위·변조 감시 시스템

2018. 11. 7

중부대학교 정보보호학과

지도교수 : 양환석 교수님

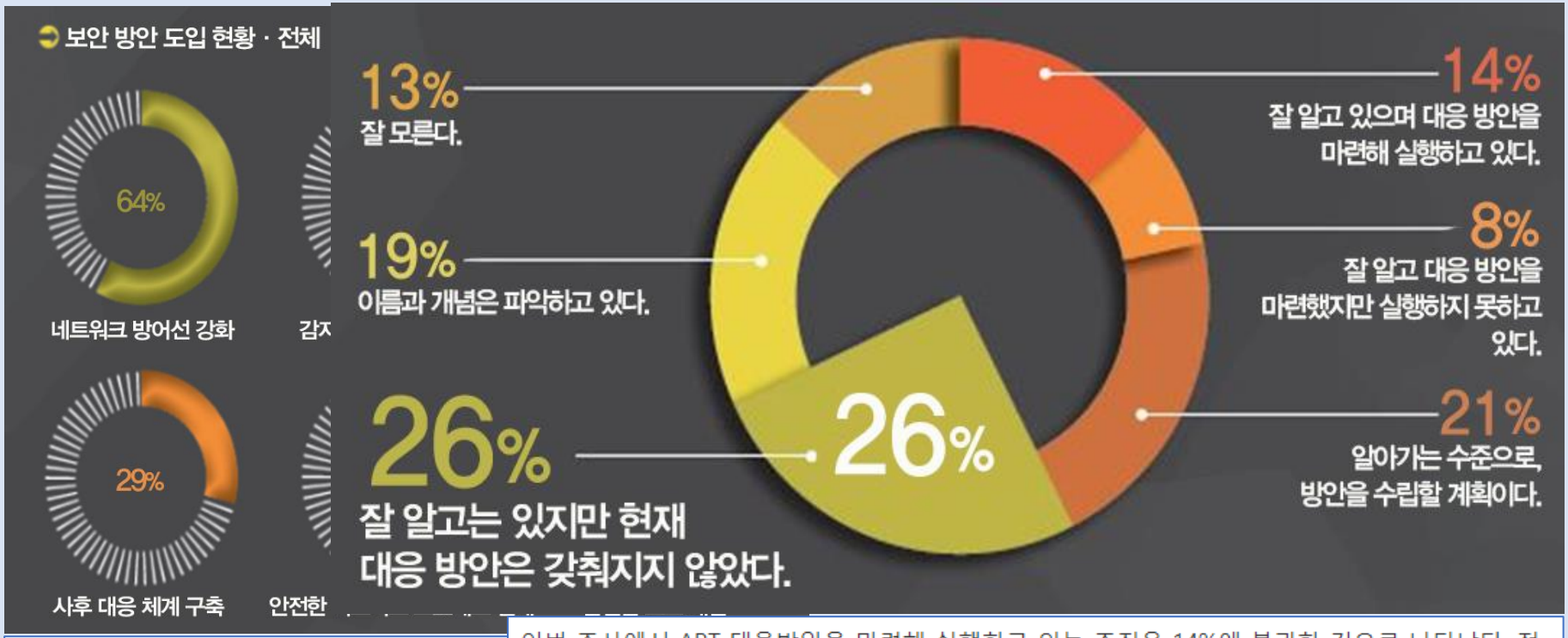
감프(감시프로그램)
(서규현, 조은서, 설희운, 장정윤, 류현호)

목 차

조원 편성

이 름	담당 임무
서규현	프로젝트 총괄, 서버 연동, hash값 업데이트, DB구축
류현호	서버 구축, 파일 차단/변경, 관리자 알림
조은서	위•변조 파일 저장, 관리자 알림
장정윤	GUI 설계 및 구현, 파일 차단/변경, 외부 공격 IP차단 외부 업로드파일 위•변조 감시 기능 구현
설희운	파일 차단/변경, DB구축

주제 선정



이번 조사에서 조직들은 약 3.1개의 보안 솔
이러스(74%), 스팸메일 필터링 솔루션(51%),
2%), APT 대응 솔루션(19%) 도입은 미흡했다
4%)을 보였으나, APT 대응 솔루션 도입률은

이번 조사에서 보안 수준에 대한 자신감은 현
. 자사가 APT에 충분히 대응할 수 있냐는 질문에 긍정적으로는 부정적인 견해가 52%, 중립적인 견
해가 48%로 균형을 이뤘다.

이번 조사에서 APT 대응방안을 마련해 실행하고 있는 조직은 14%에 불과한 것으로 나타났다. 전체 응답자의 86%가 APT 대응 방안이 없다고 답변했다. 중소규모 조직(8%)은 물론, 대규모 조직 또한 대응 방안을 마련, 실행한다는 응답은 19%에 불과했다.

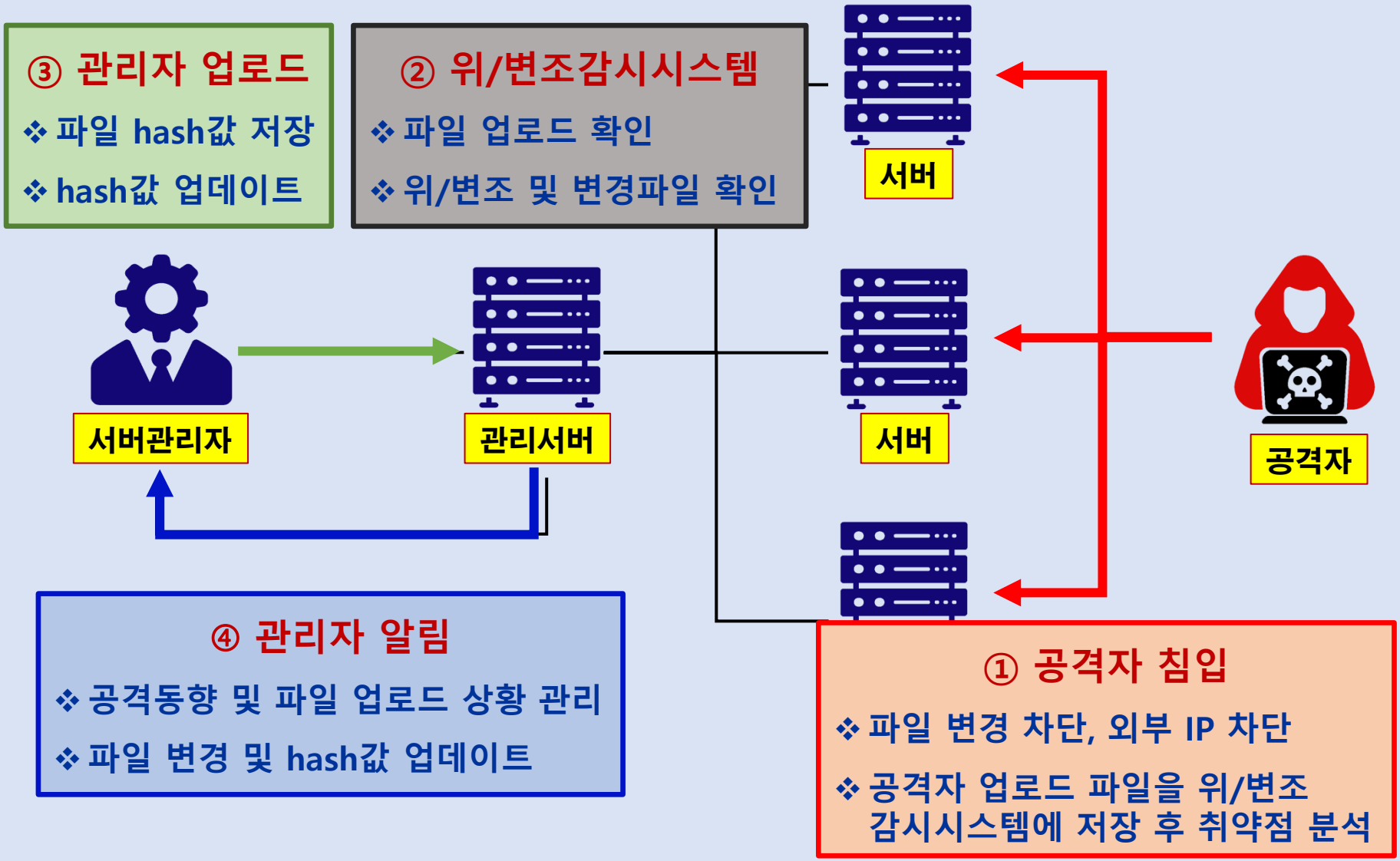
APT에 대해 잘 알고는 있지만 현재 대응방안은 갖춰지지 않았다(26%), 알아가는 수준으로 방안을 수립할 계획이다(21%), 잘 알고 대응 방안을 마련했지만 실행하지 못하고 있다(8%) 등으로 55%가 APT에 대해 잘 알고 있지만, 대응 방안 수립에는 미흡한 것으로 조사됐다.

