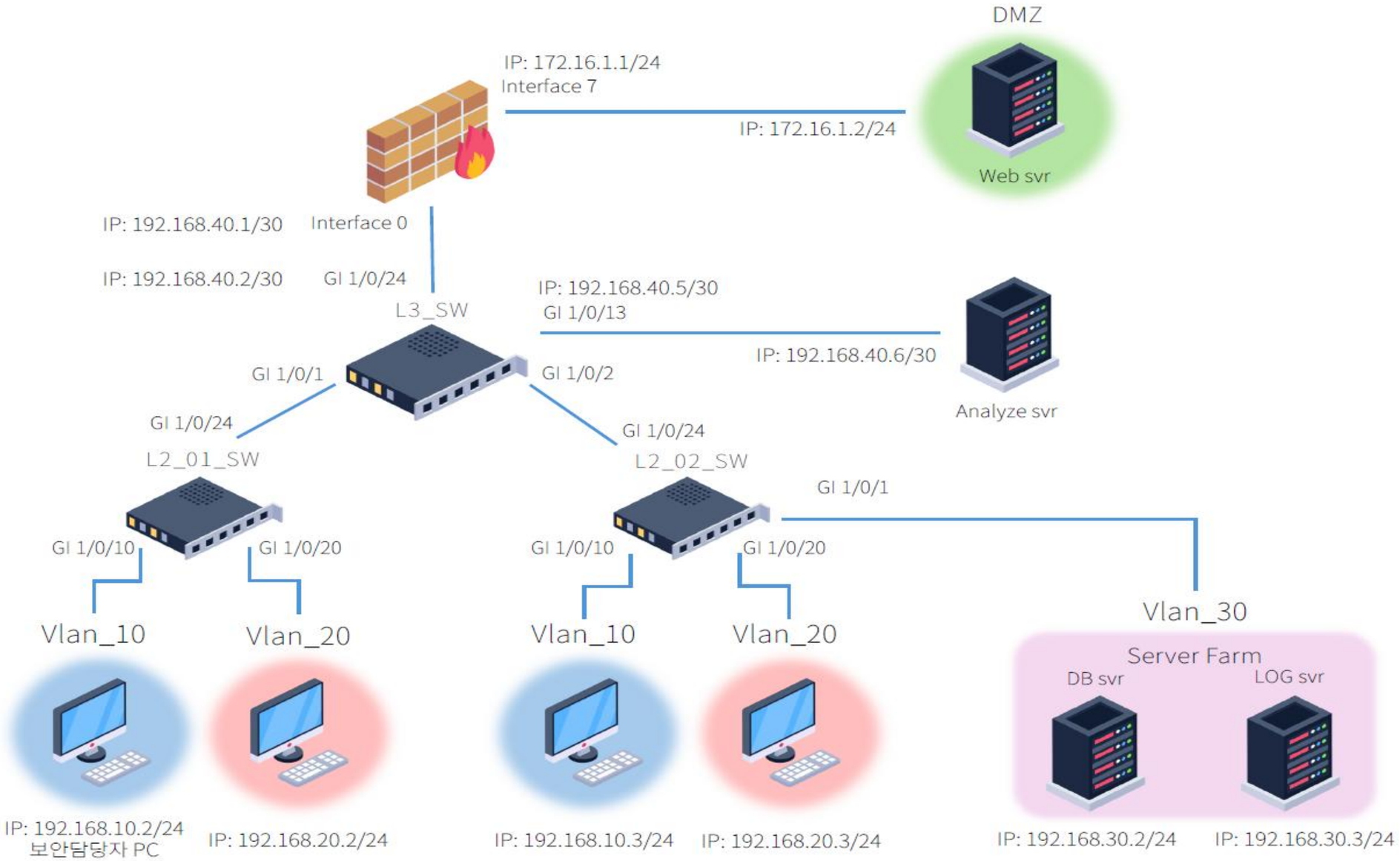
보안 구축 목표
제로 트러스트 보안 모델 구현 , 다층 보안 방어체계 구축 , 보안 사고 대응 시간 30분 이내 달성

네트워크 관리
VLAN 라우팅을 활성화하고 포트 미러링(SPAN) 설정을 통해 트래픽을 감시하여
네트워크 상태 및 보안 분석 기반을 마련

접근 통제
방화벽에서 INT, DMZ, EXT 3단계로 자산을 구분하고
모든 트래픽은 명시적 허용 시에만 통과하는 최소 권한 원칙을 준수

지속적 강화 계획
향후 VPN 구축과 보안 계층 다각화 계획, 데이터베이스 백업·복구 정책 수립 등을 통해
지속적으로 보안 역량을 강화할 예정

네트워크 구성도


DMZ
IP: 172.16.1.1/24
Interface 7
IP: 172.16.1.2/24
Web svr
IP: 192.168.40.1/30
Interface 0
IP: 192.168.40.2/30
GI 1/0/24
L3_SW
IP: 192.168.40.5/30
GI 1/0/13
IP: 192.168.40.6/30
Analyze svr
GI 1/0/1
GI 1/0/2
GI 1/0/24
L2_01_SW
GI 1/0/24
L2_02_SW
GI 1/0/1
GI 1/0/10
GI 1/0/20
GI 1/0/10
GI 1/0/20
Vlan_30
Vlan_10
Vlan_20
Vlan_10
Vlan_20
Server Farm
DB svr
LOG svr
IP: 192.168.10.2/24
보안담당자 PC
IP: 192.168.20.2/24
IP: 192.168.10.3/24
IP: 192.168.20.3/24
IP: 192.168.30.2/24
IP: 192.168.30.3/24

네트워크 장비 초기화 단계별 절차

Cisco Catalyst C3650 스위치 및 AhnLab TrusGuard 50B 방화벽의 체계적 초기화 과정으로 네트워크 안정성 및 보안 재구성 기반 마련

01

장비 콘솔 연결

각 장비의 콘솔 포트에 연결하여 초기화 명령어 입력 준비

02

초기화 명령어 실행

Cisco Catalyst C3650 및 TrusGuard 방화벽에 설정 및 VLAN 정보를 완전 삭제하는 명령어 입력

03

설정 및 VLAN 정보 삭제 확인

초기화 명령 실행 후 모든 사용자 정의 설정과 VLAN이 삭제되었는지 확인

04

장비 재시작

초기화된 기본 환경 상태를 반영하기 위해 장비를 재부팅

05

초기화 완료 및 네트워크 정상화

네트워크 안정화와 보안 정책 재구성을 위한 초기화 완료 상태 확인

VLAN 라우팅 및 모니터링 구축결과

VLAN 10/20 IP 할당 및 port Mirroring 설정

```
L3_SW(config)#monitor session 1 source interface GigabitEther1/0/24
L3_SW(config)#monitor session 1 destination interface GigabitEthernet1/0/13
L3_SW(config)#do show ip int br
Interface              IP-Address      OK? Method Status                Protocol
Vlan1                  unassigned      YES unset  administratively down down
Vlan10                 192.168.10.1    YES manual up                    up
Vlan20                 192.168.20.1    YES manual up                    up
Vlan30                 192.168.30.1    YES manual up                    up
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet1/0/1   unassigned      YES unset  up                    up
GigabitEthernet1/0/2   unassigned      YES unset  up                    up
GigabitEthernet1/0/3   unassigned      YES unset  down                  down
GigabitEthernet1/0/4   unassigned      YES unset  down                  down
GigabitEthernet1/0/5   unassigned      YES unset  down                  down
GigabitEthernet1/0/6   unassigned      YES unset  down                  down
GigabitEthernet1/0/7   unassigned      YES unset  down                  down
GigabitEthernet1/0/8   unassigned      YES unset  down                  down
GigabitEthernet1/0/9   unassigned      YES unset  down                  down
GigabitEthernet1/0/10  unassigned      YES unset  down                  down
GigabitEthernet1/0/11  unassigned      YES unset  down                  down
GigabitEthernet1/0/12  unassigned      YES unset  down                  down
GigabitEthernet1/0/13  192.168.40.5    YES manual up                    down
GigabitEthernet1/0/14  unassigned      YES unset  down                  down
GigabitEthernet1/0/15  unassigned      YES unset  down                  down
GigabitEthernet1/0/16  unassigned      YES unset  down                  down
GigabitEthernet1/0/17  unassigned      YES unset  down                  down
GigabitEthernet1/0/18  unassigned      YES unset  down                  down
GigabitEthernet1/0/19  unassigned      YES unset  down                  down
GigabitEthernet1/0/20  unassigned      YES unset  down                  down
GigabitEthernet1/0/21  unassigned      YES unset  down                  down
GigabitEthernet1/0/22  unassigned      YES unset  down                  down
GigabitEthernet1/0/23  unassigned      YES unset  down                  down
GigabitEthernet1/0/24  192.168.40.2    YES manual up                    up
GigabitEthernet1/1/1   unassigned      YES unset  down                  down
GigabitEthernet1/1/2   unassigned      YES unset  down                  down
GigabitEthernet1/1/3   unassigned      YES unset  down                  down
GigabitEthernet1/1/4   unassigned      YES unset  down                  down
L3_SW(config)#
```

📶 VLAN 라우팅 및 모니터링 구축 결과

🛠️ L3 스위치 설정

가상 인터페이스 생성
VLAN 10과 VLAN 20의 가상 인터페이스 (Vlan10, Vlan20)를 생성
IP 주소 할당
VLAN 10 192.168.20.1 IP주소 할당
VLAN 20 192.168.30.1 IP주소 할당
VLAN 간 라우팅 활성화

🔁 트래픽 모니터링 (Port Mirroring) 설정

Source (감시 대상):
GigabitEthernet0/24 포트의 트래픽을 감시
Destination (미러링 포트):
감시 대상 트래픽을 GigabitEthernet0/13 포트로 복사하여 전송

🔑 핵심 기능 요약 (SVI, 라우팅, SPAN)

SVI (Switch Virtual Interface):
스위치 가상 인터페이스를 통해 L3 기능을 제공
VLAN 라우팅: 서로 다른 VLAN 간의 통신을 가능
SPAN (Switch Port Analyzer):
트래픽 미러링 기능을 통해
네트워크 상태 및 보안 분석을 위한 기반을 마련

방화벽 라우팅 정책으로 네트워크 트래픽 최적화

인터페이스별 명확한 경로 지정으로 보안 강화 및 효율적 통신 관리

규칙 번호	목적지	게이트웨이	인터페이스
1	내부망 대역 (예: 192.168.0.0/24) 192.168.20.0/24 192.168.10.0/24 192.168.30.0/24	직접 연결 192.168.40.2	eth2
2	외부 인터넷 (0.0.0.0/0)	외부 게이트웨이 IP 10.10.70.1	eth1
3	기본 라우팅	기본 게이트웨이 10.10.70.1	eth1

