

보안 네트워크 아키텍처 구축

프로젝트 결과 보고서

목차

1

프로젝트 개요

2

프로젝트 요구사항

3

아키텍처 구축 과정

4

프로젝트 결과

1. 프로젝트 개요

프로젝트 개요

프로젝트 목표

O1

다계층 보안 구조 설계

- DMZ 및 VLAN을 이용한 네트워크 분리
- 최소 권한 부여를 통한 접근 제어

O2

통합 관제 시스템 구축

- Beats를 이용한 로그 수집
- ELK 구축을 통한 실시간 로그 분석

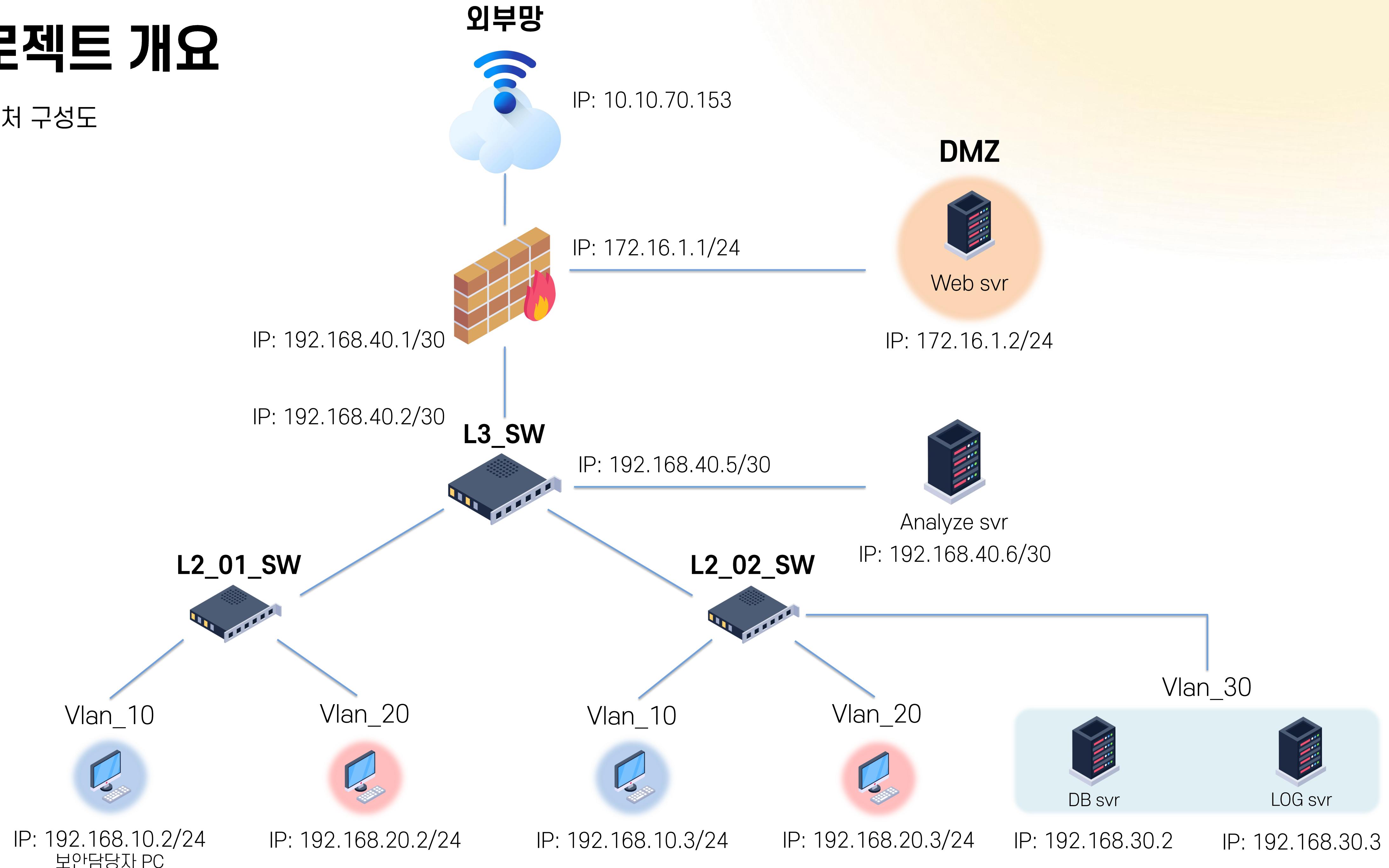
O2

운영 안정성 확보

- 역할 기반의 서버 분리
- 보안 담당자의 아키텍처 통합 관리

프로젝트 개요

아키텍처 구성도



2. 프로젝트 요구사항

프로젝트 요구사항



네트워크 구성

1. VLAN

- VLAN 10 내부 사용자 PC: 192.168.10.0/24
- 보안 담당자 PC IP: 192.168.10.2
- VLAN 20 내부 사용자 PC: 192.168.20.0/24
- VLAN 30 내부 서버: 192.168.30.0/24

2. DMZ

- 웹 서버: 172.16.1.2

3. 구간 네트워크

- 방화벽 <-> L3: 192.168.40.0/30
- L3 <-> 분석 서버: 192.168.40.4/30

프로젝트 요구사항



1. 웹 서버

- OS: Ubuntu Linux / Apache
- Agent: Filebeat, Metricbeat

4. 분석 서버

- OS: Windows /Wireshark
- 내부 사용자와 웹 서버 간 실시간 패킷 확인

2. DB 서버

- OS: Ubuntu Linux / MariaDB
- Agent: Filebeat, Metricbeat

5. NTP 서버

- time.bora.net

3. 로그 서버

- OS: Ubuntu Linux
- 각 네트워크 장비 및 서버의 syslog 수집

프로젝트 요구사항



1. 방화벽

[NAT 정책]

- 내부 사용자가 웹 서버 외부 IP로 접속 허용
- 내부 80, 443 포트로 접속 시 공인 IP를 웹 서버 사설 IP로 변환
- 내부 사용자와 웹 서버가 외부 네트워크 통신 시 공인 IP로 변환

[IPv4 기반 정책]

-
- 로그 서버의 데이터 수집 포트로의 접근 허용
- 내부 사용자가 외부의 모든 네트워크로의 접근 허용
- 웹 서버와 DB 서버 간 MariaDB 포트만을 이용한 접근 허용
- 보안 담당자 PC에서 모든 서버 및 장비로의 SSH 접속 허용

2. 스위치

- 관리자 모드 접속 및 원격 접속 패스워드 설정
- 비인가자 접근 경고 배너 설정

3. 아키텍처 구축 과정

아키텍처 구축 과정 - L2 스위치

VLAN 설정

1. VLAN 10, 20, 30 구성
2. VLAN에 IP 주소 할당

L2_01 스위치

```
interface Vlan1
no ip address
shutdown
!
interface Vlan10
no ip address
!
interface Vlan20
ip address 192.168.20.200 255.255.255.0
```