보안시스템 도입 저조

대응방안 미흡

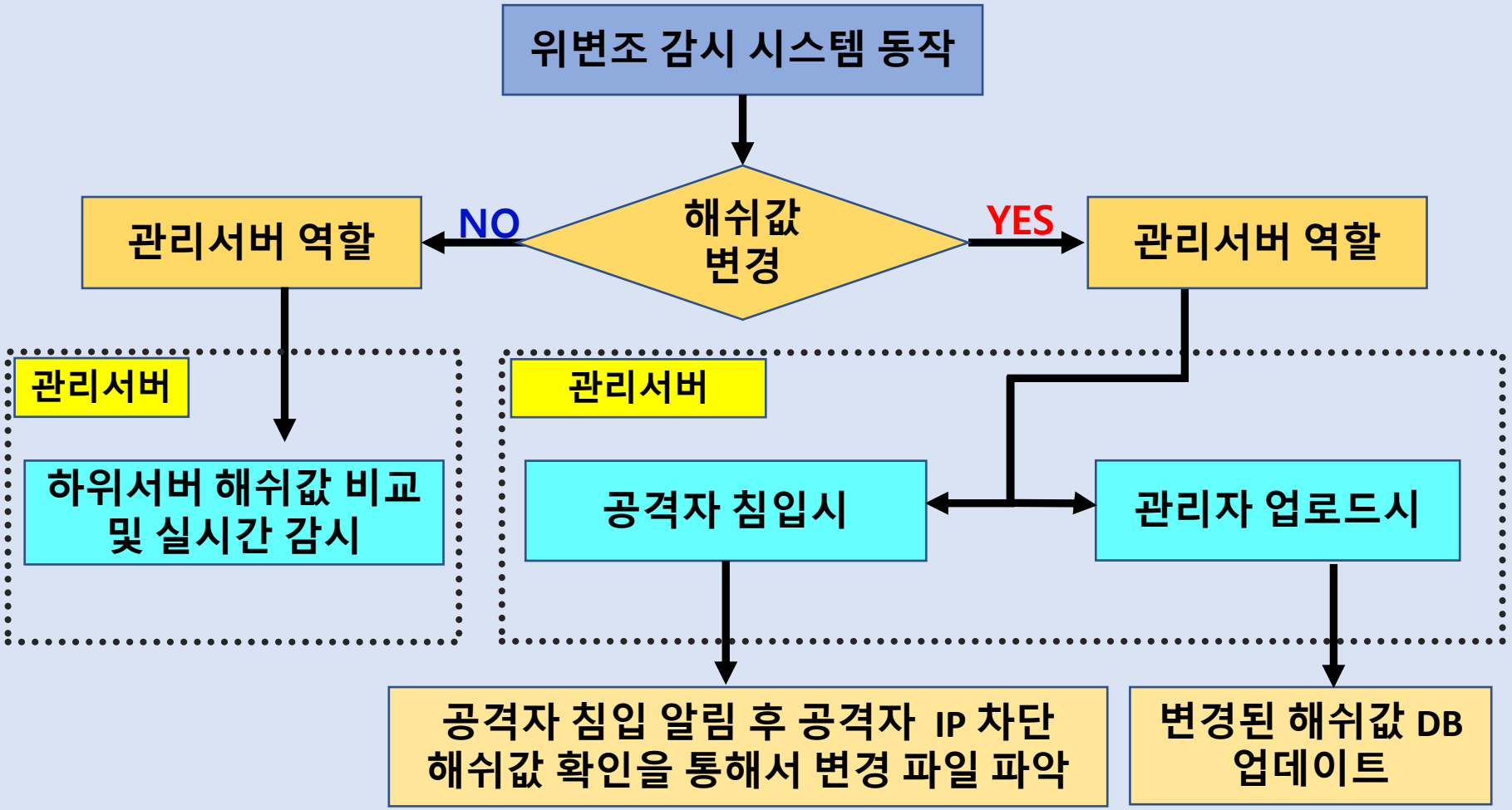
바이러스나 악성 코드 등에 의한 파일 위/변조 감시시스템을 구현

구상도(1/2)


③ 관리자 업로드
❖ 파일 hash값 저장
❖ hash값 업데이트
② 위/변조감시시스템
❖ 파일 업로드 확인
❖ 위/변조 및 변경파일 확인
서버
서버관리자
관리서버
서버
공격자
④ 관리자 알림
❖ 공격동향 및 파일 업로드 상황 관리
❖ 파일 변경 및 hash값 업데이트
① 공격자 침입
❖ 파일 변경 차단, 외부 IP 차단
❖ 공격자 업로드 파일을 위/변조 감시시스템에 저장 후 취약점 분석

구상도(2/2)

작업 계통도


위변조 감시 시스템 동작
해쉬값 변경
NO
YES
관리서버 역할
관리서버 역할
관리서버
하위서버 해쉬값 비교 및 실시간 감시
관리서버
공격자 침입시
관리자 업로드시
공격자 침입 알림 후 공격자 IP 차단
해쉬값 확인을 통해서 변경 파일 파악
변경된 해쉬값 DB 업데이트

추진 경과

구분 \ 기간(월)	3	4	5	6	7	8	9	10
구상도 설계								
자료 조사								
서버 구축								
프로그램 개발								
연동체계 구축								
프로그램 수정/보완								
연동 확인								
시스템 종합								
PPT, 보고서 작성								

개발 환경 및 시스템 개발(1/11)

개발 환경

운영 체제

개발 언어

개발 환경 및 시스템 개발(2/11)

관리서버 구축

❖ Hash값 저장을 위한 관리서버 DB 구축

DB 구축

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 5.7.24-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| subhash            |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

Table 생성

```
mysql> use subhash;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+------------------+
| Tables_in_subhash |
+------------------+
| sub_server1      |
+------------------+
1 row in set (0.00 sec)
```

개발 환경 및 시스템 개발(3/11)

관리서버와 하위서버 연동(1/2)