내부망 간 통신은 **eth2** 인터페이스2를 통해 안전하게 분리 및 관리

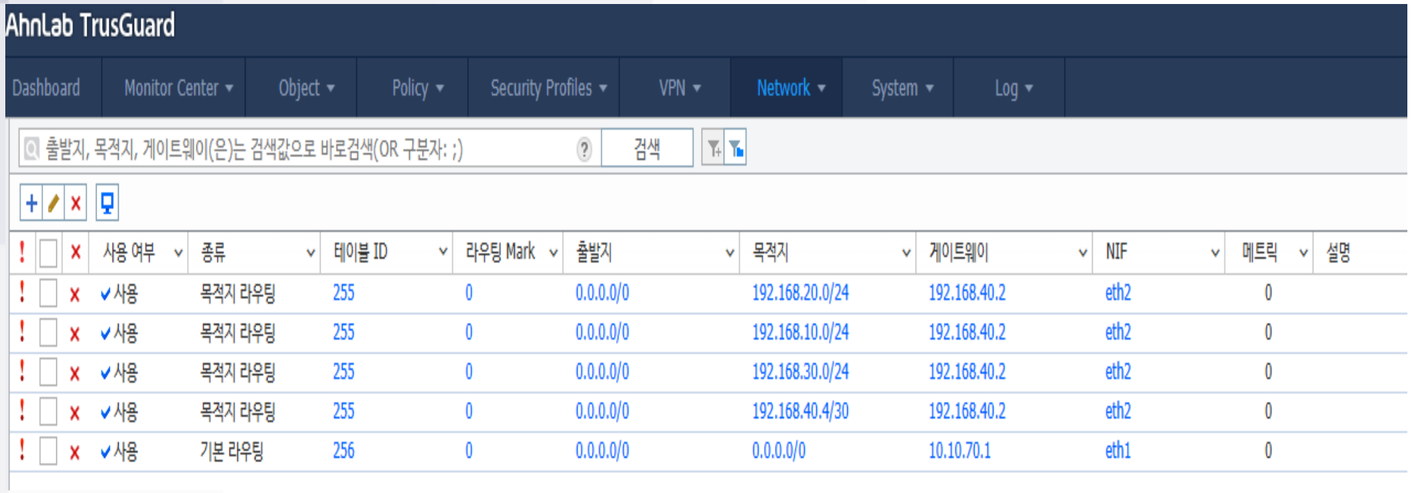
외부 인터넷 접근은 **eth1** 인터페이스1로 명확히 라우팅되어 보안 강화

목적지 기반 라우팅 정책으로 트래픽 경로를 효율적으로 통제

체계적 라우팅 관리로 네트워크 효율성 향상 및 보안 위협 최소화 실현

방화벽 라우팅 정책으로 네트워크 트래픽 최적화

인터페이스별 명확한 경로 지정으로 보안 강화 및 효율적 통신 관리



사용 여부	종류	테이블 ID	라우팅 Mark	출발지	목적지	게이트웨이	NIF	메트릭	설명
✔사용	목적지 라우팅	255	0	0.0.0.0/0	192.168.20.0/24	192.168.40.2	eth2	0	
✔사용	목적지 라우팅	255	0	0.0.0.0/0	192.168.10.0/24	192.168.40.2	eth2	0	
✔사용	목적지 라우팅	255	0	0.0.0.0/0	192.168.30.0/24	192.168.40.2	eth2	0	
✔사용	목적지 라우팅	255	0	0.0.0.0/0	192.168.40.4/30	192.168.40.2	eth2	0	
✔사용	기본 라우팅	256	0	0.0.0.0/0	0.0.0.0/0	10.10.70.1	eth1	0	