VLAN	Name	Status	Ports
1	default	active	Gil/0/21, Gil/0/22, Gil/0/23 Gil/1/1, Gil/1/2, Gil/1/3 Gil/1/4
10	VLAN_10	active	Gil/0/1, Gil/0/2, Gil/0/3 Gil/0/4, Gil/0/5, Gil/0/6 Gil/0/7, Gil/0/8, Gil/0/9 Gil/0/10
20	VLAN_20	active	Gil/0/11, Gil/0/12, Gil/0/13 Gil/0/14, Gil/0/15, Gil/0/16 Gil/0/17, Gil/0/18, Gil/0/19 Gil/0/20

L2_02 스위치

```
interface Vlan1
no ip address
!
interface Vlan10
no ip address
!
interface Vlan20
no ip address
!
interface Vlan30
ip address 192.168.30.200 255.255.255.0
```

L2_02_SW#show vlan			
VLAN	Name	Status	Ports
1	default	active	Gil/1/1, Gil/1/2, Gil/1/3 Gil/1/4
10	VLAN_10	active	Gil/0/1, Gil/0/2, Gil/0/3 Gil/0/4, Gil/0/5, Gil/0/6 Gil/0/7, Gil/0/8, Gil/0/9 Gil/0/10
20	VLAN_20	active	Gil/0/11, Gil/0/12, Gil/0/13 Gil/0/14, Gil/0/15, Gil/0/16 Gil/0/17, Gil/0/18, Gil/0/19 Gil/0/20
30	VLAN_30	active	Gil/0/21, Gil/0/22, Gil/0/23

아키텍처 구축 과정 – L3 스위치

VLAN 설정

1. VLAN 10, 20, 30 구성
2. VLAN에 IP 주소 할당

```
interface Vlan1
  no ip address
!
interface Vlan10
  ip address 192.168.10.1 255.255.255.0
!
interface Vlan20
  ip address 192.168.20.1 255.255.255.0
!
interface Vlan30
  ip address 192.168.30.1 255.255.255.0
```

서버 설정

1. 분석 서버 설정

```
L3_SW#show monitor session 1
Session 1
-----
Type          : Local Session
Source Ports :
  Both        : GigabitEthernet1/0/24
Destination Ports :
  GigabitEthernet1/0/13
Encapsulation : Native
Ingress       : Disabled
```

```
L3(config)#monitor session 1 source interface gigabitEthernet 1/0/24
L3(config)#monitor session 1 destination interface gigabitEthernet 1/0/13
```

네트워크 설정

1. 라우팅 설정
2. 인터페이스 IP 주소 설정

[라우팅 설정]

```
L3_SW(config)#ip route 0.0.0.0 0.0.0.0 192.168.40.1
L3_SW(config)#ip routing
```

```
Gateway of last resort is 192.168.40.1 to network 0.0.0.0

S*   0.0.0.0/0 [1/0] via 192.168.40.1
    192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.10.0/24 is directly connected, Vlan10
      L   192.168.10.1/32 is directly connected, Vlan10
    192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.20.0/24 is directly connected, Vlan20
      L   192.168.20.1/32 is directly connected, Vlan20
    192.168.30.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.30.0/24 is directly connected, Vlan30
      L   192.168.30.1/32 is directly connected, Vlan30
    192.168.40.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.40.0/30 is directly connected, GigabitEthernet1/0/24
      L   192.168.40.2/32 is directly connected, GigabitEthernet1/0/24
```

[인터페이스 IP 주소 설정]

GigabitEthernet1/0/13	192.168.40.5	YES manual	down
GigabitEthernet1/0/24	192.168.40.2	YES manual	up

아키텍처 구축 과정 – 전체 스위치

보안 설정

1. 배너 설정
2. 관리자 모드/SSH 접속 패스워드 설정

[배너 설정]

```
banner motd ^C
$=====
*** AUTHORIZED ACCESS ONLY ***
This system is the property of CBNET and is for authorized use only.
Unauthorized access is strictly prohibited. ALL connections are logged.
By proceeding, you consent to this monitoring. Violators will be prosecuted.
$=====
```

[패스워드 설정]

```
SW(config)#crypto key generate rsa
SW(config)#username root secret Ycdc
SW(config)#enable secret Ycdc
SW(config)#line vty 0 4
SW(config-line)#transport input ssh
SW(config-line)#login local
```

서버 설정

1. 로그 서버 설정
2. NTP 서버 설정

[로그 서버 설정]

```
logging trap debugging
logging facility local6
logging host 192.168.30.3
```

```
SW(config)#logging on
SW(config)#logging trap 7
SW(config)#logging 192.168.30.3
SW(config)#logging facility local6
```

[NTP 서버 설정]

```
ntp server 203.248.240.140
```

```
SW(config)#ntp server 203.248.240.140
SW(config)#clock timezone KST 9
```

아키텍처 구축 과정 – 전체 서버

보안 설정

1. SSH 설정
2. SSH 접속 사용자 생성

[SSH 설정]

- sudo apt install ssh
- sudo ufw allow from 192.168.10.2 to any port 22
- sudo ufw reload

[SSH 접속 사용자 생성]

- 계정명 sec_user
- 패스워드 Ycdc
- /etc/sudoers 파일에서 sec_user 추가

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

서버 설정

1. 로그 서버 설정
2. NTP 서버 설정

[로그 서버 설정]

```
#
# Include all config files in /etc/rsyslog.d/
#
$IncludeConfig /etc/rsyslog.d/*.conf
*.* @192.168.30.3:514
```

[NTP 서버 설정]

- sudo vi /etc/systemd/timesyncd.conf
- sudo systemctl restart systemd-timesyncd

```
[Time]
NTP=203.248.240.140
#FallbackNTP=ntp.ubuntu.com
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
#ConnectionRetrySec=30
#SaveIntervalSec=60
```

아키텍처 구축 과정 – DB 서버

DB 구성

1. DB 및 테이블 생성
2. 사용자 생성 및 권한 설정

전체 테이블

```
MariaDB [cb_db]> show tables;  
+-----+  
| Tables_in_cb_db |  
+-----+  
| comments        |  
| posts          |  
| users          |  
+-----+
```

게시글 테이블

```
MariaDB [cb_db]> desc posts;  
+-----+-----+-----+-----+-----+  
| Field    | Type     | Null | Key | Default      | Extra      |  
+-----+-----+-----+-----+-----+  
| id       | int(11)  | NO  | PRI | NULL         | auto_increment |  
| user_id   | int(11)  | NO  | MUL | NULL         | |  
| username  | varchar(50) | NO  |     | NULL         | |  
| title     | varchar(255) | NO  |     | NULL         | |  
| content   | text      | NO  |     | NULL         | |  
| filepath  | varchar(255) | YES |     | NULL         | |  
| created_at | timestamp | YES |     | current_timestamp() | |  
+-----+-----+-----+-----+-----+
```