❖ 관리서버에서 하위서버로 save명령어 실행 요청

```
#!/bin/bash

ssh root@192.168.40.134 ./save.sh
```

❖ 관리서버에서 하위서버로 hash값 파일 요청

```
#!/bin/bash

ssh root@192.168.40.134 ./md5.sh

scp root@192.168.40.134:/root/md5test.txt /var/lib/mysql-files/log/

read pw

mysql -uroot -p$pw -D MD5 -e"LOAD DATA INFILE '/var/lib/mysql-files/log/md5test.
txt' INTO TABLE submd5 FIELDS TERMINATED BY ' ' LINES TERMINATED BY '\n';"
```

❖ 관리서버에서 하위서버로 위변조 감시명령어 실행

```
#!/bin/bash

ssh root@192.168.40.134 "./intDtc.sh start &"
```

- ❖ 관리서버에서 하위서버로 명령어 실행 시 SSH 이용
- ❖ 파일 요청 및 수신 시 SCP 사용
- ❖ 하위서버 hash값은 관리서버에 저장

개발 환경 및 시스템 개발(4/11)

관리서버와 하위서버 연동(2/2)

❖hash값 업데이트 및 관리서버로 전송

```
#!/bin/bash

ori="./md5test.txt"
copy="./md5copy.txt"

cp "$ori" "$copy"

NOW=$(date +"%Y%m%d%H%M%S")

md5log=`find /etc/ -type f -printf '"%p"\n'|xargs md5sum -b > ./md5compare$NOW.txt`;

md5compare=`find /etc/ -type f -printf '"%p"\n'|xargs md5sum -b > ./md5compare.txt`;
compare="./md5compare.txt"

log="./md5compare$NOW.txt"

if [ -z "`diff $copy $compare`" ]
then
        echo "변경된 값이 없습니다."
else
        echo "변경된 값이 있습니다."
        #diff $copy $compare
        cp "$compare" "$ori"
        scp $log root@192.168.40.141:/root
```

❖ 서버 관리자가 하위서버 파일을 변경할 경우 변경된 파일의 hash값을 업데이트 후 관리서버로 전송

개발 환경 및 시스템 개발(5/11)

GUI 처리부 구현

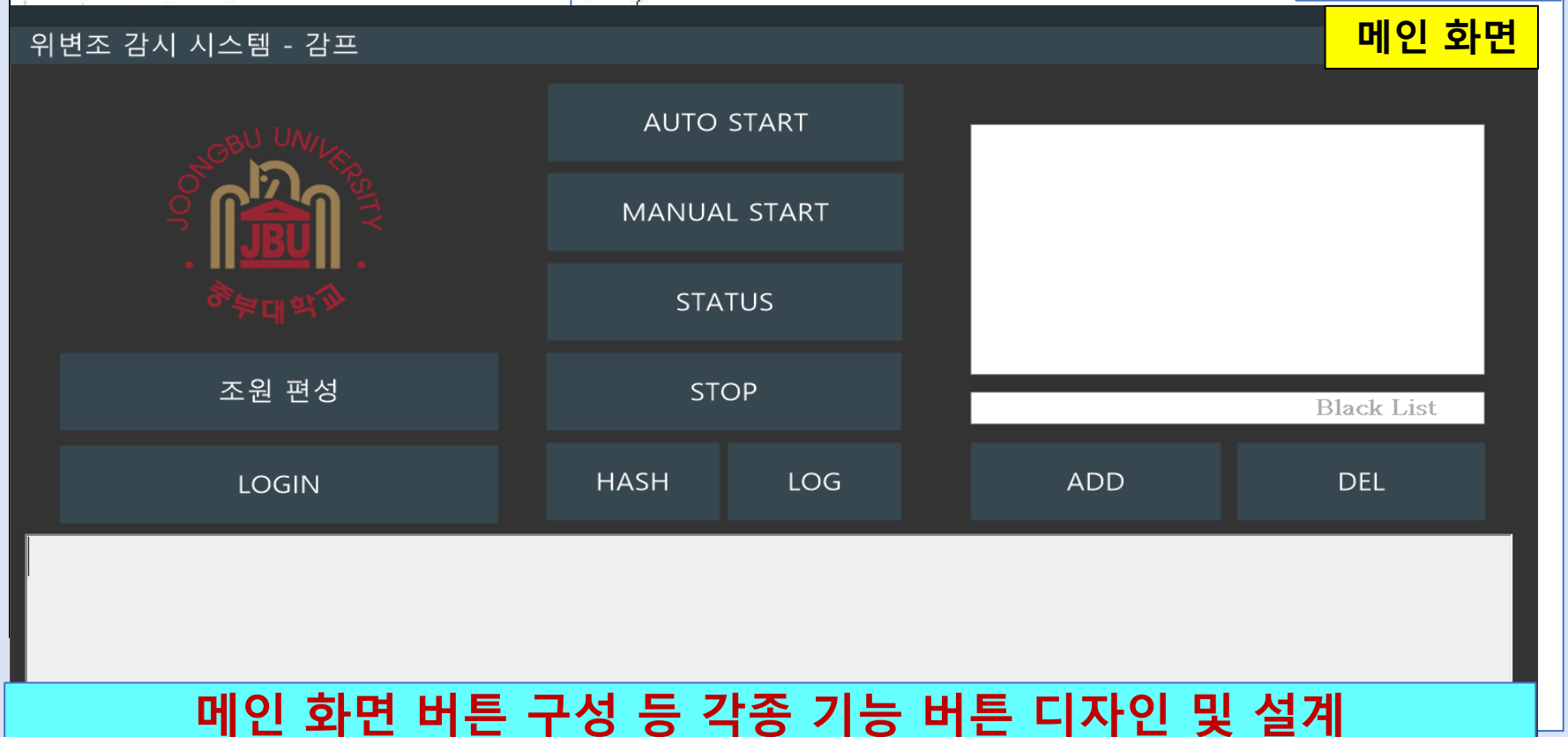
```
private void StatusBtn_Click(object sen
{
    if (CheckConnectedSsh() == false)
    {
```

기능 버튼 구성

```
private void OnCmd_Add(object sender, EventArgs e)        {
    if (ipListBox.Items.Contains(ipTextBox.Text) == true)
```

IP 추가/삭제

메인 화면



메인 화면 버튼 구성 등 각종 기능 버튼 디자인 및 설계

개발 환경 및 시스템 개발(6/11)

로그인 기능처리

로그인 유효성 확인

정상 로그인 확인

```
private void LoginBtn_Click(object sender, EventArgs e
{
    var loginForm = ne
    loginForm.ShowDial

    var splitString =
    if (splitString.Le
    {
        AppendToOutput
    }

    mUserName = splitS
    mHost = splitStrin
    mPassword = loginF
```

```
try {
    if (string.IsNullOrEmpty(loginForm.PpkFilePath) == true)
        mSshClient = new SshClient(mHost, mUserName, mPassword);
        mScpClient = new ScpClient(mHost, mUserName, mPassword); }
    else {
        mPPK = new PrivateKeyFile(loginForm.PpkFilePath);
        mSshClient = new SshClient(mHost, mUserName, mPPK);
        mScpClient = new ScpClient(mHost, mUserName, mPPK);  }

    mSshClient.ConnectionInfo.Timeout = new TimeSpan(0, 0, 5);
    mScpClient.ConnectionInfo.Timeout = new TimeSpan(0, 0, 5);

    AppendToOutputTextBox("로그인 중...");
    mSshClient.Connect();
    mScpClient.Connect();
    AppendToOutputTextBox("로그인 성공");

    var ret = mSshClient.RunCommand(string.Format("[ ! -d {0} ] && echo not_found",
    if (ret.Result == "not_found₩n") {
        RunCommand($"mkdir {GetScriptsPath()}");  }
    else {
```

