

트래픽 변화량을 이용한 DDoS 공격 탐지 프로그램

가라DoS

20191893 김강식

20191911 유승준

20191918 임동건



국립

한밭대학교

HANBAT NATIONAL UNIVERSITY

목차

1

팀 소개

2

개요

3

시스템 구성

4

주요기능

5

추진 일정

6

시연 영상

팀 소개



가라DoS

김강식

- 공격자 가상환경 구축 (VMware, Kali Linux)
- DDoS 공격 프로그램 제어 (XERO SPLOIT, Hping3)
- Locust 프로그램을 통한 웹서버 부하 테스트 진행

유승준

- 희생자 환경에서 Flask기반 웹서버 구축 및 공유기 포트 포워딩
- 패킷 분석 결과를 바탕으로 감지 알고리즘 작성
- 공격 감지 시 알림 기능 구현

임동건

- 웹서버 부하 테스트 결과에 따른 트래픽 임계치 설정
- 웹서버 봇 필터링을 위한 구글 reCAPTCHA API 적용
- Wireshark로 공격 유형에 따른 패킷 분석 후 특징 추출

공통

- DDoS 공격 및 감지 시나리오 작성, 플로우 차트 설계
- 화이트박스 / 블랙박스 테스트 수행
- 코드 리팩토링 및 디버깅

개요

DDoS 공격 유형 분석

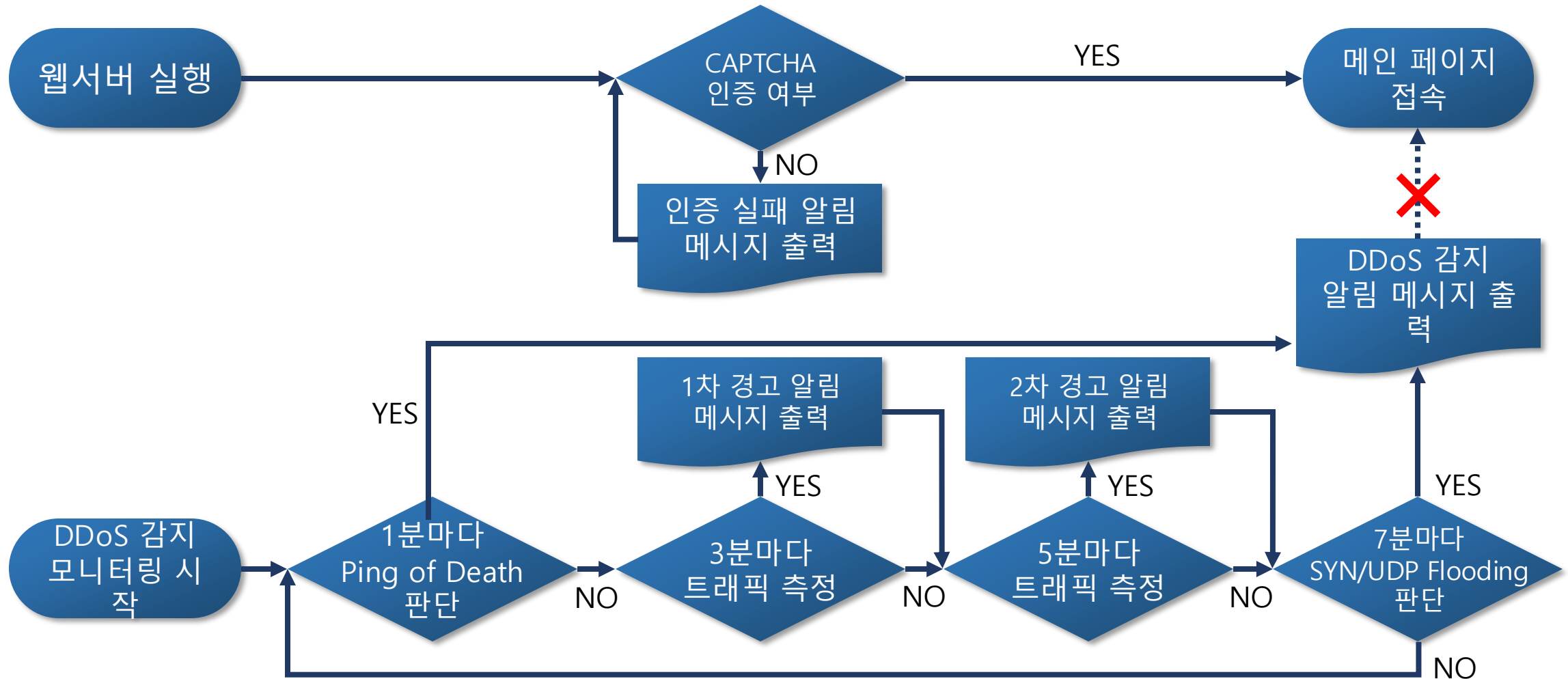
공격 트래픽 감지 및 판단 알고리즘 설계

DDoS 공격을 신속히 탐지하여 빠른 방어 대책 마련

시스템 구성