사용자 테이블

```
MariaDB [cb_db]> desc users;  
+-----+-----+-----+-----+-----+  
| Field    | Type     | Null | Key | Default      | Extra      |  
+-----+-----+-----+-----+-----+  
| id       | int(11)  | NO  | PRI | NULL         | auto_increment |  
| username | varchar(50) | NO  | UNI | NULL         | |  
| email    | varchar(100) | NO  | UNI | NULL         | |  
| password | varchar(255) | NO  |     | NULL         | |  
| created_at | timestamp | YES |     | current_timestamp() | |  
+-----+-----+-----+-----+-----+
```

댓글 테이블

```
MariaDB [cb_db]> desc comments;  
+-----+-----+-----+-----+-----+  
| Field    | Type     | Null | Key | Default      | Extra      |  
+-----+-----+-----+-----+-----+  
| id       | int(11)  | NO  | PRI | NULL         | auto_increment |  
| post_id  | int(11)  | NO  | MUL | NULL         | |  
| user_id  | int(11)  | NO  | MUL | NULL         | |  
| username | varchar(50) | NO  |     | NULL         | |  
| content  | text      | NO  |     | NULL         | |  
| created_at | timestamp | YES |     | current_timestamp() | |  
+-----+-----+-----+-----+-----+
```

아키텍처 구축 과정 – DB 서버

DB 구성

1. DB 및 테이블 생성
2. 사용자 생성 및 권한 설정

web_user : 웹 서버 연동 계정

```
+-----+  
| Grants for web_user@172.16.1.2 |  
+-----+  
| GRANT USAGE ON *.* TO `web_user`@`172.16.1.2` IDENTIFIED BY PASSWORD '*5CA52AB4C57B7BFFB9B25164F713B3892DC839E5' |  
| GRANT SELECT, INSERT, UPDATE, DELETE ON `cb_db`.* TO `web_user`@`172.16.1.2` |  
+-----+
```

sec_user : SSH 접속 계정

```
+-----+  
| Grants for sec_user@localhost |  
+-----+  
| GRANT USAGE ON *.* TO `sec_user`@`localhost` IDENTIFIED BY PASSWORD '*5CA52AB4C57B7BFFB9B25164F713B3892DC839E5' |  
| GRANT ALL PRIVILEGES ON `cb_db`.* TO `sec_user`@`localhost` WITH GRANT OPTION |  
+-----+
```

metricbeat_user : Metricbeat 접속 계정

```
+-----+  
| Grants for metricbeat_user@localhost |  
+-----+  
| GRANT PROCESS, BINLOG MONITOR ON *.* TO `metricbeat_user`@`localhost` IDENTIFIED BY PASSWORD '*43426197F5C433D50745FED8081C6356DA037C92' |  
+-----+
```

아키텍처 구축 과정 – DB 서버

보안 설정

1. Iptables 설정

[iptables 내용]

- 보안관리자 PC에서 SSH 접속 허용
- 웹 서버에서 MariaDB 포트로의 통신 허용
- 전체 출발지에서 SSH 접속 시 “SSH_DROP”을 추가하여 syslog에 기록
- 전체 출발지에서 전체 목적지로 차단

```
root@db:/etc# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source          destination
ACCEPT     6    --  192.168.10.2      0.0.0.0/0        tcp dpt:22
ACCEPT     6    --  172.16.1.2       0.0.0.0/0        tcp dpt:3306
LOG        6    --  0.0.0.0/0       0.0.0.0/0        tcp dpt:22 LOG flags 0 level 4 prefix "SSH_DROP"
DROP       0    --  0.0.0.0/0       0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
```

아키텍처 구축 과정 - 로그 서버

RAID 구성

1. RAID-1로 syslog 디렉터리 구성

[파티션 생성]

- fdisk /dev/sdb
- fdisk /dev/sdc

```
sda      8:0    0    20G  0 disk
└─sda1   8:1    0    1M   0 part
└─sda2   8:2    0    20G  0 part  /
sdb      8:16   0    5G   0 disk
└─sdb1   8:17   0    5G   0 part
  └─md0   9:0    0    5G   0 raid1 /var/log/syslog
sdc      8:32   0    5G   0 disk
└─sdc1   8:33   0    5G   0 part
  └─md0   9:0    0    5G   0 raid1 /var/log/syslog
```

[마운트 설정]

- sudo vi /etc/fstab

```
# <file system> <mount point> <type> <options>        <dump> <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/2bdd24fc-64e5-4f4e-aaaae-766dbd7c4f8f / ext4 defaults 0 1
/swapp.img     none    swap    sw    0    0
UUID=88399507-5b89-4375-b748-d4b156ecd110      /var/log/syslog ext4 defaults 0 0
```

[RAID-1 어레이 설정 및 파일 시스템 생성]

- sudo mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sdb1 /dev/sdc1
- sudo mkfs.ext4 /dev/md0

```
/dev/md0:
      Version : 1.2
      Creation Time : Tue Sep 30 10:38:29 2025
      Raid Level : raid1
      Array Size : 5236736 (4.99 GiB 5.36 GB)
      Used Dev Size : 5236736 (4.99 GiB 5.36 GB)
      Raid Devices : 2
      Total Devices : 2
      Persistence : Superblock is persistent

      Update Time : Thu Oct  2 20:30:50 2025
      State : clean
      Active Devices : 2
      Working Devices : 2
      Failed Devices : 0
      Spare Devices : 0

      Consistency Policy : resync

              Name : LOG:0 (local to host LOG)
              UUID : aa9111e4:060974c5:da5c0583:e8fb3076
              Events : 19

      Number  Major  Minor  RaidDevice State
          0      8      17        0      active sync  /dev/sdb1
          1      8      33        1      active sync  /dev/sdc1
```

아키텍처 구축 과정 - 로그 서버

서버 구성

1. 로그 서버 설정

TCP/UDP를 이용하여 로그 수집

```
#####
##### MODULES #####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")
```

로그 파일 이름 규칙 생성

```
$template FILENAME, "/var/log/syslog/%fromhost-ip%/$YEAR%-$MONTH%-$DAY%_%fromhost-ip%.log"
*.* ?FILENAME
```

아키텍처 구축 과정 – 웹 서버

서버 구성

1. /var/www/html 아래에 cbnet 디렉터리 구성 후 필요한 php 파일 생성
2. 로그 서버 연동
3. 루트 디렉터리 설정

[로그 서버 연동]

- sudo vi /var/www/html/cbnet/db.php

```
<?php
// db.php

$db_host = "192.168.30.2";
$db_name = "cb_db";
$db_user = "web_user";
$db_pass = "Ycdc2024!";

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

[루트 디렉터리 설정]

- sudo vi /etc/apache2/sites-available/000-default.conf
- sudo vi /etc/apache2/apache2.conf

```
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # 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

<Directory /var/www/html/cbnet>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

아키텍처 구축 과정 - 방화벽

방화벽 설정

1. 라우팅 설정
2. NTP 서버 설정
3. 로그 서버 설정

[라우팅 설정]