보안 사용자(Sec User) 생성 및 Root 권한 위임

SSH 접속 환경 구축 및 Sudo 권한 설정

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
sec_user        ALL=(ALL:ALL) ALL
```

🔒 **보안 사용자(Sec User) 생성 및 Root 권한 위임**

🛡️ **보안 운영 정책 수립**

일반 사용자 접근: 일반 사용자는 SSH 접속 후,
필요한 경우에만 sudo 명령어를 통해 일시적으로
Root 권한을 획득하여 사용하는 보안 운영 정책을 확립

리스크 최소화: Root 계정 직접 사용을 최소화하여
보안 리스크를 줄이고, 작업 이력 관리를 용이하게 하여
효율성을 높임

🔑 **핵심 키워드**

SSH 접속, 일반 사용자 계정, sudo 설정
관리 권한 위임, 보안 운영 정책

최소 권한 원칙 기반 Zone별 보안 운영 체계

INT, EXT, DMZ 3단계 권한 통제와 실시간 모니터링으로 보안 대응력 강화

Zone별 엄격한 권한 통제

INT, EXT, DMZ 3단계로 자산을 구분하여 모든 트래픽은 명시적 허용 시에만 통과하며 **최소 권한 원칙**을 준수

VLAN 및 NAT 서비스 관리

VLAN과 NAT 기반 서비스 포트를 정의하여 트래픽 분리 및 제어를 구현, 네트워크 보안의 체계적 관리 기반을 마련

실시간 모니터링 체계

24시간 실시간 모니터링으로 이상 트래픽 및 위협을 즉시 탐지, 보안 운영의 **가시성과 대응력**을 극대화

보안 관리 효율성 및 대응력 강화

일관된 보안 정책 적용과 실시간 감시로 내부 위협과 외부 공격에 효과적으로 대응하며 관리 효율성을 향상

방화벽 인터페이스 관리와 네트워크 영역의 전략적 정의

각 인터페이스의 IP·상태 관리로 보안 경계 강화 및 트래픽 최적화 실현

인터페이스	IP 주소	CIDR	영역	상태
eth0	10.0.0.254	24	NONE	활성
eth1	10.10.70.153	24	외부 (EXT)	활성
eth2	192.168.40.1	30	내부 (INT)	활성
eth3	172.16.1.1 172.16.1.2	24	DMZ Web svr(웹서버)	활성

물리적 인터페이스별 IP 할당과 링크 상태는 네트워크 영역 구분 및 트래픽 관리의 핵심 요소

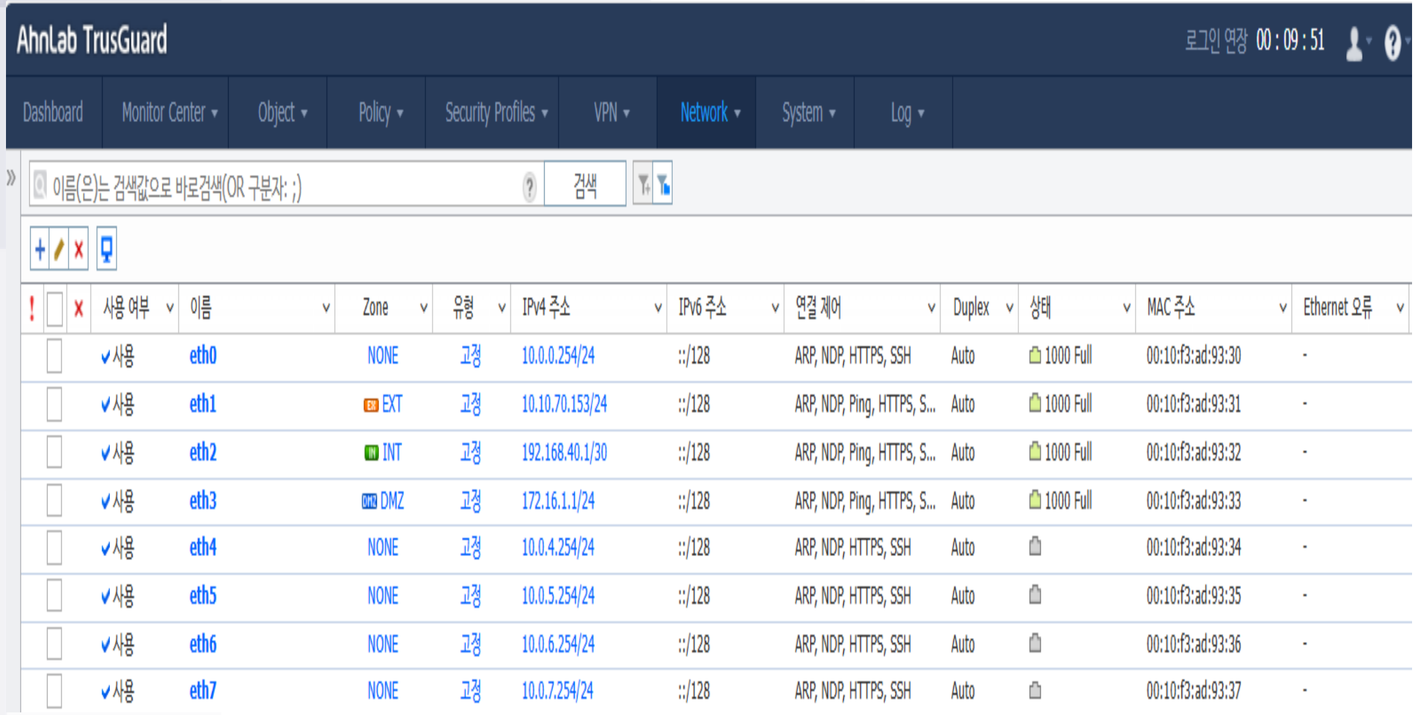
내부(INT), 외부(EXT), DMZ, NONE(기본네트워크) 등 각 영역은 보안 정책 적용에서 명확한 경계 역할

인터페이스 상태 모니터링은 장애조기 탐지 및 네트워크 안정성 확보에 필수적

관리자는 영역별 정책을 적용해 네트워크 보안성을 극대화하고 무단 접근을 차단

방화벽 인터페이스 관리와 네트워크 영역의 전략적 정의

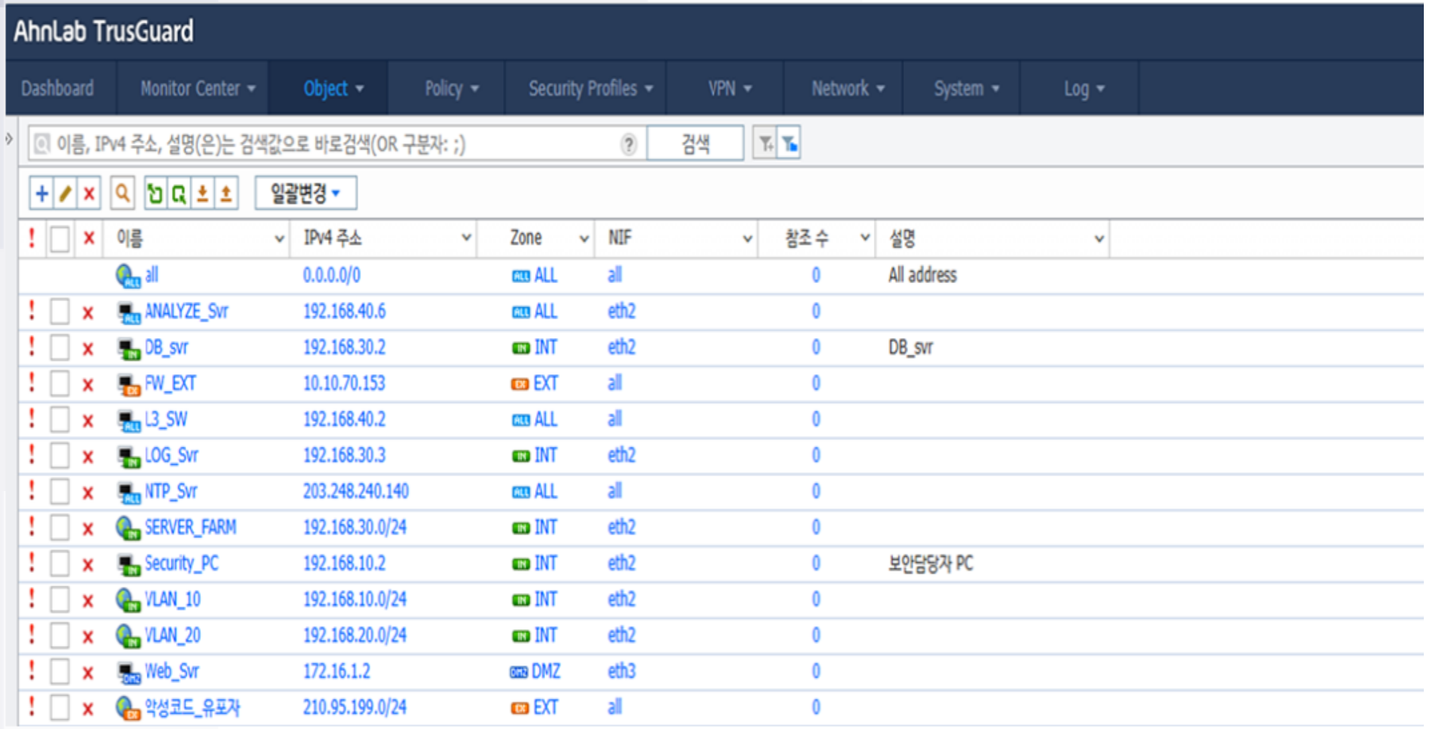
각 인터페이스의 IP·상태 관리로 보안 경계 강화 및 트래픽 최적화 실현



사용 여부	이름	Zone	유형	IPv4 주소	IPv6 주소	연결 제어	Duplex	상태	MAC 주소	Ethernet 오류
사용	eth0	NONE	고정	10.0.0.254/24	::/128	ARP, NDP, HTTPS, SSH	Auto	1000 Full	00:10:f3:ad:93:30	-
사용	eth1	EXT	고정	10.10.70.153/24	::/128	ARP, NDP, Ping, HTTPS, S...	Auto	1000 Full	00:10:f3:ad:93:31	-
사용	eth2	INT	고정	192.168.40.1/30	::/128	ARP, NDP, Ping, HTTPS, S...	Auto	1000 Full	00:10:f3:ad:93:32	-
사용	eth3	DMZ	고정	172.16.1.1/24	::/128	ARP, NDP, Ping, HTTPS, S...	Auto	1000 Full	00:10:f3:ad:93:33	-
사용	eth4	NONE	고정	10.0.4.254/24	::/128	ARP, NDP, HTTPS, SSH	Auto		00:10:f3:ad:93:34	-
사용	eth5	NONE	고정	10.0.5.254/24	::/128	ARP, NDP, HTTPS, SSH	Auto		00:10:f3:ad:93:35	-
사용	eth6	NONE	고정	10.0.6.254/24	::/128	ARP, NDP, HTTPS, SSH	Auto		00:10:f3:ad:93:36	-
사용	eth7	NONE	고정	10.0.7.254/24	::/128	ARP, NDP, HTTPS, SSH	Auto		00:10:f3:ad:93:37	-

IP 주소 객체 정의를 통한 접근 통제 강화

Zone 기반 통제를 위한 서버/VLAN 객체 명세화 현황



🏠 IPv4 주소 객체 목록

객체 기반 관리: 방화벽을 중심으로 내부망(INT), DMZ 외부망(EXT), 서버, VLAN 등 네트워크의 주요 영역을 객체로 명확히 정의

체계적 정책 적용: 정의된 객체들을 기반으로 접근 통제 정책을 체계적으로 적용하여 관리 효율성을 높임

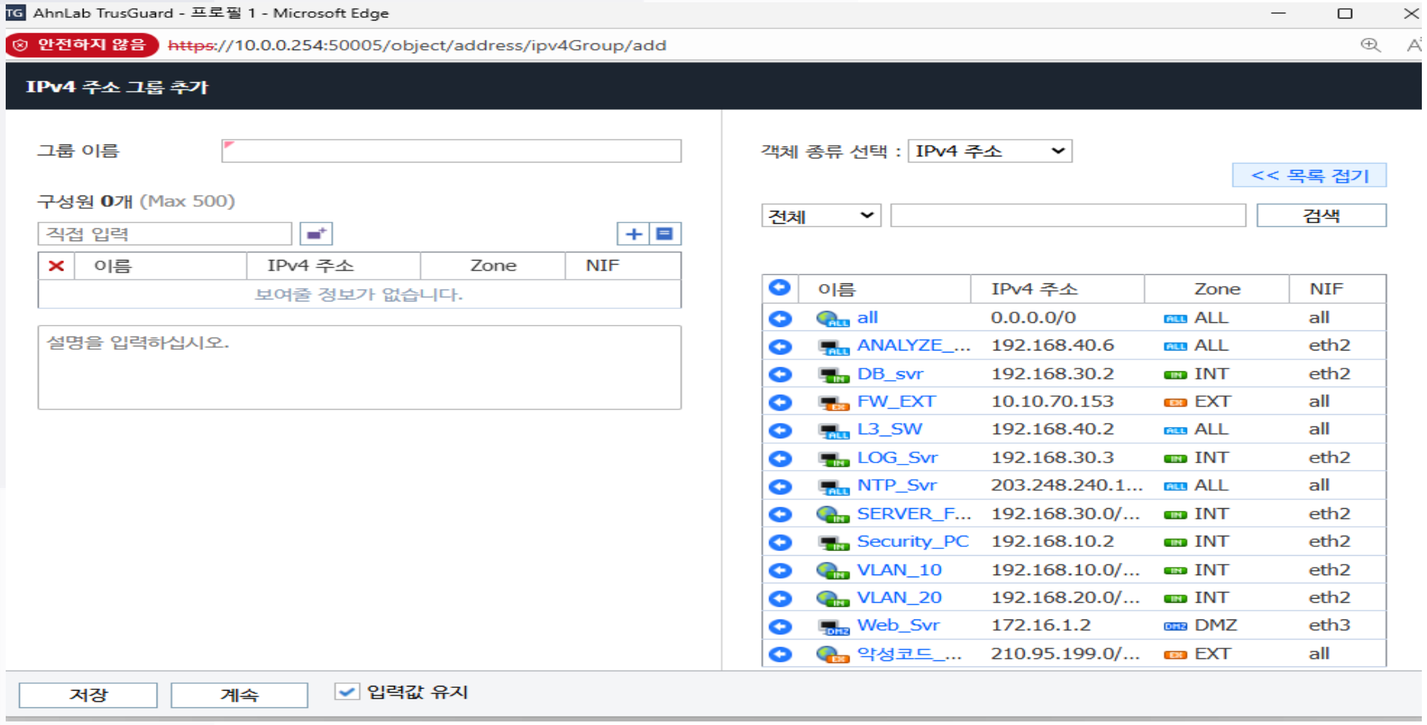
🌐 객체 관리의 목적

트러스가이드 오브젝트 설계: 오브젝트 설계 화면에서 주요 기기 및 영역을 명확히 분리하여 관리

보안 강화:
각기 다른 보안 정책을 통해 접근을 통제하고,
악성코드 유포지, 위험 요소 등 위협 대상을 별도로 관리하여네트워크 전체의 안전성을 높임

방화벽 네트워크 객체(Object) 정의 및 Zone 기반 영역

각 주소 그룹관리와 네트워크 영역 (DMZ INT EXT) 설정을 통한 보안 경계강화



보안 정책 강화 기반 구축
목적: 각 주소 객체 관리와 네트워크 영역
(DMZ, INT, EXT 등) 설정을 통합하여 보안 경계를 강화

IPv4 주소 그룹 추가 (그룹 지정)
오른쪽 목록에서 기존에 정의된 개별 IP 주소 객체들
(DB 서버, Web 서버, L3 스위치, VLAN 10/20 등)을 선택

선택한 객체들을 왼쪽 그룹 구성원 목록에 추가하여
논리적인 보안 그룹

핵심 사항
객체 재활용: 개별적으로 정의된 IPv4 주소 객체를
재활용하여 그룹 단위로 묶음으로써, 복잡한 정책을
단순화하고 관리 효율성을 높임

Zone 기반
객체들이 각각 Zone(INT, DMZ, EXT) 정보와 함께 관리되어,
그룹 정책 적용 시 네트워크 영역 기반의 통제가 용이

방화벽 보안 정책 설정 원칙 및 규칙

명확한 허용과 거부 규칙으로 네트워크 보호와 비즈니스 연속성 확보

01 방화벽 보안 정책 개요

- 내부와 외부 네트워크 트래픽을 허용과 거부 규칙으로 제어
- 정책은 네트워크 보안성을 향상시키며 비즈니스 연속성 유지에 필수적

02 보안 정책의 기본 원칙

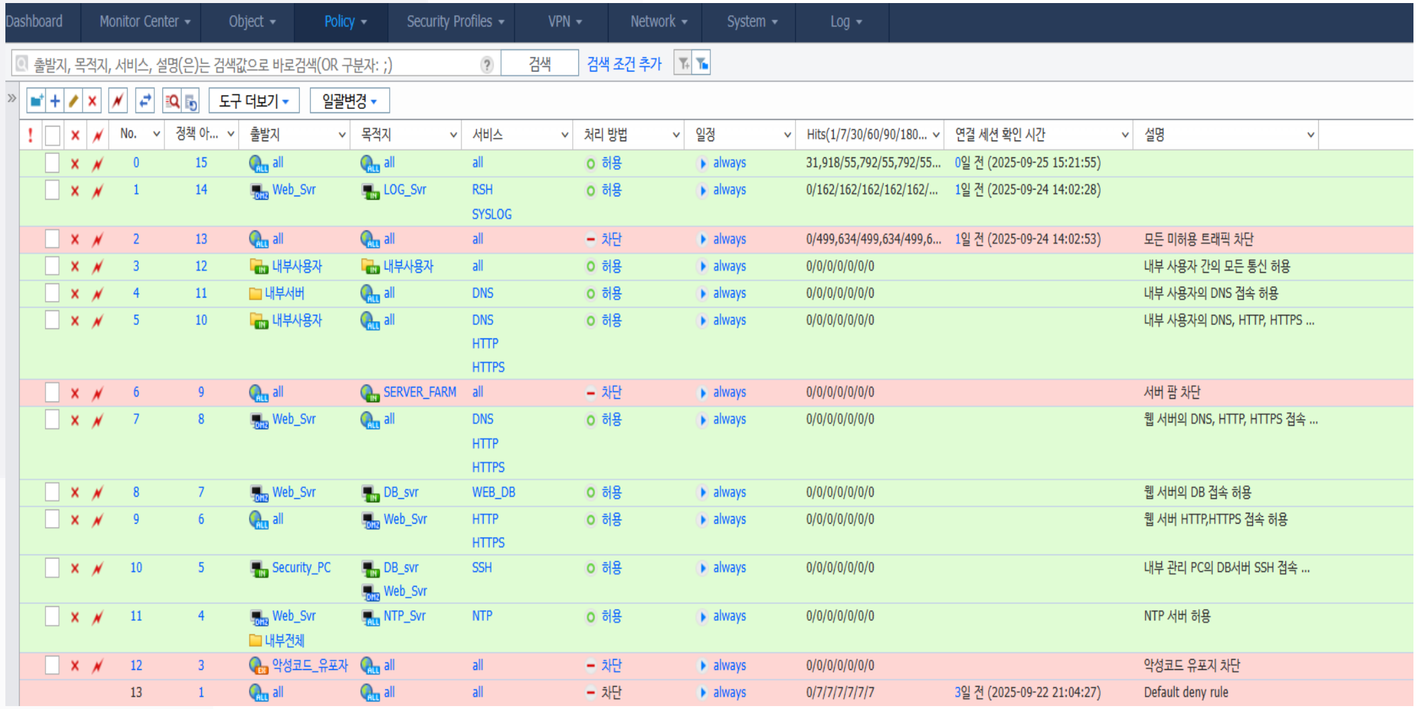
- **명시적 허용 기반**: 허용된 트래픽만 처리하고 나머지는 차단
- **최소 권한 원칙**: 필요한 서비스와 포트만 허용하여 위험 최소화
- **규칙 우선순위 설정**: 충돌 방지를 위해 규칙을 체계적으로 정렬

03 효과적인 정책의 역할

- 잠재적 위협을 차단하여 네트워크 안전성 확보
- 비즈니스 연속성을 보장하는 핵심 요소

내부망(INT) 통제 강화 및 최소화에 초점

필요 최소한의 서비스 (DNS, HTTP/S, DB등) 제외한 모든 트래픽 차단 정책 적용



No.	정책 아...	출발지	목적지	서비스	처리 방법	일정	Hits(1/7/30/60/90/180...	연결 세션 확인 시간	설명
0	15	all	all	all	허용	always	31,918/55,792/55,792/55...	0일 전 (2025-09-25 15:21:55)	
1	14	Web_Svr	LOG_Svr	RSH SYSLOG	허용	always	0/162/162/162/162/...	1일 전 (2025-09-24 14:02:28)	
2	13	all	all	all	차단	always	0/499,634/499,634/499,6...	1일 전 (2025-09-24 14:02:53)	모든 미허용 트래픽 차단