로그인 정보 확인/인증 등 로그인 처리 기능 설계

개발 환경 및 시스템 개발(7/11)

HASH 값 관리

❖ 관리서버에서 하위서버로 hash값 파일 요청 및 DB에 저장

```
#!/bin/bash

ssh root@192.168.40.134 ./md5.sh

scp root@192.168.40.134:/root/md5test.txt /var/lib/mysql-files/log/

read pw

mysql -uroot -p$pw -D MD5 -e"LOAD DATA INFILE '/var/lib/mysql-files/log/md5test.txt'
 INTO TABLE submd5 FIELDS TERMINATED BY ' ' LINES TERMINATED BY '\n';"
```

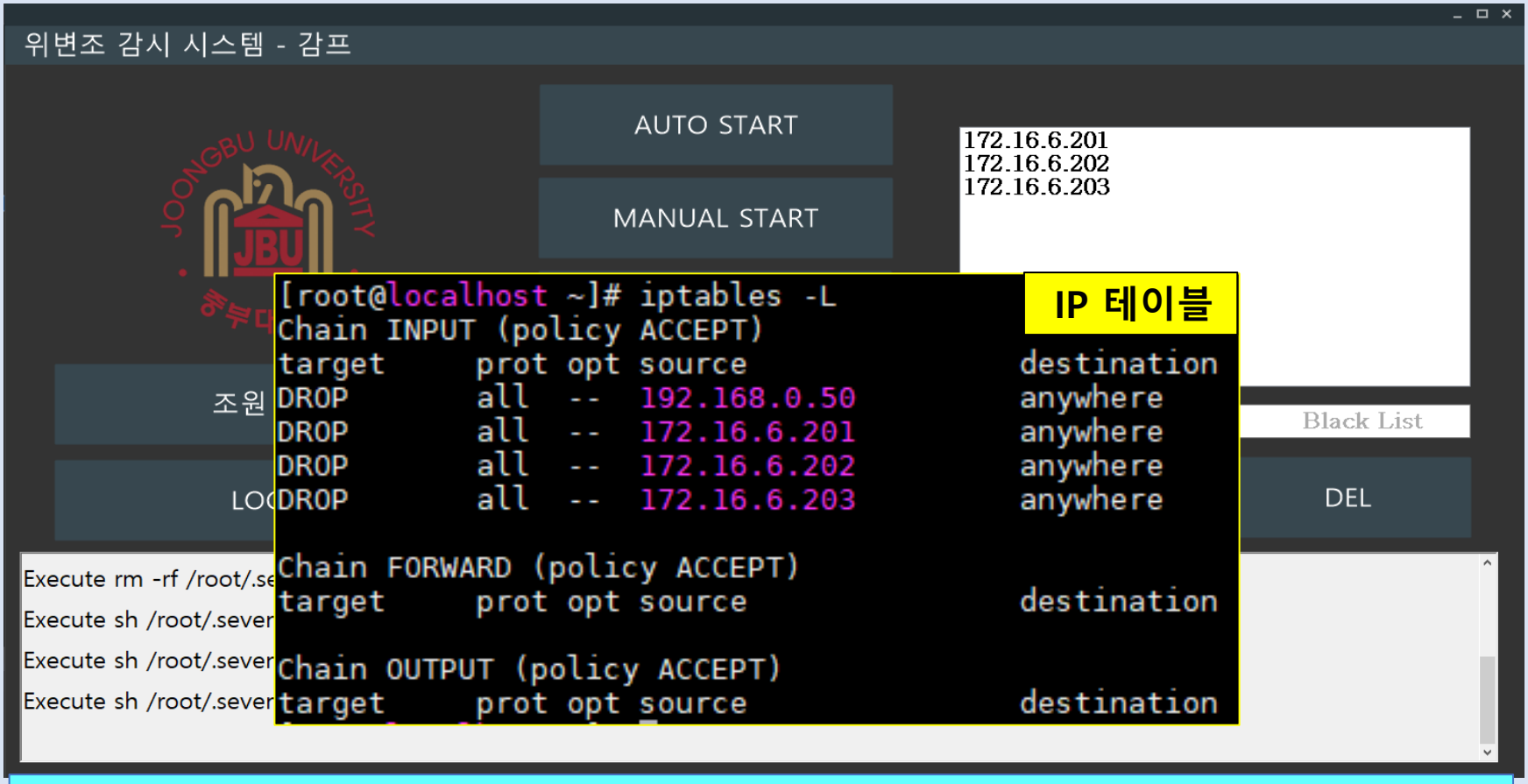
```
mysql> describe sub_server1
    -> ;
+------------+------+------+-----+---------+-------+
| Field      | Type | Null | Key | Default | Extra |
+------------+------+------+-----+---------+-------+
| Hash_value | text | YES  |     | NULL    |       |
| Path       | text | YES  |     | NULL    |       |
+------------+------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

❖ 관리자가 하위서버에 hash값 요청 후 하위서버의 hash값을 DB에 저장

❖ 관리서버 DB의 테이블에 hash값과 경로를 관리

개발 환경 및 시스템 개발(8/11)

블랙리스트 관리



블랙리스트 등록시 관리서버 및 하위서버 iptables에서 IP 관리

개발 환경 및 시스템 개발(9/11)

위·변조 감시(1/2)

감시 스크립트

```
#!/bin/bash

case "$1" in

        start)
        psch=`ps -ef | grep $0 | grep -c start`

        if [ $psch -gt "2" ]
        then
                echo "already working"
        exit 1
        fi


        if [ -f /root/md5test.txt ]
        then
                echo "already copied"
        else
        find /etc/ -type f -printf '"%p"\n'|xargs md5sum -b > ./md5test.txt
        chmod 400 ./md5test.txt
        fi

while true
        do
                mdch1="./md5test.txt"
                mdch2="./md5ori.txt"
                if [ -z "`diff $mdch1 $mdch2`"]
                then
                        echo "변 경 된  값 이  없 습 니 다 ."
                else
                        echo "hash file changed"
```

실시간 감시 및 공격자 파일 교체시 관리자에게 알림

개발 환경 및 시스템 개발(10/11)

위·변조 감시(2/2)

감시 스크립트

```
status)
psch=`ps -ef |grep $0 |grep -c start`
if [ $psch = "1" ]
        then
        echo "working"
else
        echo "maybe stopped"
fi
;;


stop)
psch=`ps -ef |grep $0 |grep -c start`
if [ $psch = "1" ]
        then
        kill -9 `ps -ef | grep $0 | grep start | awk '{print $2}'`
else
        echo "already stopped"
fi
;;



add)
echo "enter the ip address"
read vicip
iptables -A INPUT -s $vicip -j DROP
;;


del)
read vicip
iptables -A INPUT -s $vicip -j ACCEPT
;;
```

의심 IP 차단, 관리자의 IP 허용, 차단한 IP 해제 등 조치

개발 환경 및 시스템 개발(11/11)

Telegram 알림

```
# message reply function
def get_message(bot, update) :
    update.message.reply_text("got text")
    update.message.reply_text(update.message.text)

def start_detection(bot,update):
    update.message.reply_text("g
    t = Thread(target=enqueue_out
    t.daemon = True
    t.start()
    update.message.reply_text(upd

def enqueue_output(update):
    global isRun
    isRun = True
    while isRun:
        print isRun
        p = subprocess.Popen('ex
```