- 외부 네트워크에서 내부 VLAN 및 분석 서버 통신 시 eth2를 통하여 전달
- 기본 라우팅으로 eth1을 통하여 외부망으로 통신

!	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용 여부	▼	종류	▼	테이블 ID	▼	라우팅 Mark	▼	출발지	▼	목적지	▼	게이트웨이	▼	NIF	▼	메트릭	▼	설명
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용		목적지 라우팅		255		0		0.0.0.0/0		192.168.10.0/24		192.168.40.2		eth2		0	VLAN_10	
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용		목적지 라우팅		255		0		0.0.0.0/0		192.168.20.0/24		192.168.40.2		eth2		0	VLAN_20	
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용		목적지 라우팅		255		0		0.0.0.0/0		192.168.30.0/24		192.168.40.2		eth2		0	VLAN_30	
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용		목적지 라우팅		255		0		0.0.0.0/0		192.168.40.4/30		192.168.40.2		eth2		0	ANALYZE_svr	
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용		기본 라우팅		256		0		0.0.0.0/0		0.0.0.0/0		10.10.70.1		eth1		0	외부망	

로그 서버 설정

IP 주소	▼	포트	▼	가상 시스템	▼	설명	▼
local server							
192.168.30.3		514		main		LOG_svr 설정	

NTP 서버 설정

시간	
시스템 시간	2025-10-12 13:12:59
표준 시간대	(UTC+09:00) Seoul, Tokyo
시간 설정 방법	<input checked="" type="radio"/> NTP 서버와 동기화
기본 서버 주소	203.248.240.103

아키텍처 구축 과정 - 방화벽

보안 설정

1. NAT 설정
2. IPv4 정책 설정

[NAT 설정]

- 내부 사용자가 웹 서버 외부 IP로 접속 허용
- 내부 80, 443 포트로 접속 시 공인 IP를 웹 서버 사설 IP로 변환
- 내부 사용자와 웹 서버가 외부 네트워크 통신 시 공인 IP로 변환

!	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용 여부	No.	NAT 아이디	↔	변환 전 출발지	변환 전 목적지	변환 전 서비스	변환 후 출발지	변환 후 목적지	변환 후 서비스	출발지 포트 재사용	규칙 개수	설명
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용	0	7		내부 사용자	FW_ext	all	FW_ext	WEB_svr	변환 안 함	사용 안 함	4	내부 사용자 웹 서버 외부 IP 접속 허용

!	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용 여부	No.	NAT 아이디	유형	↔	변환 전 출발지	송신 인터페이스	변환 후 출발지	변환 전 목적지	수신 인터페이스	변환 후 목적지	프로토콜	변환 전 포트	변환 후 포트	규칙 개수
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용	0	4	Static NAT	↔	변환 안 함	all	변환 안 함	10.10.70.153	eth1	WEB_svr	TCP	80-80	80-80	1
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용	1	3	Static NAT	↔	변환 안 함	all	변환 안 함	10.10.70.153	eth1	WEB_svr	TCP	443-443	443-443	1
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용	2	2	Dynamic NAT		내부 사용자	eth1	10.10.70.153	변환 안 함	-	변환 안 함	전체	1-65535	1-65535	2
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	사용	3	1	Dynamic NAT		WEB_svr	eth1	10.10.70.153	변환 안 함	eth3	변환 안 함	전체	1-65535	1-65535	1

아키텍처 구축 과정 - 방화벽

보안 설정

1. NAT 설정
2. IPv4 정책 설정

IPv4 정책 설정

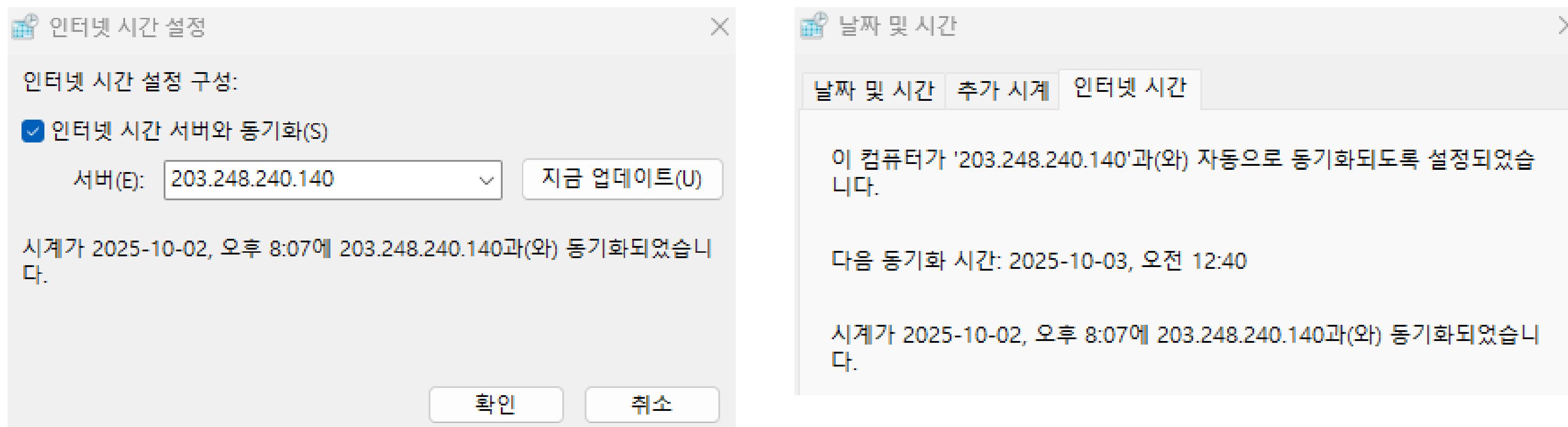
	No.	정책 아...	출발지	목적지	서비스	처리 방법	일정	Hits(1/7/30/60/90/180...)	연결 세션 확인 ...	설명
!	0	14	악성코드_유포지	all	all	- 차단	▶ always	0/0/0/0/0/0		악성코드 유포지 차단
	1	13	WEB_svr	LOG_svr	LOG Logstash	허용	▶ always	27/27/27/27/27/27		웹 서버에서 로그 서버로의 통신 허용
	2	12	WEB_svr	NTP_svr 내부전체	NTP	허용	▶ always	135/2,498/3,159/3,159/3,...	2일 전 (2025-10-0...)	NTP 서버 접속 허용
	3	11	Security_PC	WEB_svr	SSH	허용	▶ always	5/5/6/6/6/6	10일 전 (2025-09-...)	보안 관리자 PC에서 웹 서버로의 SSH 접속 허용
	4	10	all	WEB_svr	HTTP HTTPS	허용	▶ always	6/6/33/33/33/33	0일 전 (2025-10-1...)	외부망에서 웹 서버로의 통신 허용
	5	9	WEB_svr	DB_svr	WEB_DB	허용	▶ always	4/4/26/26/26/26	0일 전 (2025-10-1...)	웹 서버와 DB 서버 간 통신 허용
	6	8	WEB_svr	all	WEB	허용	▶ always	352/352/616/616/616/61...	0일 전 (2025-10-1...)	웹 서버의 인터넷 접속 허용
	7	7	all	SERVER_FARM	all	- 차단	▶ always	0/0/0/0/0/0		서버팜 접속 차단
	8	6	내부사용자	all	WEB	허용	▶ always	271/271/271/271/271/27...	0일 전 (2025-10-1...)	내부 사용자의 인터넷 접속 허용