3	12	내부사용자	내부사용자	all	허용	always	0/0/0/0/0/0/0		내부 사용자 간의 모든 통신 허용
4	11	내부서버	all	DNS	허용	always	0/0/0/0/0/0/0		내부 사용자의 DNS 접속 허용
5	10	내부사용자	all	DNS HTTP HTTPS	허용	always	0/0/0/0/0/0/0		내부 사용자의 DNS, HTTP, HTTPS ...
6	9	all	SERVER_FARM	all	차단	always	0/0/0/0/0/0/0		서버 팜 차단
7	8	Web_Svr	all	DNS HTTP HTTPS	허용	always	0/0/0/0/0/0/0		웹 서버의 DNS, HTTP, HTTPS 접속 ...
8	7	Web_Svr	DB_svr	WEB_DB	허용	always	0/0/0/0/0/0/0		웹 서버의 DB 접속 허용
9	6	all	Web_Svr	HTTP HTTPS	허용	always	0/0/0/0/0/0/0		웹 서버 HTTP,HTTPS 접속 허용
10	5	Security_PC	DB_svr Web_Svr	SSH	허용	always	0/0/0/0/0/0/0		내부 관리 PC의 DB서버 SSH 접속 ...
11	4	Web_Svr 내부전체	NTP_Svr	NTP	허용	always	0/0/0/0/0/0/0		NTP 서버 허용
12	3	악성코드_유포자	all	all	차단	always	0/0/0/0/0/0/0		악성코드 유포지 차단
13	1	all	all	all	차단	always	0/7/7/7/7/7/7	3일 전 (2025-09-22 21:04:27)	Default deny rule

내부망 (INT 정책)

🎯 내부망 통제 강화 및 최소화에 초점
- 필요한 최소한 서비스 (DNS, HTTP/S, DB등)허용하고 모든 트래픽을 차단하는 정책

📝 Zone 기반 허용 및 차단현황
-총 13개의 방화벽 정책이 정의되어 있으며 대부분 차단

💻 **Web server 관련 (필수 서비스 허용)**
-DNS/HTTP/HTTPS (NO.7) 및 DB접속정책(NO.8)
필수 서비스로 허용

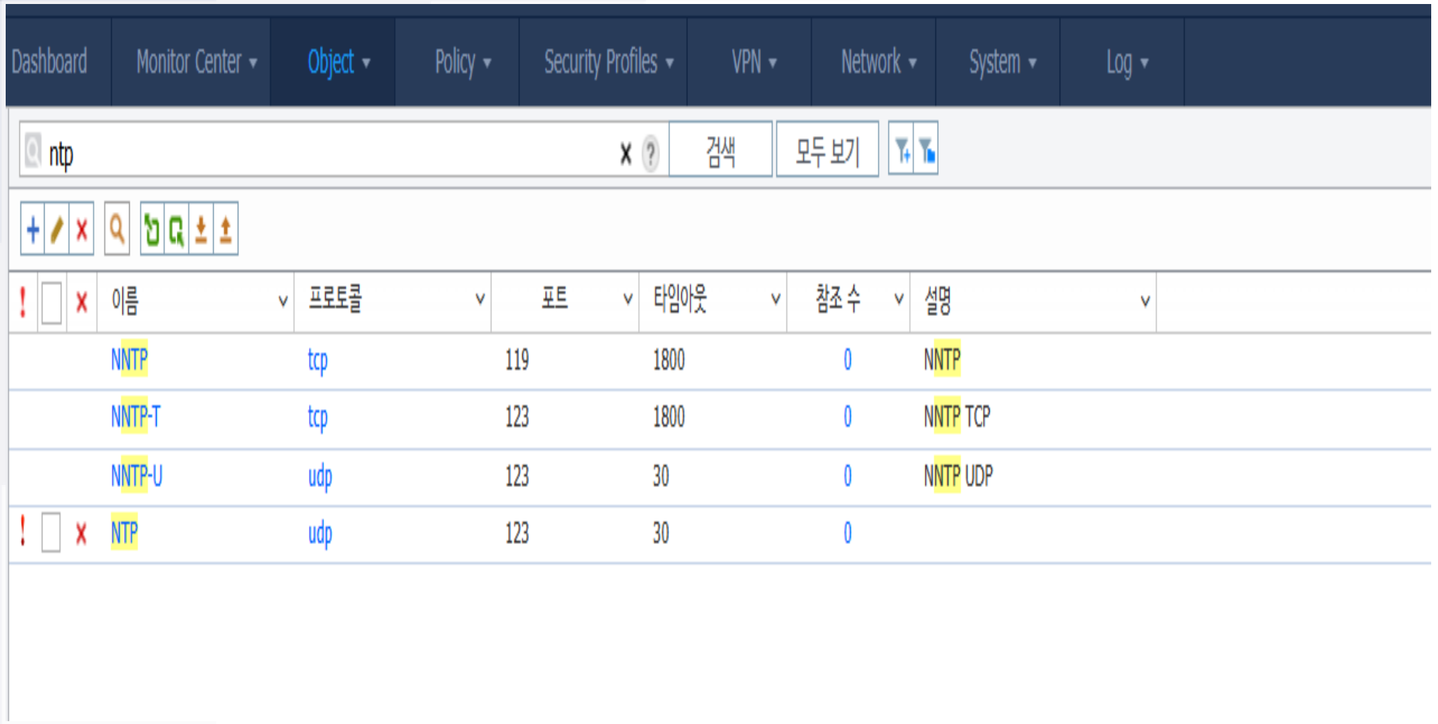
👥 **내부사용자관련 (허용정책)**
-DNS/HTTP등을 허용하는 정책이 존재 (NO.4,5)

🚫 **차단정책**
트래픽 차단 : 불필요 트래픽은 차단
SERVER_FARM 대상을 차단하는 정책(No.6)
악성코드유포지 대상을 차단하는정책(NO.12)
기본정책인 Default dent rule(N0.13)으로 모든 트래픽을 최종적으로 차단

정책마다 출발지 목적지 서비스 허용/차단 여부가 명확하게 정의

네트워크 보안 Zone 기반 필수 서비스 정책 구축

시간 동기화(NTP)를 통한 방화벽 Zone 제어 및 정책 설정



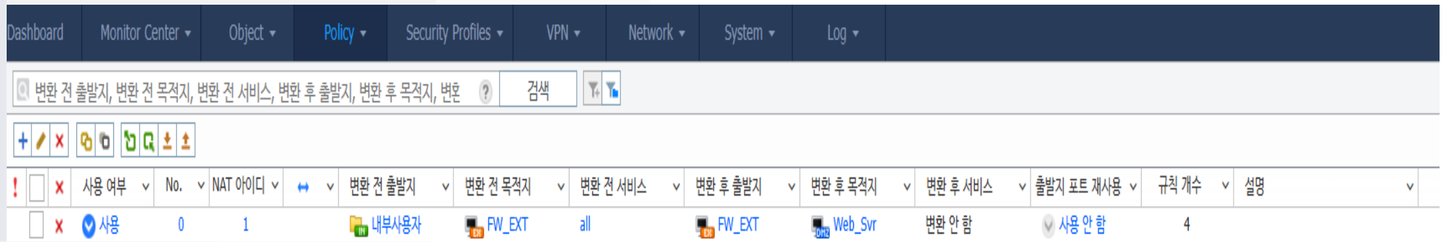
🔑 주요 포트 설명 (NTP)

UDP 123번
서비스: NTP (Network Time Protocol)
기능: 인터넷의 시간 서버와 동기화하여 모든 컴퓨터 및 네트워크 장치의 시스템 시간을 정확하게 맞춰주는 규칙
로그 분석, 보안 이벤트 시간 확인 등 네트워크 운영의 정확성을 위해 필수적인 서비스

TCP 119번
서비스: NNTP (Network News Transfer Protocol)
기능: 게시판 서버와 클라이언트 간의 게시물 사용

NAT 정책구현

내부 사용자 인터넷／외부 서비스 접근 및 서버 서비스 노출을 위한 IP 변환 규칙



🌐 NAT 정책 구현

목적: 내부망 사용자/서버의 IP를 외부망 접근을 위해 변환하거나, 외부 사용자의 내부 서버 접근을 위해 IP를 변환하는 규칙을 설정

규칙 ID : 1번부터 시작
NAT 종류: Dynamic NAT (유동 NAT)과 Static NAT (고정 NAT)이 혼합되어 사용
외부 인터페이스: FW_EXT (방화벽 외부망 인터페이스)를 사용하여 IP 변환이 이루어짐
대상 서비스: Web_Svr (웹 서버) 등 특정 서버로의 접근을 위해 설정

🔄 NAT 정책 구현

규칙 0, 1, 2 (Dynamic NAT)
유형: 유동 NAT (Dynamic NAT)
기능: 내부사용자(0,1,2그룹) 외부 인터넷에 접속할때 IP FW_EXT의 공인 IP로 변환하여 내부망 정보 숨김
규칙 3,4 (Static NAT)
유형: 고정 NAT (Static NAT)
기능 : 외부 사용자가 특정포트 (TCP 80/443)로 접속할때 목적지 IP를 내부 웹서버 (10.10.75.153) 1:1로 변환
목적: 웹서비스 접근성을 확보하면서 특정서버에 대해서만 통제를 집중하여 보안을 강화

두 이미지 IP변환이라는 동일한 목표 아래 유동 NAT와 고정 NAT가 어떻게 구현되었는지 목록과 세부사항을 나누어 보여줌

NAT 정책: 동적 NAT와 정적 NAT 비교

내부 IP주소 변환 방식과 용도별 구현 차이 이해

동적 NAT

- 내부 서버의 소스 IP를 외부 IP로 자동 변환
- 내부 여러 IP를 제한된 수의 공인 IP로 공유 가능
- 외부에서는 내부 서버 직접 접근 불가
- 내부 사용자 중심의 외부 인터넷 접속에 적합

정적 NAT

- 내부 특정 IP를 외부 고정 IP에 1:1 매핑
- 외부에서 내부 서버 서비스 접근 가능
- 포트 포워딩으로 서비스 노출 제어
- 내부 서버의 외부 서비스 제공에 최적화

NAT Loopback으로 내부/외부 동일 URL 접근 기술

내부 사용자가 외부 IP로도 내부 서버에 접근 가능해져 네트워크 운영 효율성과 사용자 편의 극대화

01 Bridge

01 NAT Loopback은 내부사용자의 외부 IP 접근 요청을 방화벽 내부에서 재처리 하여 외부로 나가지 않고 내부서버로 연결 경로 전환(Loopback)하는 기능.

02 Puzzle

01 내부외부 사용자의 접근 경로를 하나로 맞추는 NAT Loopback은 복잡한 네트워크 환경에서의 문제 해결과 관리 단순화를 가능하게 하는 **퍼즐 조각 맞추기**와 같습니다.

MariaDB 데이터베이스 운영 및 관리 핵심

MySQL 호환 명령어로 데이터 구조 확인과 안정적 운영 보장

웹 서버와 DB 서버 간 안전한 연동 및 접근 통제

3306포트에 대한 엄격한 접근 제한으로 데이터베이스 보안을 강화

웹 서버는 MariaDB 데이터베이스 서버와 연동되어 안정적인 데이터 관리 환경을 제공

3306 포트 접근은 웹 서버 및 애플리케이션 서버에 한해 엄격히 허용하는 정책 운영

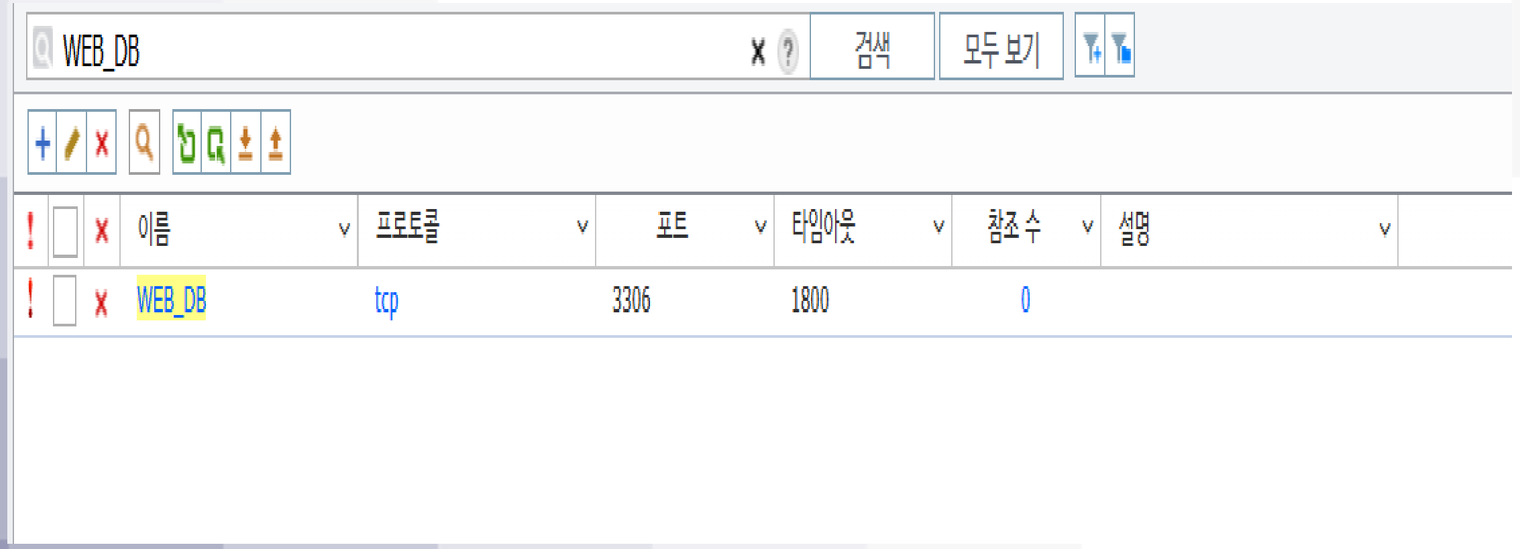
무단 접근 차단으로 **데이터베이스 보안**을 강화하고 데이터 유출 위험 최소화

접근 통제는 **최소 권한 원칙**에 기반해 불필요한 권한 부여를 방지하고 시스템 안정성 유지

이 정책은 전체 시스템의 보안 강화와 신뢰성 확보에 핵심적 역할 수행

웹 서버와 DB 서버 간 안전한 연동을 위한 접근 통제

DB포스 (TCP3306)에 대한 최소 권한 접근 제어 적용



🔒 정책 목적 및 원칙

목적
웹 서버와 DB 서버 간 연동 트래픽을 식별하고 제어하여 안전한 통신 환경을 구축
원칙: DB 포트 (TCP 3306)에 대해 최소 권한 접근 제어를 적용

⚙ 적용 대상 서비스

데이터베이스 서비스: MySQL 포트인 TCP 3306을 대상

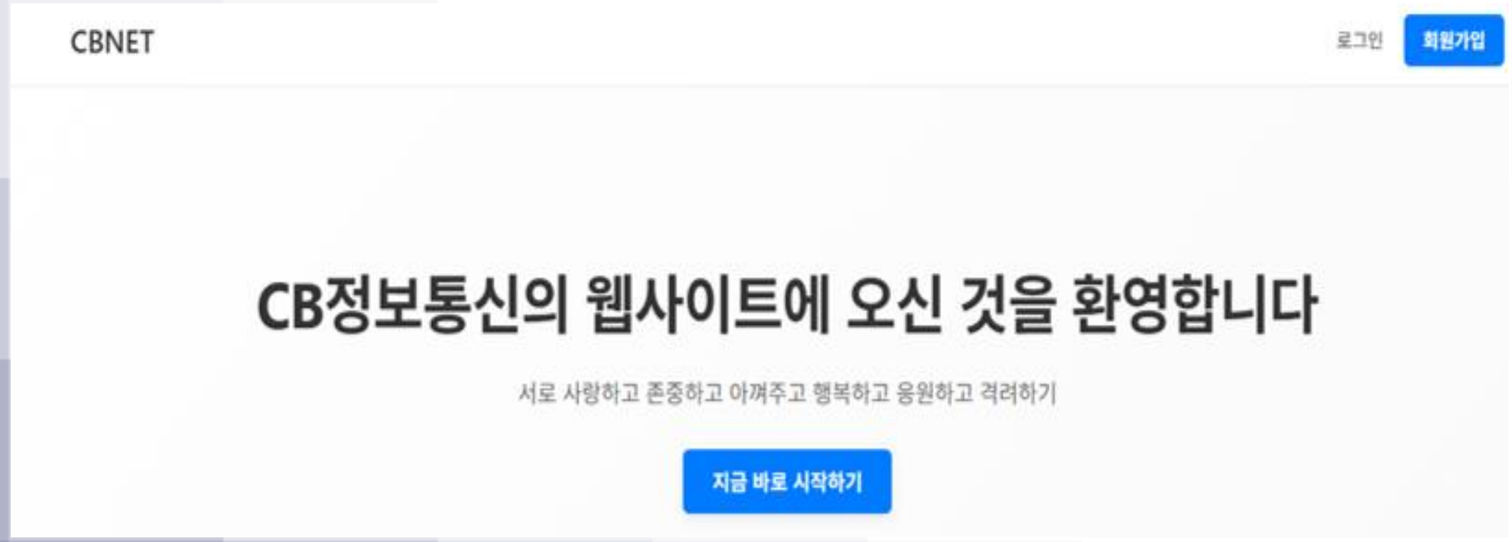
🔥 방화벽 정책의 활용

최소 권한 원칙: WEB_DB라는 이름의 서비스 객체를 정의하여, TCP 3306 포트를 사용하는 트래픽에 대해서만 접근을 허용

세분화된 허용: 이 서비스 객체를 방화벽 규칙에 적용할 때, 특정 웹 서버 IP 주소(출발지)와 특정 DB 서버 IP 주소(목적지)만을 지정하여 세부적으로 허용함으로써 보안성을 극대화

웹 서버와 DB 서버 간 안전한 연동 및 접근 통제

정책 적용 후 웹사이트 접속테스트 및 서비스 정상 운영확인



정책 적용

웹사이트 접속 테스트

웹 서버와 DB 서버 간 안전한 연동 및 접근 통제

DB 접근 통제 정책 확인 및 데이터 베이스 테이블정보 조회