```
def end_detection(bot,
    global isRun
    isRun= False
    update.message.repl
    update.message.repl

def start_record(bot,update):
    update.message.reply_text("Record the hash value.")
    p1 = subprocess.Popen('exec ./record.sh', shell=True, stdout=sub
    update.message.reply_text(update.message.text)

def start_check(bot,update):
    te.message.reply_text("check")
     subprocess.Popen('exec ./check.sh', shell=True, stdout=subp
    1 = p2.stdout.read()
    ext1 != "":
    update.message.reply_text(text1)
```

Telegram 통신 프로그램

❖Telegram을 이용, 관리자 알림

- 위·변조 상황 안내
- 외부에서 관리서버 접속알림

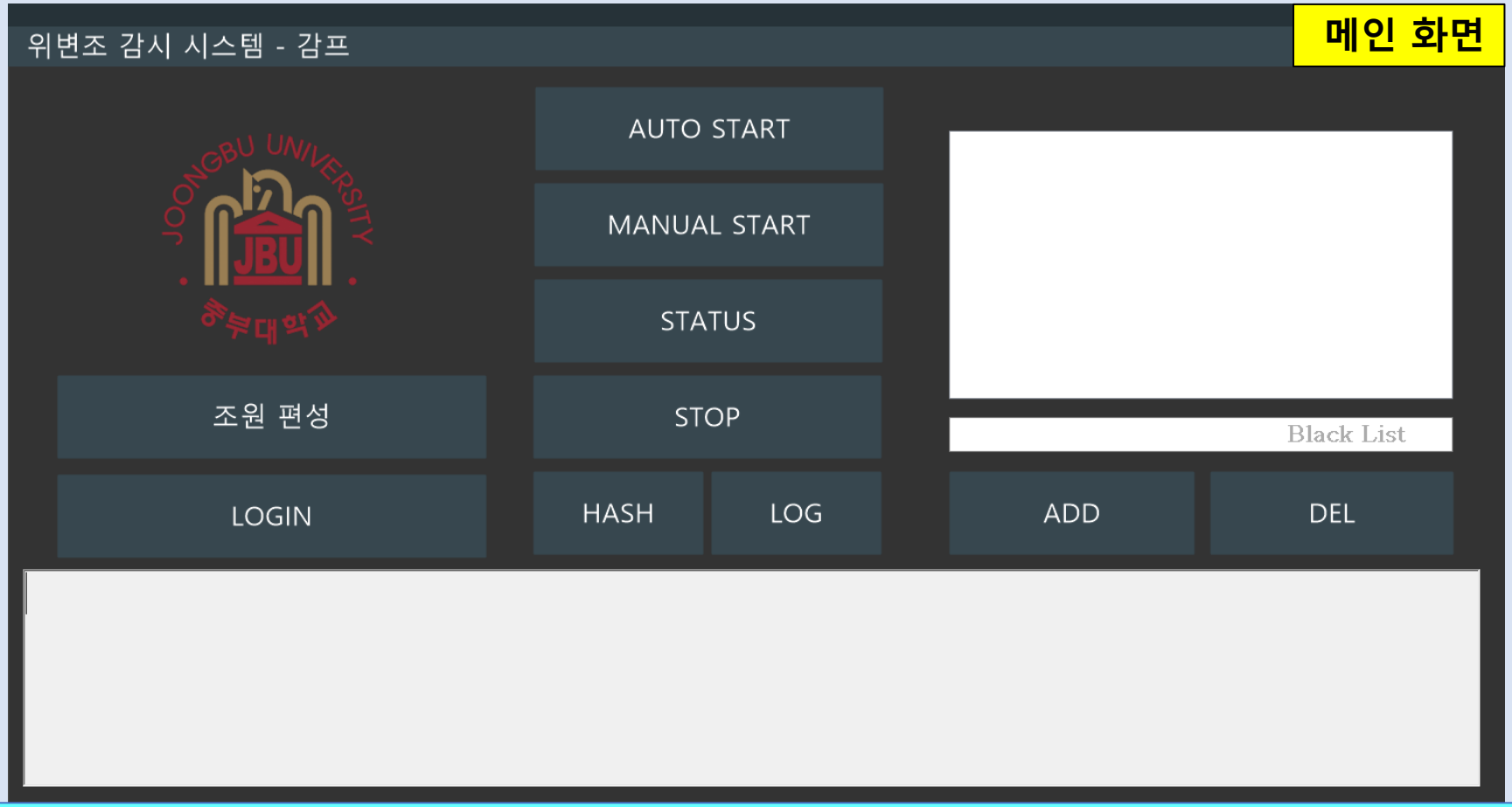
※ IP, 접속시간, 국가, 계정 등

❖Telegram을 이용, 명령 전달

- 관리서버 → 하위서버

개발 시스템 운영(1/7)

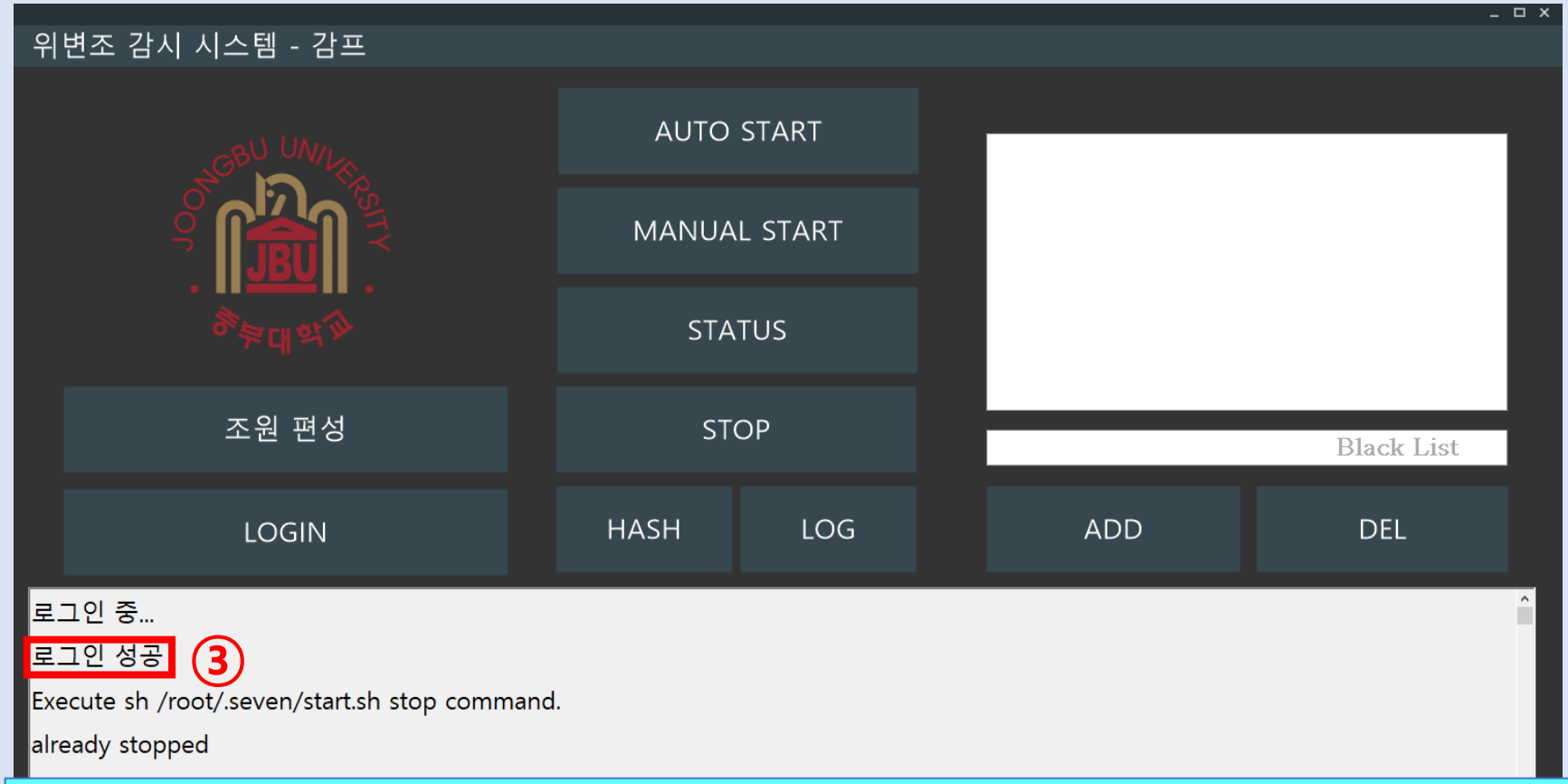
시스템 기동



로그인, 시작/정지, HASH 값 출력, 블랙리스트 추가/삭제 등 기능 구현

개발 시스템 운영(2/7)