아키텍처 구축 과정 – 사용자 PC 및 분석 서버

서버 설정

1. NTP 서버 설정

NTP 설정



아키텍처 구축 과정 – ELK

웹 서버 설정

1. Filebeat 설정
2. Metricbeat 설정

[Filebeat 설정]

- vi /etc/filebeat/filebeat.yml

```
- type: filestream
  id: access-log
  enabled: true
  paths:
    - /var/log/apache2/access.log
  tags: ["apache-access-log"]

- type: filestream
  id: auth-log
  enabled: true
  paths:
    - /var/log/auth.log
  tags: ["apache-auth-log"]
```

```
output.logstash:
  # The Logstash hosts
  hosts: ["192.168.30.3:5044"]
```

[Metricbeat 설정]

- vi /etc/metricbeat/modules.d/apache.yml

```
# Module: apache
# Docs: https://www.elastic.co/guide/en/beats/metricbeat/9.1/metricbeat-module-apache.html

- module: apache
  metricsets:
    - status
  period: 10s
  hosts: ["http://172.16.1.2/server-status?auto"]
  #username: "user"
  #password: "secret"
```

아키텍처 구축 과정 – ELK

DB 서버 설정

1. Filebeat 설정
2. Metricbeat 설정

[Filebeat 설정]

- vi /etc/filebeat/filebeat.yml

```
- type: filestream
  id: mariadb-query-log
  enabled: true
  paths:
    - /var/log/mysql/mysql.log
  tags: ["mariadb-query"]
  multiline.type: pattern
  multiline.pattern: '^[0-9]{6}\s'
  multiline.negate: true
  multiline.match: after

- type: filestream
  id: mariadb-auth-log
  enabled: true
  paths:
    - /var/log/auth.log
  tags: ["mariadb-ssh"]

output.logstash:
  # The Logstash hosts
  hosts: ["192.168.30.3:5044"]
```

[Metricbeat 설정]

- vi /etc/metricbeat/modules.d/apache.yml

```
# Module: mysql
# Docs: https://www.elastic.co/guide/en/beats/metricbeat/9.1/metricbeat-module-mysql.html

- module: mysql
  metricsets:
    - status
    # - galera_status
    - performance
    # - query
  period: 10s

  # Host DSN should be defined as "user:pass@tcp(127.0.0.1:3306)/"
  # or "unix(/var/lib/mysql/mysql.sock)/",
  # or another DSN format supported by <https://github.com/Go-SQL-Driver/MySQL/>.
  # The username and password can either be set in the DSN or using the username
  # and password config options. Those specified in the DSN take precedence.
  hosts: ["metricbeat_user:log-metric@tcp(127.0.0.1:3306)"]
```

아키텍처 구축 과정 – ELK

로그 서버 설정

1. 패스워드 설정
2. Logstash.conf 파일 설정
3. Data View 생성

[패스워드 설정]

- 보안 강화를 위해 ELK stack 별로 기본 패스워드에서 변경

```
ELASTIC_VERSION=9.1.4

## Passwords for stack users
#
# User 'elastic' (built-in)
#
# Superuser role, full access to cluster management and data indices.
# https://www.elastic.co/guide/en/elasticsearch/reference/current/built-in-users.html
ELASTIC_PASSWORD='log-elastic'

# User 'logstash_internal' (custom)
#
# The user Logstash uses to connect and send data to Elasticsearch.
# https://www.elastic.co/guide/en/logstash/current/ls-security.html
LOGSTASH_INTERNAL_PASSWORD='log-logstash'

# User 'kibana_system' (built-in)
#
# The user Kibana uses to connect and communicate with Elasticsearch.
# https://www.elastic.co/guide/en/elasticsearch/reference/current/built-in-users.html
KIBANA_SYSTEM_PASSWORD='log-kibana'
```

아키텍처 구축 과정 - ELK

로그 서버 설정

1. 패스워드 설정
2. Logstash.conf 파일 설정
3. Data View 생성

[Logstash.conf 파일 설정]

- 태그에 따라 개별적으로 index 설정

```
output {  
    if "mariadb-query" in [tags] {  
        if "Query" in [message] {  
            elasticsearch {  
                hosts => "elasticsearch:9200"  
                user => "logstash_internal"  
                password => "${LOGSTASH_INTERNAL_PASSWORD}"  
                index => "mariadb-query"  
            }  
        }  
        else {  
            elasticsearch {  
                hosts => "elasticsearch:9200"  
                user => "logstash_internal"  
                password => "${LOGSTASH_INTERNAL_PASSWORD}"  
            }  
        }  
    }  
}
```

```
    else if "apache-access-log" in [tags]{  
        elasticsearch {  
            hosts => "elasticsearch:9200"  
            user => "logstash_internal"  
            password => "${LOGSTASH_INTERNAL_PASSWORD}"  
            index => "apache-access-log"  
        }  
    }  
    else if "mariadb-ssh" in [tags] or "apache-auth-log" in [tags]{  
        elasticsearch {  
            hosts => "elasticsearch:9200"  
            user => "logstash_internal"  
            password => "${LOGSTASH_INTERNAL_PASSWORD}"  
            index => "auth-ssh-log"  
        }  
    }  
    else{  
        elasticsearch {  
            hosts => "elasticsearch:9200"  
            user => "logstash_internal"  
            password => "${LOGSTASH_INTERNAL_PASSWORD}"  
        }  
    }  
}
```

아키텍처 구축 과정 - ELK

로그 서버 설정

1. 패스워드 수정
2. Logstash.conf 파일 수정
3. Data View 생성

[Data View 생성]

- Index에 따라 Data View를 생성

apache-access-log	Index
auth-ssh-log	Index
logs-generic-default	Data stream
mariadb-query	Index

Data Views

Create and manage the data views that help you retrieve your data from Elasticsearch.

Search...