```
Database changed
MariaDB [cb_db]> show tables;
+------------------+
| Tables_in_cb_db |
+------------------+
| comments         |
| posts            |
| users            |
+------------------+
3 rows in set (0.000 sec)

MariaDB [cb_db]> select * from users;
+----+-----------+----------------------+--------------------------------------------------------------+----
----------------+
| id | username | email                | password                                                     | cre
ated_at           |
+----+-----------+----------------------+--------------------------------------------------------------+----
----------------+
|  1 | rlagoddn | rlatkd1089@naver.com | $2y$10$WKdxMaNlmTNcbdeBSHvHHuBcx/gnBuJr.4avYvUMPJuUTECmHdcA. | 202
5-09-25 10:50:55 |
+----+-----------+----------------------+--------------------------------------------------------------+----
----------------+
1 row in set (0.001 sec)

MariaDB [cb_db]> select * from posts;
+----+---------+----------+-----------------+-----------------------------------------+----------+---------
-------------+
| id | user_id | username | title           | content                                 | filepath | created
_at           |
+----+---------+----------+-----------------+-----------------------------------------+----------+---------
-------------+
|  1 |       1 | rlagoddn | 드디어 됐다     | ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ              | NULL     | 2025-09
-25 11:03:36 |
+----+---------+----------+-----------------+-----------------------------------------+----------+---------
-------------+
1 row in set (0.000 sec)

MariaDB [cb_db]>
```

💾 **웹DB 연동 후** 데이터베이스 접근 및 조회 확인

MariaDB 데이터베이스에 접속하여
테이블 정보를 조회하고 데이터를 추출한 콘솔화면

웹 애플리케이션 보안 코드 분석: 기본 설정 및 인증 강화

Apache 설정부터 PHP 인증까지, 보안 취약점 최소화를 위한 핵심 전략

01 **Apache 가상 호스트 설정**으로 Documentroot 경로를 명확히 지정하여 외부 접근 제어 강화

02 업로드 디렉토리 권한 관리로 **불필요한 접근 통제** 및 악성 코드 실행 방지

03 **환경 변수나 암호화된 구성 파일을** 통해 안전하게 관리한다. 장기적으로는 **키 관리 시스템**(Secure Vault) 도입을 검토하여 보안을 극대화

04 **login_process.php**에서 SQL Injection 방지를 위한 **준비된 구문(Prepared Statements)** 적용

05 비밀번호 검증 시 안전한 비교 알고리즘을 활용하여 해시 된 비밀번호를 검증

06 이러한 보안 조치는 개발 단계에서 필수적으로 적용해야 하는 **모범 사례**임

웹 애플리케이션 보안 코드 분석: 기본 설정 및 인증 강화

Apache 설정부터 PHP 인증까지, 보안 취약점 최소화를 위한 핵심 전략

root@web: /home/web/Desktop

```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port th
at
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/cbnet