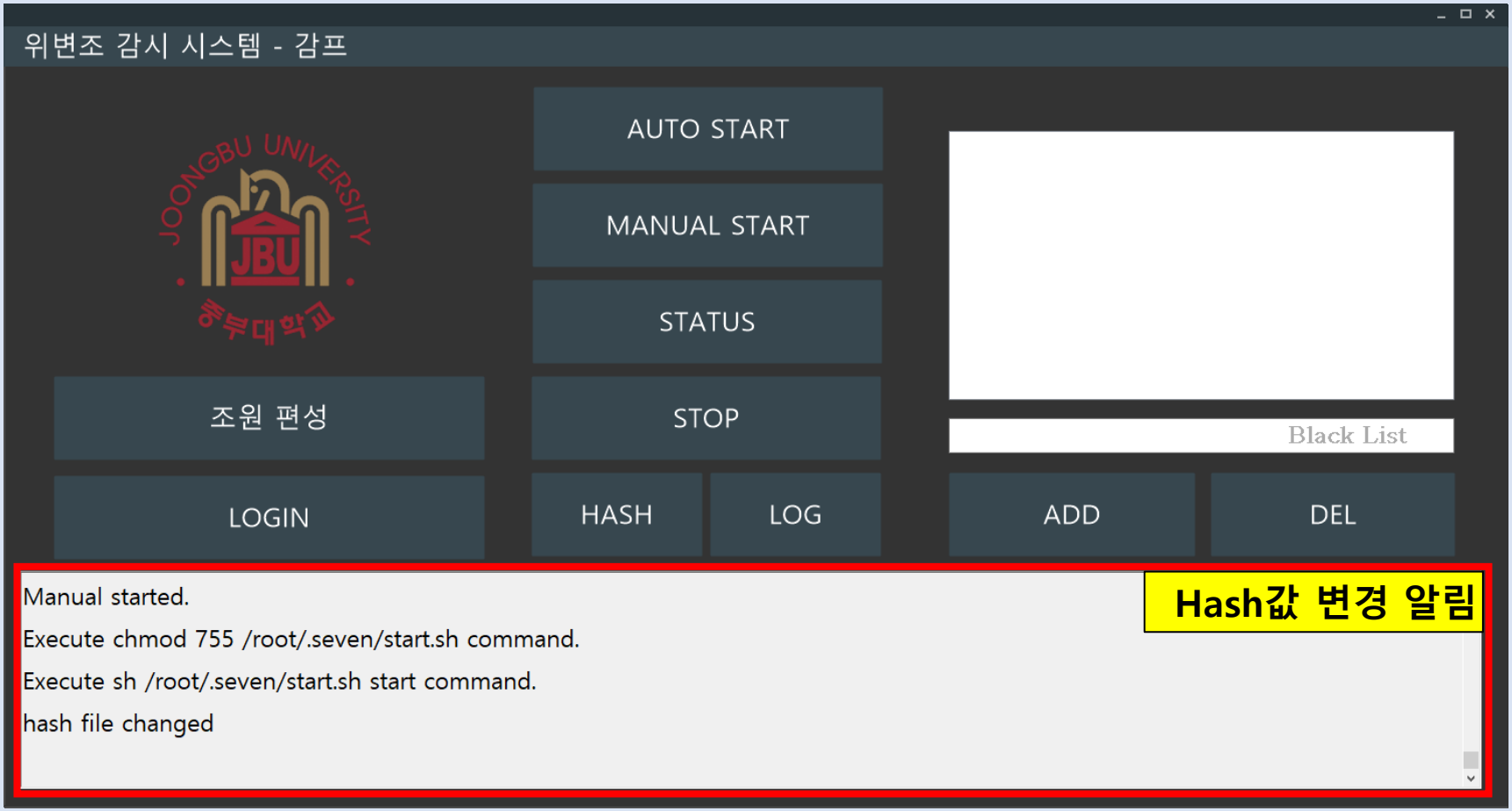
관리서버 로그인



관리서버의 ID(유저명@IP주소), 패스워드(유저PW) 입력 시 정상적으로 로그인

개발 시스템 운영(3/7)

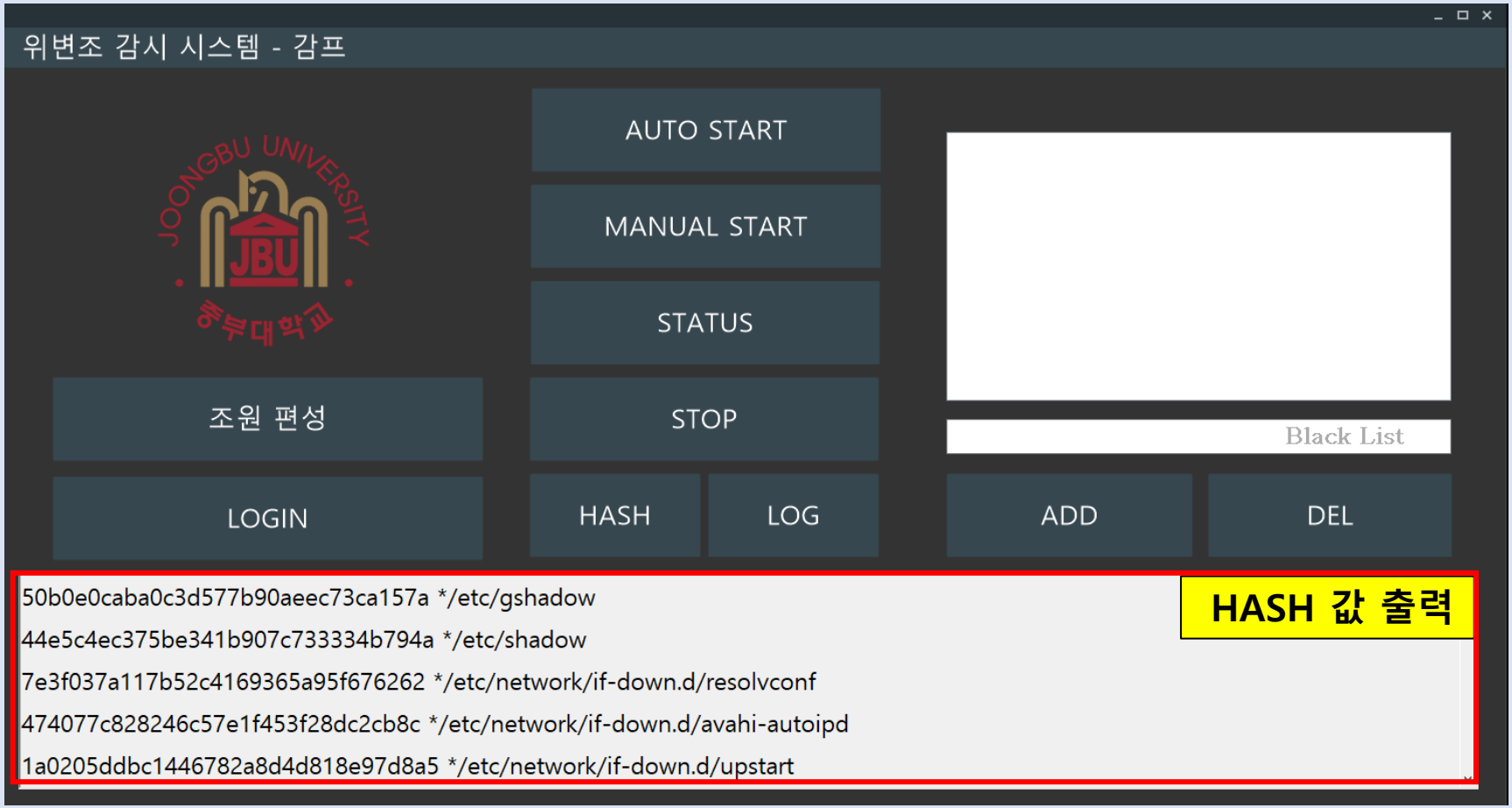
감시 시스템 동작



MANUAL START시 1회 동작

개발 시스템 운영(4/7)

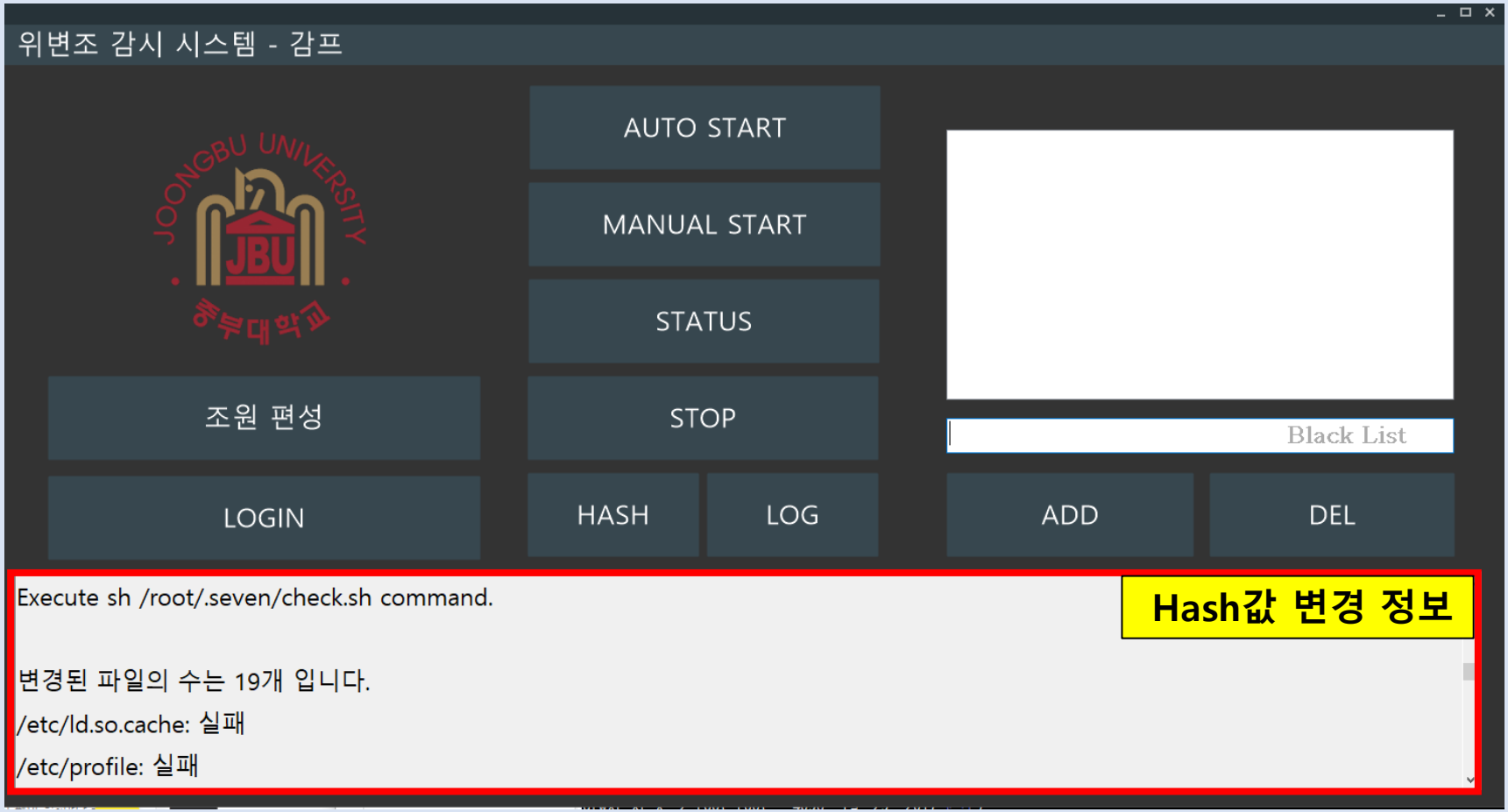
HASH 값 관리



HASH 값 데이터를 로드시켜 출력

개발 시스템 운영(5/7)

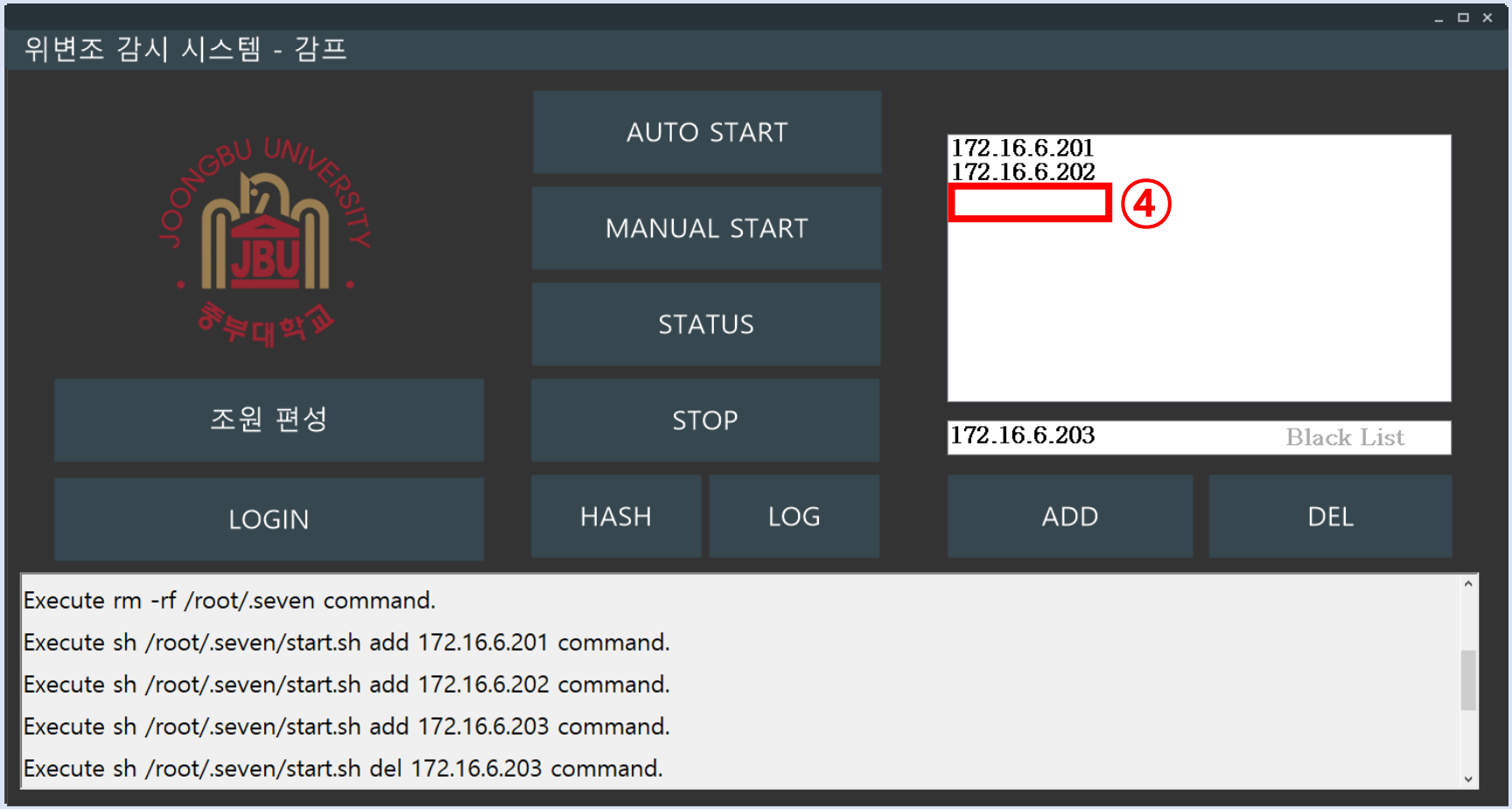
로그 확인



변경된 HASH 값 정보를 확인

개발 시스템 운영(6/7)

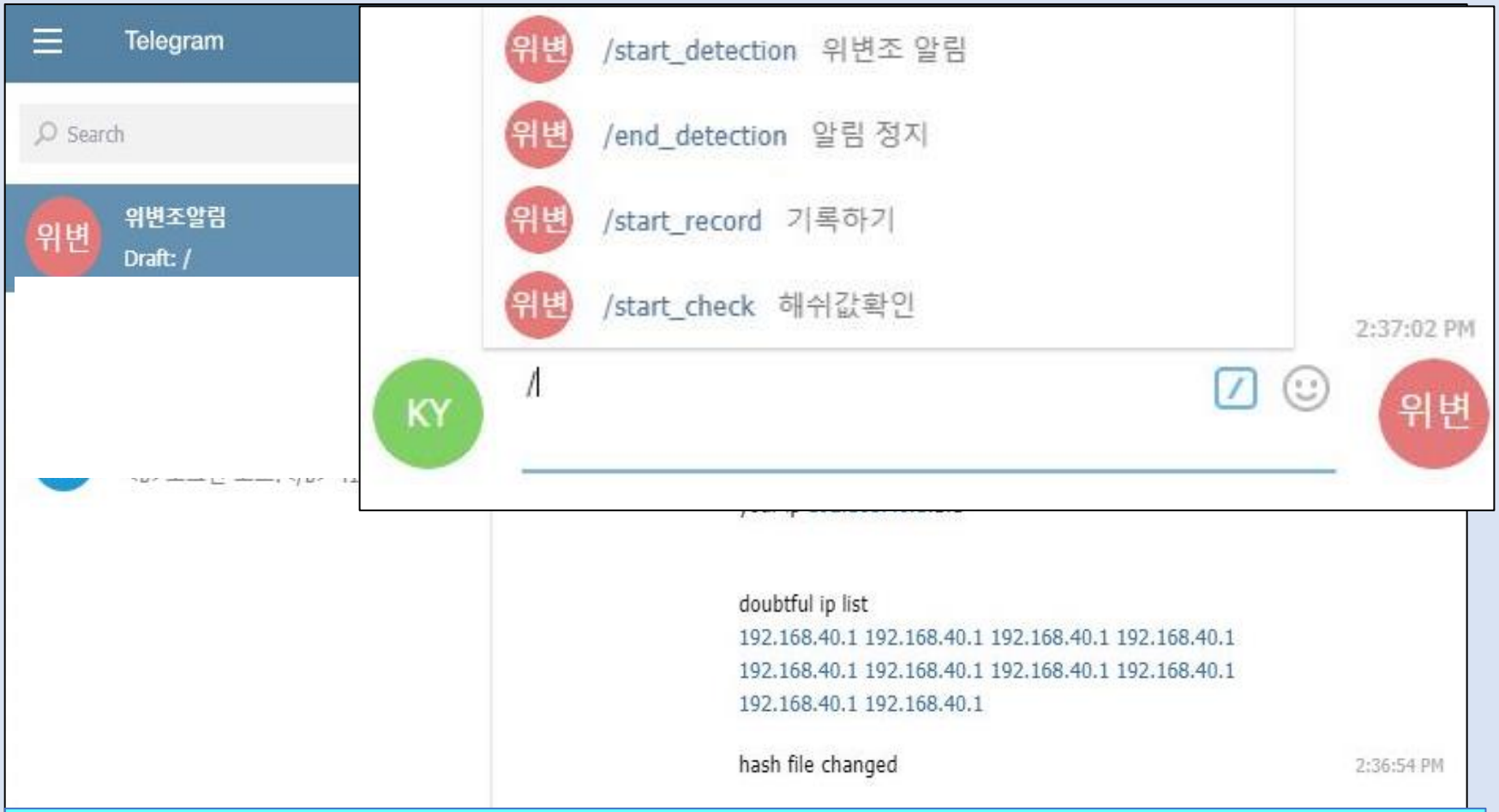
블랙리스트 관리



공격자로 의심되는 IP를 관리자가 추가 및 삭제 가능

개발 시스템 운영(7/7)

Telegram 알림



텔레그램을 활용하여 실시간 정보 알림 및 명령 전달 가능

결론 및 기대효과

❖ 위/변조 감시 시스템 개발 성과

- 개발 시스템은 외부 공격자의 위/변조 공격을 감시, 관리자에게 실시간 알리는 등 즉각적인 대응이 가능하게 하는 보안관리 환경을 제공

- 또한 감시 자료를 활용하여 취약점 분석에 활용하게 하는 등 관련 기능을 부가하여 시스템의 기능을 확대

❖ 기대 효과

- 시스템을 GUI 방식으로 구현하여 비전문가도 쉽게 활용할 수 있을 것으로 기대

- 모든 조원들이 임무를 분담하여 맡은 부분의 시스템을 책임성 있게 구현, 실무 능력 향상시키는 계기

Q & A

감 사 합 니 다