- Name ↑
- metricbeat ⓘ Default
- apache-access ⓘ
- auth-ssh-log ⓘ
- mariadb-query ⓘ

4. 프로젝트 결과

프로젝트 결과

스위치

- SSH 로그인 및 관리자 모드 접속 시 비밀번호 설정 확인
- 비인가자 접속 금지 배너 확인

```
[*] login as: root
[*] Keyboard-interactive authentication prompts from server:
| Password:
[*] End of keyboard-interactive prompts from server

*** AUTHORIZED ACCESS ONLY ***
This system is the property of CBNET and is for authorized use only.
Unauthorized access is strictly prohibited. All connections are logged.
By proceeding, you consent to this monitoring. Violators will be prosecuted.

L2_01_SW>enable
Password:
L2_01_SW#
```

프로젝트 결과

분석 서버

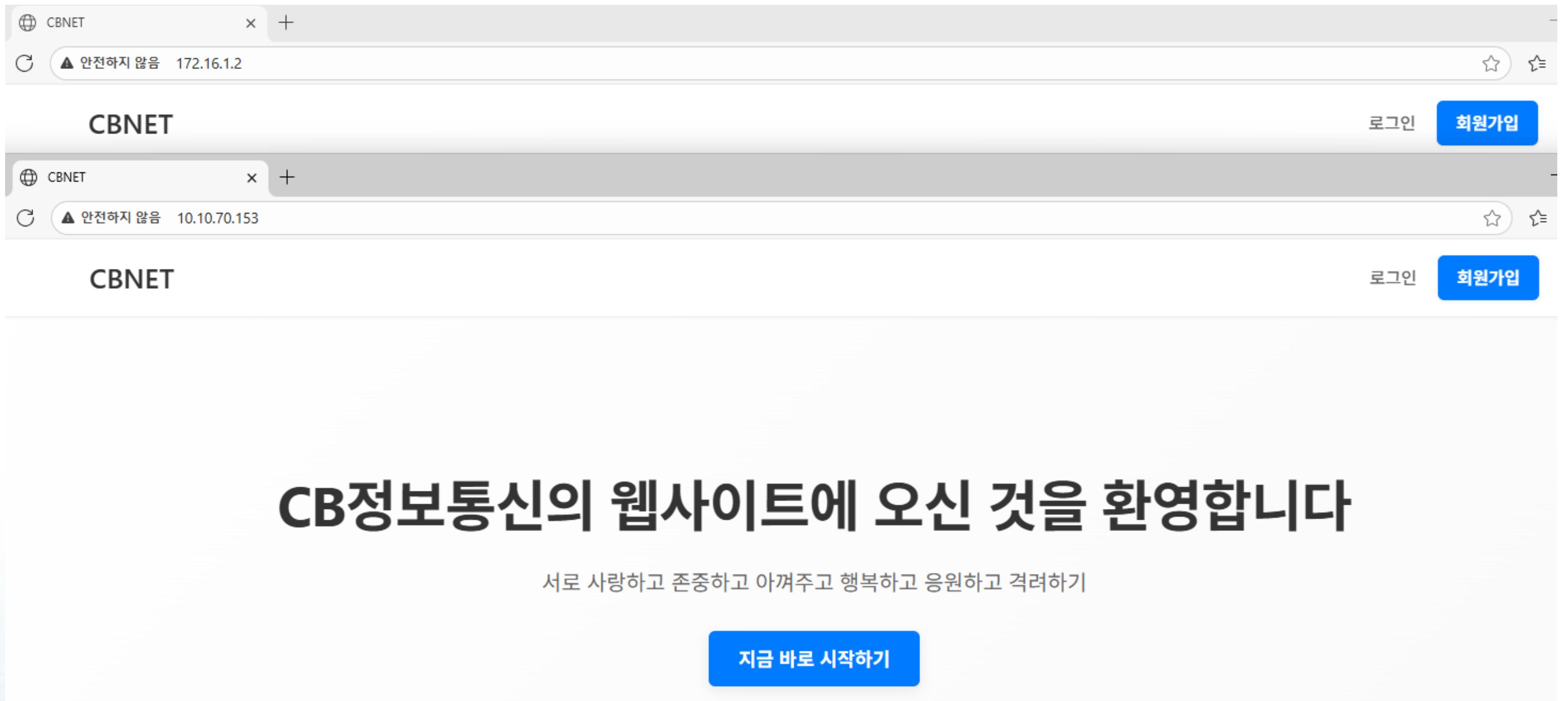
- 내부 사용자에서 웹 서버로의 실시간 패킷 확인

(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 → 43004 [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 → 43004 [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