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
root@web:/home/web/Desktop# ls -l /var/www/html/cbnet
total 72
-rw-r--r-- 1 root     root     5738 10월  13 20:09 board.php
-rw-r--r-- 1 root     root      936 10월  13 20:24 comment_process.php
-rw-r--r-- 1 root     root      277 10월  13 19:52 db.php
-rw-r--r-- 1 root     root     1361 10월  13 20:19 delete_process.php
-rw-r--r-- 1 root     root     3577 10월  13 20:17 edit.php
-rw-r--r-- 1 root     root     1995 10월  13 20:18 edit_process.php
-rw-r--r-- 1 root     root     2404 10월  13 20:06 index.php
-rw-r--r-- 1 root     root      960 10월  13 20:02 login.php
-rw-r--r-- 1 root     root      949 10월  13 20:05 login_process.php
-rw-r--r-- 1 root     root      105 10월  13 20:07 logout.php
-rw-r--r-- 1 root     root      817 10월  13 19:54 register.php
-rw-r--r-- 1 root     root     1249 10월  13 19:59 register_process.php
drwxr-xr-x 2 www-data www-data 4096 10월  13 19:43 uploads
-rw-r--r-- 1 root     root     5248 10월  13 20:15 view.php
-rw-r--r-- 1 root     root     2108 10월  13 20:10 write.php
-rw-r--r-- 1 root     root     1470 10월  13 20:13 write_process.php
```

root@web: /home/web/Desktop

```
<?php

$db_host = "192.168.30.2";
$db_name = "cb_db";
$db_user = "web_user";
$db_pass = "Ycdc";

try {
    $pdo = new PDO("mysql:host={$db_host};dbname={$db_name};charset=utf8", $db_u
ser, $db_pass);
} catch (PDOException $e) {
    die("DB 연결에 실패했습니다.");
}
?>
~
~
~
~
~
~
~
~
                                                              13,2          All
```

root@web: /home/web/Desktop

```
<?php
session_start();
require_once 'db.php';

if (isset($_POST['username'], $_POST['password'])) {
    $username = trim($_POST['username']);
    $password = trim($_POST['password']);

    $sql = "SELECT id, username, password FROM users WHERE username = :username"
;
    $stmt = $pdo->prepare($sql);
    $stmt->execute([':username' => $username]);

    $user = $stmt->fetch();

    if ($user && password_verify($password, $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['username'] = $user['username'];
        header("Location: index.php");
        exit();
    } else {
        // 401 Unauthorized 헤더 추가
        header('HTTP/1.0 401 Unauthorized');
                                                              1,2           Top
```

안전한 회원가입과 비밀번호 관리 핵심 프로세스

안전한 비밀번호 저장 및 검증을 통한 인증 보안 강화절차

01

회원가입 비밀번호 암호화

사용자가 입력한 비밀번호를 안전한 해시기법으로 암호화하여 저장

02

기존 비밀번호 검증

비밀번호 변경 시 입력한 기존 비밀번호와 저장 된 암호화 값을 검증하여 본인 확인을 수행

03

새 비밀번호 암호화 저장

검증 후 새로운 비밀번호를 다시 해시 기법으로 암호화하여 안전하게 저장

04

평문 저장 방지 및 보안 준수

모든 비밀번호는 평문 저장을 금지하며, 웹 보안 표준을 철저히 준수합니다.

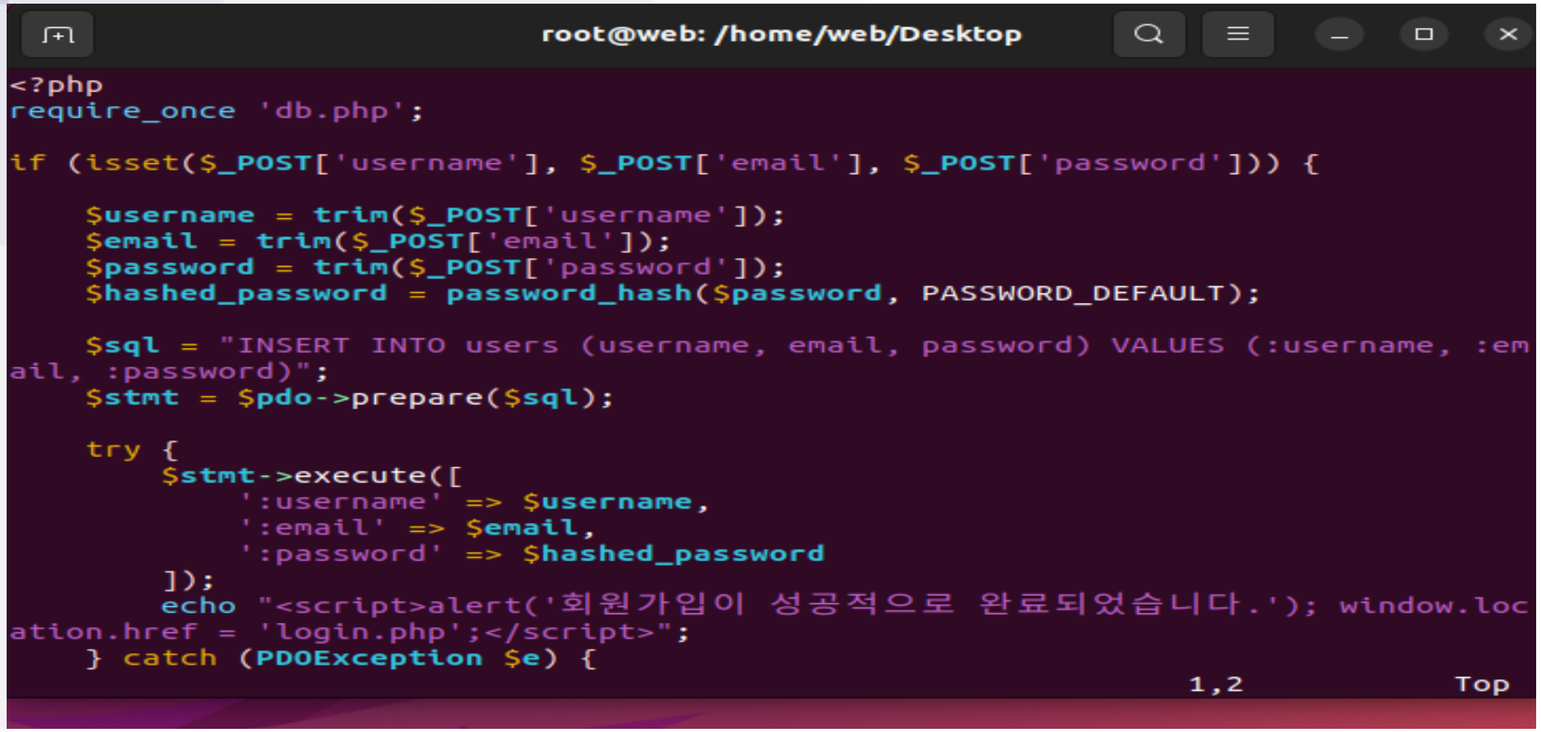
05

사용자 신뢰 및 데이터 보호 강화

안전한 암호화와 검증 기법을 통해 사용자 신뢰를 확보하고 데이터 유출 위험을 최소화합니다.

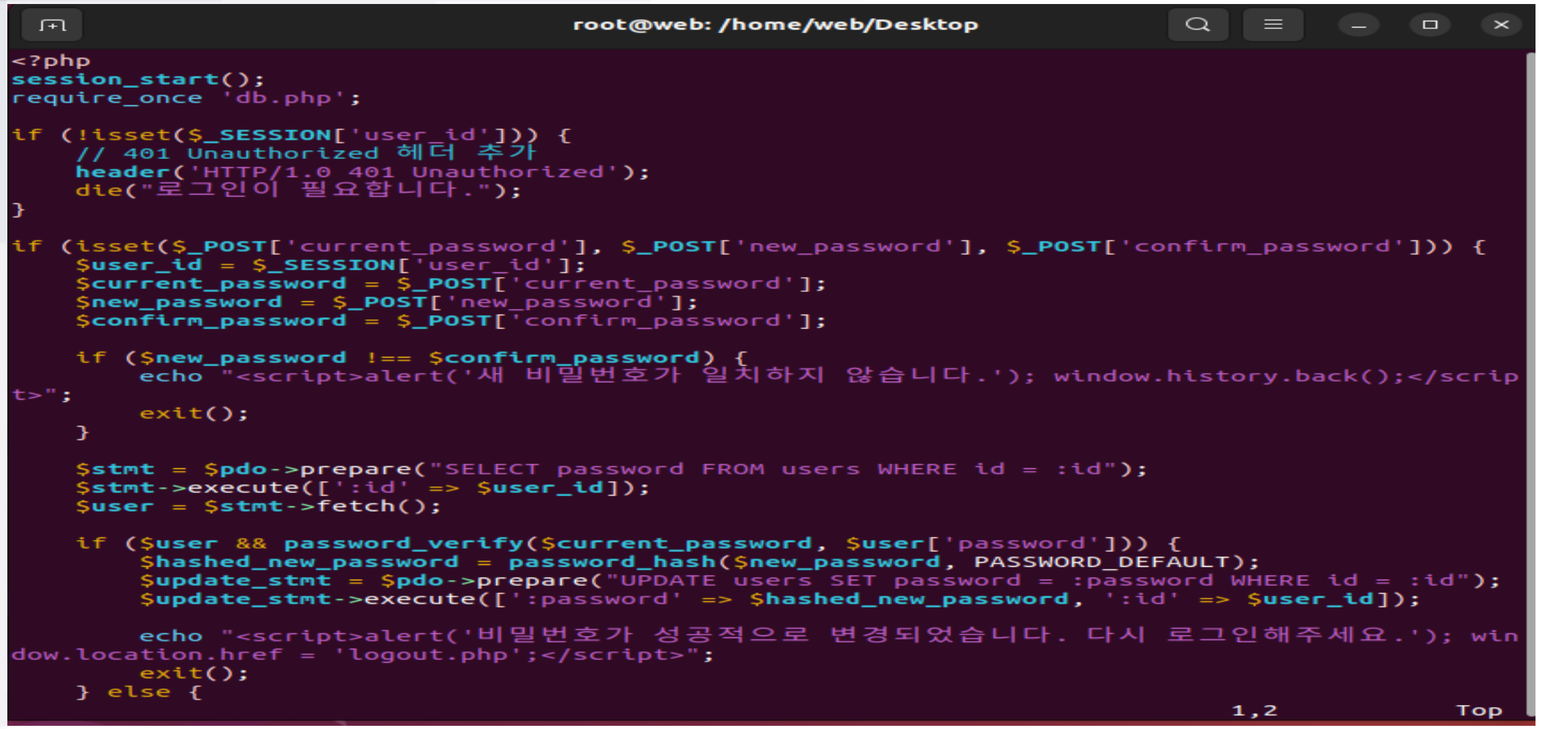
안전한 회원가입과 비밀번호 관리 핵심 프로세스

비밀번호 암호화 저장부터 안전한 인증 비교까지의 보안 표준 준수



안전한 회원가입과 비밀번호 관리 핵심 프로세스

비밀번호 암호화 저장부터 안전한 인증 비교까지의 보안 표준 준수

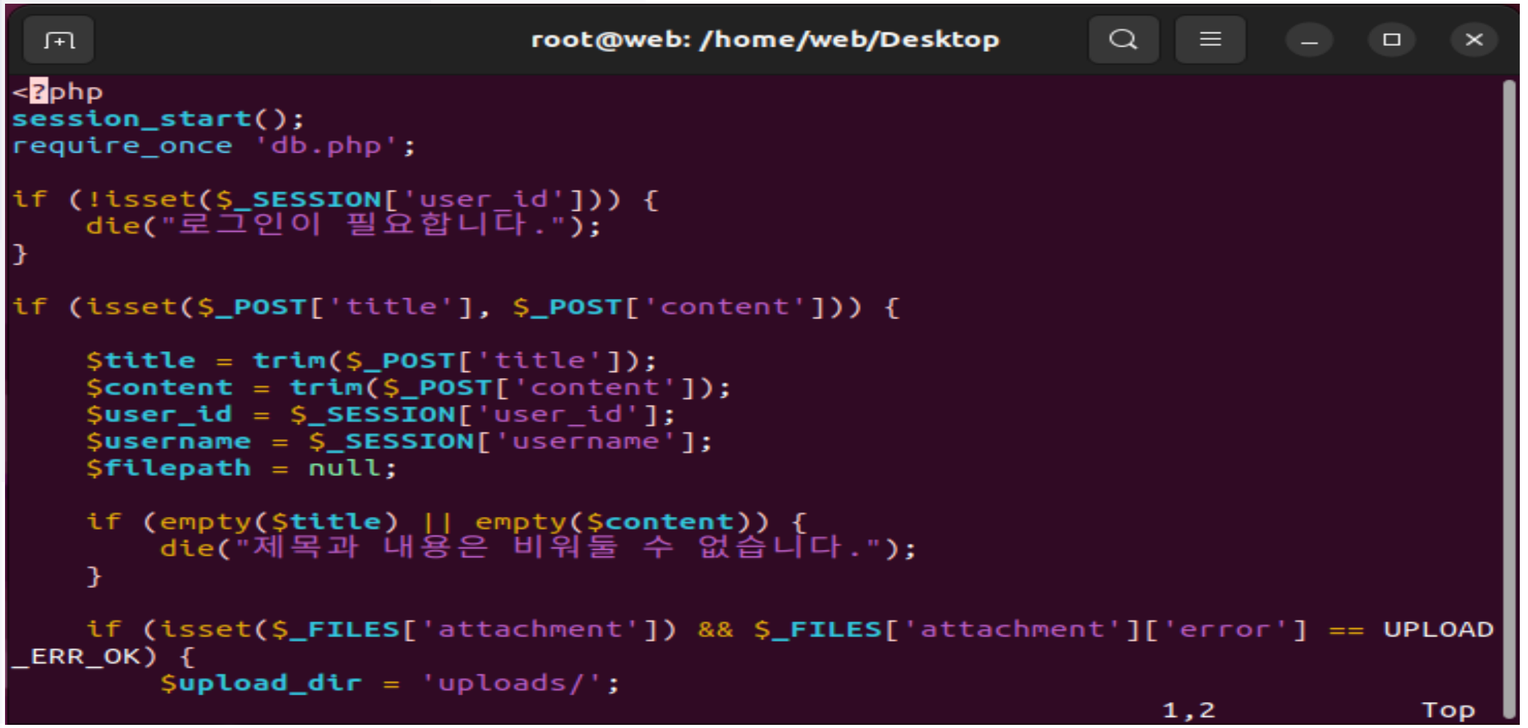


게시글 작성 및 관리 과정의 보안 절차

사용자 인증부터 데이터 삭제까지 안전한 보안 시스템 구축 방안

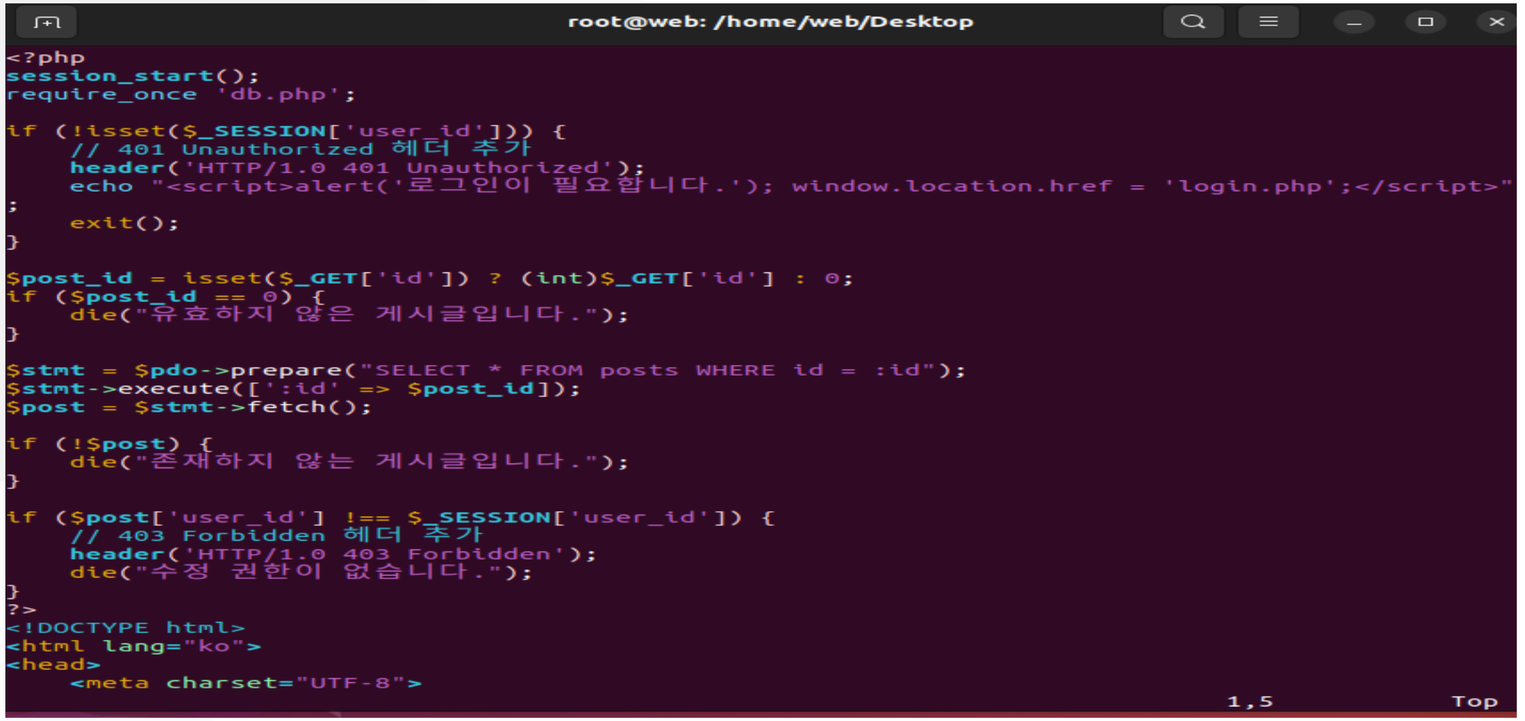
게시글 작성 및 관리 과정의 보안 절차

사용자 인가부터 첨부 파일 완전 삭제까지 보안을 위한 핵심 관리 방안



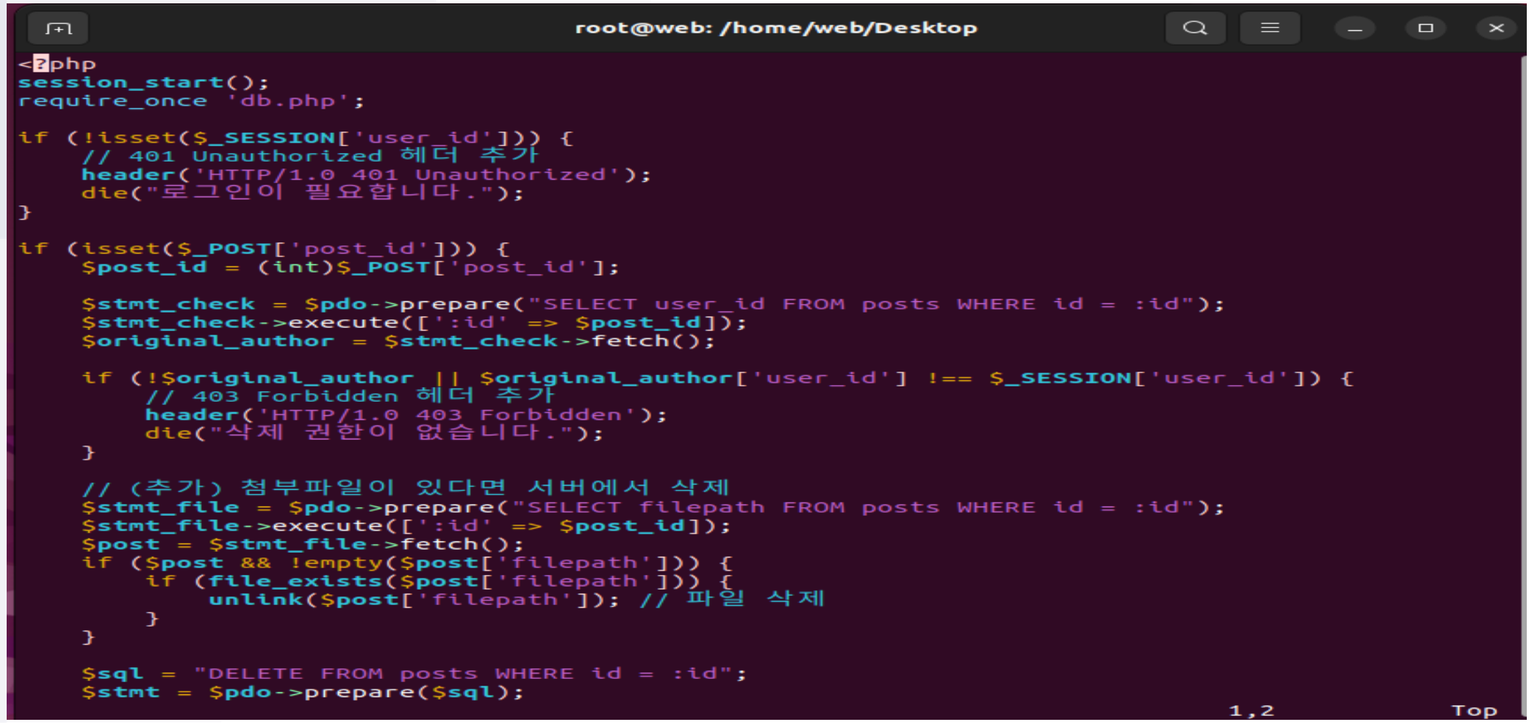
게시글 작성 및 관리 과정의 보안 절차

사용자 인가부터 첨부 파일 완전 삭제까지 보안을 위한 핵심 관리 방안



게시글 작성 및 관리 과정의 보안 절차

사용자 인가부터 첨부 파일 완전 삭제까지 보안을 위한 핵심 관리 방안



Syslog 데이터분석

네트워크 활동 및 사용자 세션중심의 보안 이벤트 분석

01 DNS 관련 경고 메시지가 감지되어 시스템의 네트워크 설정이나 서비스 상태를 점검

02 CRON 작업이 정상적으로 실행되고 root 사용자 세션이 종료되는 것을 확인

03 NAT 기능이 정상적으로 작동하고 내부 IP주소와 외부 IP 서버 등 DNS, HTTPS, HTTP 통신 시도

시스템 로그 (Syslog) 기반 보안 이벤트 분석

네트워크 트래픽, 시스템 활동, 인증 오류 기록 확인 및 통제