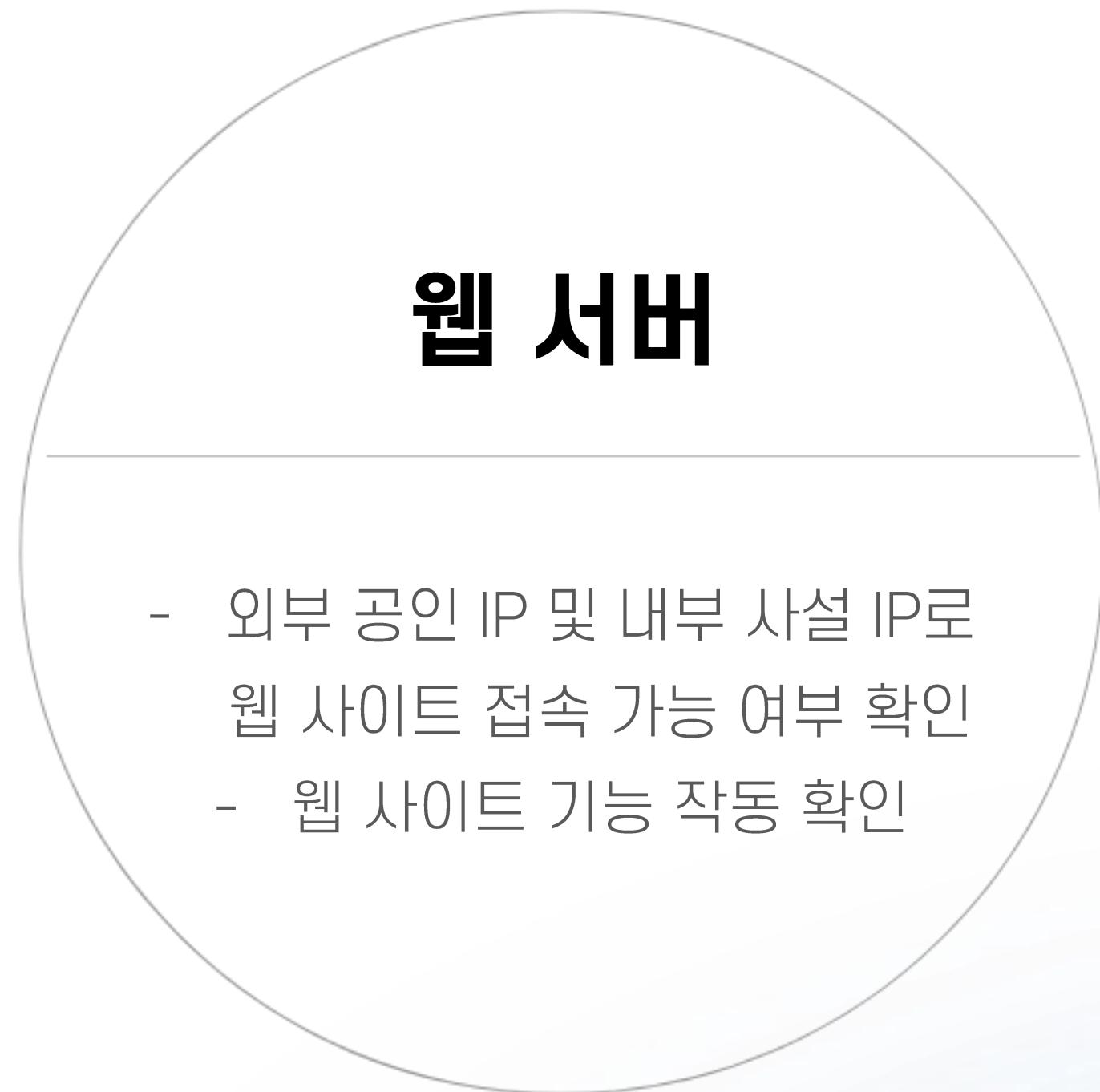
프로젝트 결과

웹 서버

- 외부 공인 IP 및 내부 사설 IP로 웹 사이트 접속 가능 여부 확인
- 웹 사이트 기능 작동 확인



프로젝트 결과



게시판

글쓰기

제목+내용 ▾

검색어를 입력하세요

검색

번호	제목	작성자	작성일
3	2025.10	test	2025-10-10 16:38:08

1

2025.10

작성자: test | 작성일: 2025-10-10 16:38:08

cPanel 구축

목록으로

수정

삭제

댓글 (1)

test 2025-10-10 16:39:11

댓글 기능

댓글을 입력하세요...

댓글 작성

프로젝트 결과

DB 서버

- 관리자 외 SSH 접속 차단 여부 및 로그 확인
- 웹 서버 연동 확인

PS C:\Users\user\Desktop> ipconfig

Windows IP 구성

이더넷 어댑터 이더넷 2:

연결별 DNS 접미사 :
링크-로컬 IPv6 주소 : fe80::4972:44e2:b611:627a%5
IPv4 주소 : 192.168.30.10
서브넷 마스크 : 255.255.255.0
기본 게이트웨이 : 192.168.30.1

PS C:\Users\user\Desktop>

192.168.30.2 - PuTTY

PuTTY Fatal Error

Network error: Connection timed out

확인

```
2025-10-10T16:23:45.828984+09:00 db kernel: SSH_DROPIN=ens33 OUT= MAC=00:0c:29:8f:3d:66:a0:36:bc:ca:  
b1:e7:08:00 SRC=192.168.30.10 DST=192.168.30.2 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=21137 DF PROTO=TCP  
SPT=57619 DPT=22 WINDOW=65535 RES=0x00 SYN URGP=0  
2025-10-10T16:23:46.837466+09:00 db kernel: SSH_DROPIN=ens33 OUT= MAC=00:0c:29:8f:3d:66:a0:36:bc:ca:  
b1:e7:08:00 SRC=192.168.30.10 DST=192.168.30.2 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=21138 DF PROTO=TCP  
SPT=57619 DPT=22 WINDOW=65535 RES=0x00 SYN URGP=0  
2025-10-10T16:23:48.840362+09:00 db kernel: SSH_DROPIN=ens33 OUT= MAC=00:0c:29:8f:3d:66:a0:36:bc:ca:  
b1:e7:08:00 SRC=192.168.30.10 DST=192.168.30.2 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=21139 DF PROTO=TCP  
SPT=57619 DPT=22 WINDOW=65535 RES=0x00 SYN URGP=0
```

프로젝트 결과

DB 서버

- 관리자 외 SSH 접속 차단 여부 및 로그 확인
- 웹 서버 연동 확인

```
MariaDB [cb_db]> select * from users;
+----+-----+-----+-----+-----+
| id | username | email           | password          | created_at      |
+----+-----+-----+-----+-----+
| 1  | test     | test@example.com | $2y$10$q7eEbrsaLry7htBBKS5Z9e/2aTmQ9A411wYQzleI46XZCqJiSz9G | 2025-10-08 11:01:47 |
| 2  | test2    | test2@naver.com  | $2y$10$Dj1sXgNpq82dpVJpCC1Af4yKXQk.H4YrQ7eJcUaqXEJ/R5Z6A7j6 | 2025-10-09 14:35:18 |
+----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [cb_db]> select * from posts;
+----+-----+-----+-----+-----+-----+
| id | user_id | username | title   | content        | filepath | created_at      |
+----+-----+-----+-----+-----+-----+
| 3  | 1       | test     | 2025.10 | cbnet 구축     | NULL    | 2025-10-10 16:38:08 |
+----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [cb_db]> select * from comments;
+----+-----+-----+-----+-----+
| id | post_id | user_id | username | content        | created_at      |
+----+-----+-----+-----+-----+
| 3  | 3       | 1       | test     | 댓글 가능     | 2025-10-10 16:39:11 |
+----+-----+-----+-----+-----+
```

프로젝트 결과

로그 서버

- /var/log/syslog 내 로그 파일 확인

```
drwxr-xr-x 2 syslog syslog 4096 Oct  2 00:00 127.0.0.1
drwxr-xr-x 2 syslog syslog 4096 Oct  2 00:00 172.16.1.2
drwxr-xr-x 2 syslog syslog 4096 Sep 30 11:54 192.168.20.200
drwxr-xr-x 2 syslog syslog 4096 Sep 30 11:53 192.168.30.1
drwxr-xr-x 2 syslog syslog 4096 Oct  2 00:00 192.168.30.2
drwxr-xr-x 2 syslog syslog 4096 Sep 30 11:54 192.168.30.200
drwxr-xr-x 2 syslog syslog 4096 Oct  2 00:00 192.168.40.1
```

```
-rw-r----- 1 syslog adm 1314816 Sep 30 23:59 2025-09-30_172.16.1.2.log
-rw-r----- 1 syslog adm 4519837 Oct  1 23:59 2025-10-01_172.16.1.2.log
-rw-r----- 1 syslog adm 11737549 Oct  2 20:34 2025-10-02_172.16.1.2.log
```

```
root@LOG:/var/log/syslog/192.168.20.200# tail -3 2025-09-30_192.168.20.200.log
2025-09-30T11:54:03.137122+09:00 192.168.20.200 56: Sep 30 11:54:02: %SYS-3-USERLOG_ERR: Message from tty2(user id: root): test
2025-09-30T11:54:03.137122+09:00 192.168.20.200 57: Sep 30 11:54:02: %SYS-3-USERLOG_ERR: Message from tty2(user id: root): test
2025-09-30T11:54:03.137122+09:00 192.168.20.200 58: Sep 30 11:54:02: %SYS-3-USERLOG_ERR: Message from tty2(user id: root): test
```

프로젝트 결과

관리자 PC

- 각 서버로에 sec_user 계정으로 SSH 접속 가능 여부 확인

```
이더넷 어댑터 이더넷:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::5705:4648:b2f6:6b0b%10  
IPv4 주소 . . . . . . . . . . : 192.168.10.2  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . : 192.168.10.1  
  
이더넷 어댑터 VMware Network Adapter VMnet1:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::55d0:33b9:dc70:133b%14  
IPv4 주소 . . . . . . . . . . : 192.168.239.1  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . :  
  
sec_user@db:~$
```