```
==> /var/log/syslog/192.168.40.1/2025-09-25_192.168.40.1.log <==
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`17`250924140250293`192.168.10.3`52379`8.8.8.8`53`eth2`eth1`SNAT`10.10.70.153`19583`````````INT`EXT`15`IB_NAT:3`4`````0`1058989`0````0`0`0``domain``US```202
50925`15:12:06```0`0`0`0```````
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`17`250924140250293`192.168.10.3`60043`8.8.8.8`53`eth2`eth1`SNAT`10.10.70.153`19584`````````INT`EXT`15`IB_NAT:3`4`````0`1058990`0````0`0`0``domain``US```202
50925`15:12:06```0`0`0`0```````
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`17`250924140250293`192.168.10.3`58931`8.8.8.8`443`eth2`eth1`SNAT`10.10.70.153`19585`````````INT`EXT`15`IB_NAT:3`4`````0`1058991`0````0`0`0``https``US```202
50925`15:12:06```0`0`0`0```````
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`17`250924140250293`192.168.10.3`53862`8.8.8.8`443`eth2`eth1`SNAT`10.10.70.153`19586`````````INT`EXT`15`IB_NAT:3`4`````0`1058992`0````0`0`0``https``US```202
50925`15:12:06```0`0`0`0```````
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`6`250924140250293`192.168.10.3`60561`23.46.155.140`443`eth2`eth1`SNAT`10.10.70.153`18113````````S`INT`EXT`15`IB_NAT:3`4`````0`1058993`0````0`0`0``https``US
```20250925`15:12:06```0`0`0`0```````
Sep 25 15:12:02 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:06`1`17`250924140250293`192.168.10.3`53907`142.250.76.234`443`eth2`eth1`SNAT`10.10.70.153`19587`````````INT`EXT`15`IB_NAT:3`4`````0`1058994`0````0`0`0``https``U
S```20250925`15:12:06```0`0`0`0```````
Sep 25 15:12:03 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:07`1`17`250924140250293`192.168.30.4`59317`8.8.8.8`53`eth2`eth1`SNAT`10.10.70.153`19588`````````INT`EXT`15`IB_NAT:6`4`````0`1058996`0````0`0`0``domain``US```202
50925`15:12:07```0`0`0`0```````
Sep 25 15:12:03 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:08`1`6`250924140250293`192.168.30.4`62334`74.178.76.44`443`eth2`eth1`SNAT`10.10.70.153`18114````````S`INT`EXT`15`IB_NAT:6`4`````0`1058997`0````0`0`0``https``US
```20250925`15:12:08```0`0`0`0```````
Sep 25 15:12:04 192.168.40.1 3`0`6`1`ad9330`1020`20250925`15: 12:09`1`6`250924140250293`192.168.30.4`62335`150.171.22.17`443`eth2`eth1`SNAT`10.10.70.153`18115````````S`INT`EXT`15`IB_NAT:6`4`````0`1058999`0````0`0`0``https``US
```20250925`15:12:09```0`0`0`0```````
```

```
ubuntu@ubuntu:/home$ tail -f /var/log/syslog/172.16.1.2/2025-09-25_172.16.1.2.log /var/log/syslog/192.168.30.1/2025-09-25_192.168.30.1.log /var/log/syslog/192.168.40.1/2025-09-25_192.168.40.1.log
==> /var/log/syslog/172.16.1.2/2025-09-25_172.16.1.2.log <==
Sep 25 15:03:15 ubuntu-virtual-machine kernel: [73037.825833] audit: type=1107 audit(1758780195.565:163): pid=705 uid=102 auid=4294967295 ses=4294967295 subj=unconfined msg='apparmor="DENIED" operation="dbus_signal"  bus="syst
em" path="/org/freedesktop/login1" interface="org.freedesktop.DBus.Properties" member="PropertiesChanged" name=":1.15" mask="receive" pid=8002 label="snap.firefox.firefox" peer_pid=745 peer_label="unconfined"
Sep 25 15:03:15 ubuntu-virtual-machine kernel: [73037.825833]  exe="/usr/bin/dbus-daemon" sauid=102 hostname=? addr=? terminal=?'
Sep 25 15:03:15 ubuntu-virtual-machine update-notifier[2265]: gtk_widget_get_scale_factor: assertion 'GTK_IS_WIDGET (widget)' failed
Sep 25 15:03:15 ubuntu-virtual-machine update-notifier[2265]: gtk_widget_get_scale_factor: assertion 'GTK_IS_WIDGET (widget)' failed
Sep 25 15:09:01 ubuntu-virtual-machine CRON[12729]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
Sep 25 15:09:01 ubuntu-virtual-machine CRON[12730]: (root) CMD (   [ -x /usr/lib/php/sessionclean ] && if [ ! -d /run/systemd/system ]; then /usr/lib/php/sessionclean; fi)
Sep 25 15:09:01 ubuntu-virtual-machine CRON[12729]: pam_unix(cron:session): session closed for user root
Sep 25 15:09:06 ubuntu-virtual-machine systemd[1]: Starting Clean php session files...
Sep 25 15:09:06 ubuntu-virtual-machine systemd[1]: phpsessionclean.service: Deactivated successfully.
Sep 25 15:09:06 ubuntu-virtual-machine systemd[1]: Finished Clean php session files.
```

```
==> /var/log/syslog/192.168.30.1/2025-09-25_192.168.30.1.log <==
Sep 25 10:29:49 _gateway 92: *Sep 25 10:29:48: %SYS-3-USERLOG_ERR: Message from tty0(user id: ): test
Sep 25 10:29:49 _gateway 93: *Sep 25 10:29:49: %SYS-3-USERLOG_ERR: Message from tty0(user id: ): test
Sep 25 10:29:51 _gateway 94: *Sep 25 10:29:49: %SYS-3-USERLOG_ERR: Message from tty0(user id: ): test
Sep 25 10:29:51 _gateway 95: *Sep 25 10:29:50: %SYS-3-USERLOG_ERR: Message from tty0(user id: ): test
```
```

Wireshark 기반 네트워크 트래픽 심층 분석

특정 IP(172.16.1.2)에서 발생하는 불필요 트래픽 분석과 네트워크 서비스 탐색이해

Wireshark 기반 보안 취약점 식별 및 조치

특정 IP(172.16.1.2) 호스트 대상 트래픽 분석 및 서비스 식별

(ip.src == 192.168.10.0/24 and ip.dst ==172.16.1.2) or (ip.src == 192.168.20.0/24 and ip.dst ==172.16.1.2)

No.	Time	Source	Destination	Protocol	Lengt	Info
434	12.897802	192.168.10.2	172.16.1.2	HTTP/J…	298	HTTP/1.1 200 OK , JSON (application/json)
440	13.090277	192.168.10.2	172.16.1.2	TCP	66	9200 → 43052 [ACK] Seq=246 Ack=9650 Win=65280 Len=0 TSval=201721385 TSecr=1585686342
441	13.093940	192.168.10.2	172.16.1.2	TCP	66	9200 → 43052 [FIN, ACK] Seq=246 Ack=9650 Win=65280 Len=0 TSval=201721388 TSecr=1585686342
480	15.902168	192.168.10.2	172.16.1.2	TCP	66	9200 → 41382 [ACK] Seq=233 Ack=1819 Win=65280 Len=0 TSval=201724196 TSecr=1585689154
481	15.905643	192.168.10.2	172.16.1.2	TCP	66	9200 → 41382 [FIN, ACK] Seq=233 Ack=1819 Win=65280 Len=0 TSval=201724200 TSecr=1585689154
515	20.003656	192.168.10.2	172.16.1.2	TCP	74	9200 → 41386 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM TSval=201728298 TSecr=1585693256
521	20.005018	192.168.10.2	172.16.1.2	TCP	66	9200 → 41386 [ACK] Seq=1 Ack=3648 Win=65280 Len=0 TSval=201728299 TSecr=1585693257
522	20.122219	192.168.10.2	172.16.1.2	HTTP/J…	298	HTTP/1.1 200 OK , JSON (application/json)
558	22.132802	192.168.10.2	172.16.1.2	TCP	74	9200 → 33794 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM TSval=201730427 TSecr=1585695385
562	22.134546	192.168.10.2	172.16.1.2	TCP	66	9200 → 33794 [ACK] Seq=1 Ack=1560 Win=65280 Len=0 TSval=201730429 TSecr=1585695387
567	22.208730	192.168.10.2	172.16.1.2	HTTP/J…	298	HTTP/1.1 200 OK , JSON (application/json)
603	23.127336	192.168.10.2	172.16.1.2	TCP	66	9200 → 41386 [ACK] Seq=233 Ack=3649 Win=65280 Len=0 TSval=201731422 TSecr=1585696379
604	23.131071	192.168.10.2	172.16.1.2	TCP	66	9200 → 41386 [FIN, ACK] Seq=233 Ack=3649 Win=65280 Len=0 TSval=201731425 TSecr=1585696379
821	25.212167	192.168.10.2	172.16.1.2	TCP	66	9200 → 33794 [ACK] Seq=233 Ack=1561 Win=65280 Len=0 TSval=201733506 TSecr=1585698464
823	25.215829	192.168.10.2	172.16.1.2	TCP	66	9200 → 33794 [FIN, ACK] Seq=233 Ack=1561 Win=65280 Len=0 TSval=201733510 TSecr=1585698464
1065	30.006687	192.168.10.2	172.16.1.2	TCP	74	9200 → 33810 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM TSval=201738301 TSecr=1585703258
1075	30.008645	192.168.10.2	172.16.1.2	TCP	66	9200 → 33810 [ACK] Seq=1 Ack=4097 Win=65280 Len=0 TSval=201738303 TSecr=1585703261
1078	30.008989	192.168.10.2	172.16.1.2	TCP	66	9200 → 33810 [ACK] Seq=1 Ack=11337 Win=65280 Len=0 TSval=201738303 TSecr=1585703261
1080	30.009203	192.168.10.2	172.16.1.2	TCP	66	9200 → 33810 [ACK] Seq=1 Ack=14282 Win=65280 Len=0 TSval=201738304 TSecr=1585703261
1081	30.143796	192.168.10.2	172.16.1.2	HTTP/J…	317	HTTP/1.1 200 OK , JSON (application/json)
1173	32.844833	192.168.10.2	172.16.1.2	TCP	74	9200 → 43084 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM TSval=201741139 TSecr=1585706097
1178	32.846486	192.168.10.2	172.16.1.2	TCP	66	9200 → 43084 [ACK] Seq=1 Ack=1867 Win=65280 Len=0 TSval=201741141 TSecr=1585706098
1179	32.916771	192.168.10.2	172.16.1.2	HTTP/J…	298	HTTP/1.1 200 OK , JSON (application/json)
1183	33.146095	192.168.10.2	172.16.1.2	TCP	66	9200 → 33810 [ACK] Seq=252 Ack=14283 Win=65280 Len=0 TSval=201741440 TSecr=1585706398
1184	33.148753	192.168.10.2	172.16.1.2	TCP	66	9200 → 33810 [FIN, ACK] Seq=252 Ack=14283 Win=65280 Len=0 TSval=201741443 TSecr=1585706398

필터 설정: ip.addr == 172.16.1.2

내부 사용자에서 웹서버 실시간 패킷확인

핵심 성과 및 미래 보안 전략

네트워크 안정화와 보안 강화를 위한 단계별 실행 계획

네트워크 장비 초기화 완료로 안정적 운영 기반 확보

방화벽 보안 정책 강화로 외부 위협 차단 및 내부 보안 강화

MariaDB 서버 운영 및 효율적 데이터 관리 체계 구축

네트워크 **트래픽 분석**을 통해 성능 최적화 및 이상 탐지 시행

향후 **VPN 구축**과 보안 계층 다각화 계획

데이터베이스 백업·복구 정책 수립으로 데이터 안정성 강화

네트워크 성능 모니터링과 클라우드 보안 솔루션 연구를 통한 지속적 보안 역량 강화

지속적 학습과 혁신은 IT 보안 경쟁력 유지의 핵심 동력