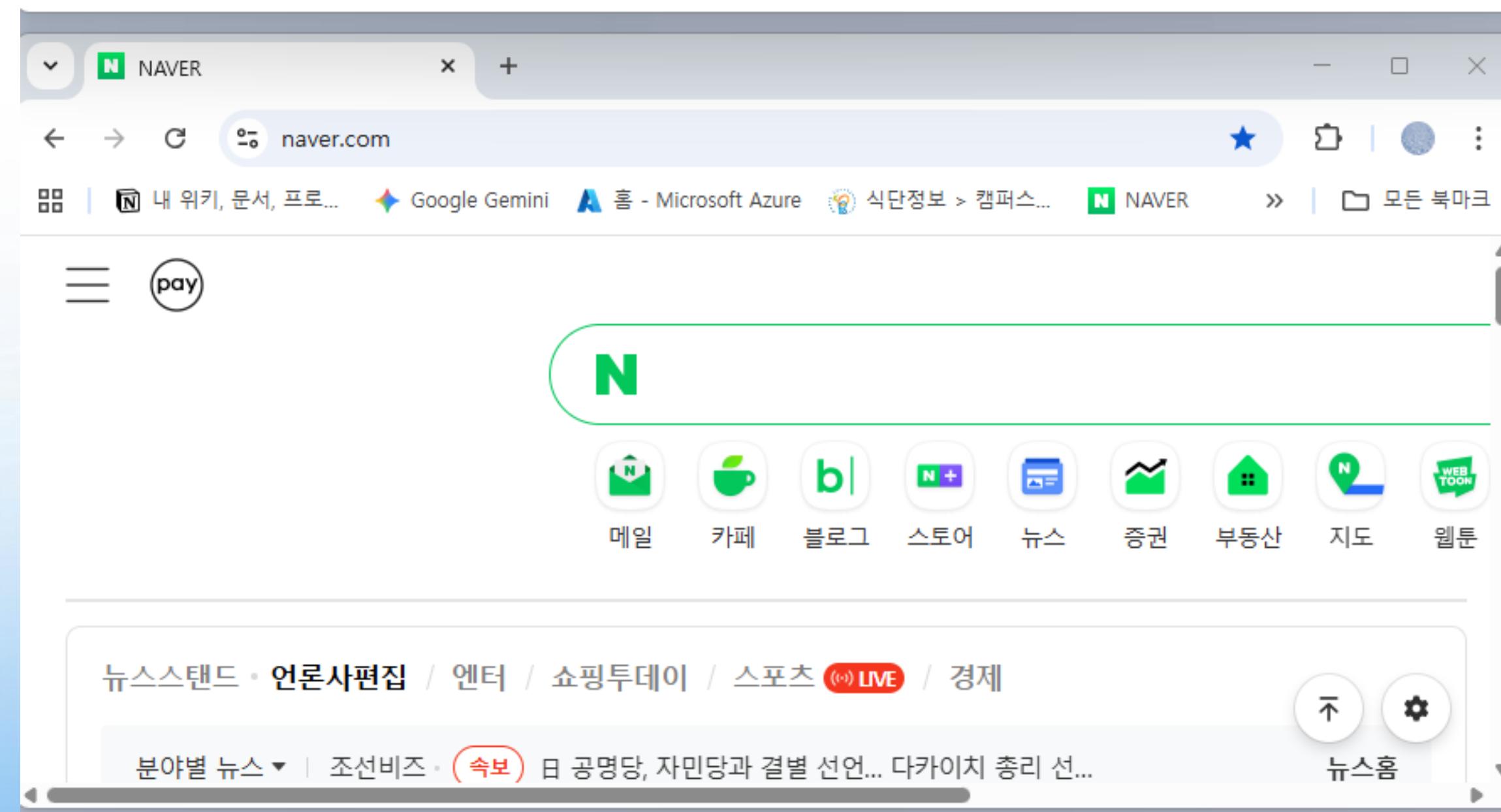
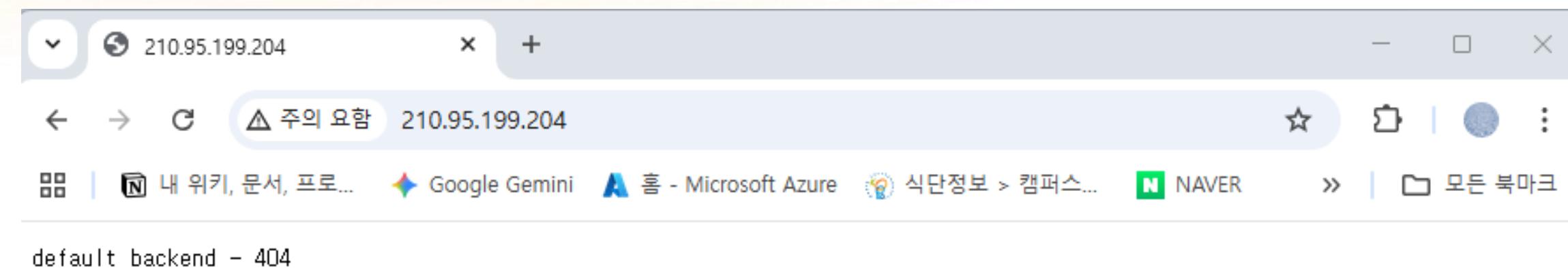
```
PS C:\Users\user> ipconfig  
Windows IP 구성  
  
이더넷 어댑터 이더넷:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::5705:4648:b2f6:6b0b%10  
IPv4 주소 . . . . . . . . . . : 192.168.10.2  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . : 192.168.10.1  
  
이더넷 어댑터 VMware Network Adapter VMnet1:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::55d0:33b9:dc70:133b%14  
IPv4 주소 . . . . . . . . . . : 192.168.239.1  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . :  
  
sec_user@web:~$
```

```
PS C:\Users\user> ipconfig  
Windows IP 구성  
  
이더넷 어댑터 이더넷:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::5705:4648:b2f6:6b0b%10  
IPv4 주소 . . . . . . . . . . : 192.168.10.2  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . : 192.168.10.1  
  
이더넷 어댑터 VMware Network Adapter VMnet1:  
연결별 DNS 접미사 . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::55d0:33b9:dc70:133b%14  
IPv4 주소 . . . . . . . . . . : 192.168.239.1  
서브넷 마스크 . . . . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . . . :  
  
sec_user@LOG:~$
```

프로젝트 결과

내부 사용자

- 가상 악성코드 유포지 차단 확인



프로젝트 결과

ELK

- Data View에서 로그 수집 확인
- Kibana 대시보드 확인



프로젝트 결과

ELK

- Data View에서 로그 수집 확인
- Kibana 대시보드 확인