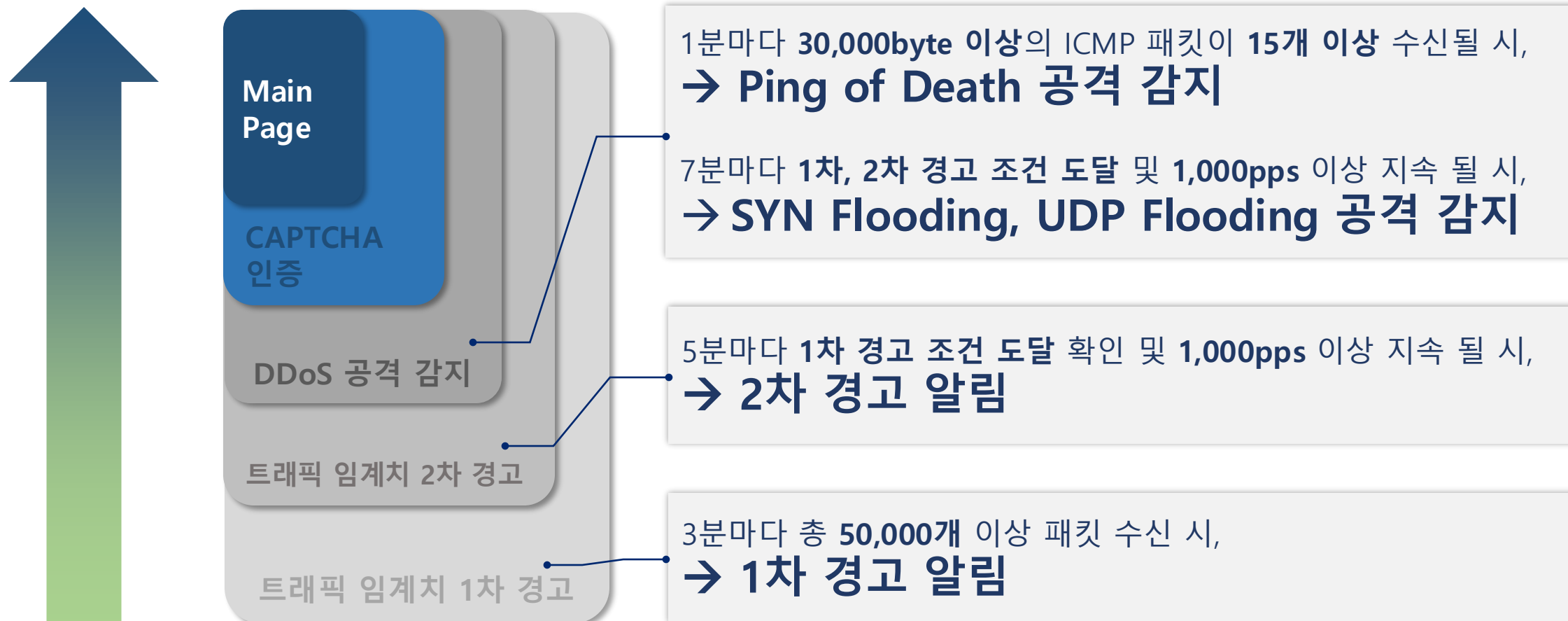


시스템 구동



주요 기능 : 공격 감지 알고리즘 설정

DDoS 공격 트래픽 임계치 설정



주요 기능 : 주요 공격 패킷 별 감지 기능

Flags: 0x002 (SYN)

```
000. .... = Reserved: Not set
...0 .... = Accurate ECN: Not set
.... 0... = Congestion Window Reduced: Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...0 = Acknowledgment: Not set
..... 0.. = Push: Not set
..... ..0. = Reset: Not set
```

..... ..1. = Syn: Set

▼ [Expert Info (Chat/Sequence): Connection establish request (SYN): server port 21]

SYN Flooding

```
"(
    tcp.flags.syn == 1 &&
    tcp.flags.ack == 0 &&
    tcp.analysis.retransmission &&
    tcp.dstport == 21
)"
```

▼ User Datagram Protocol, Src Port: 395

Source Port: 39590

Destination Port: 1234

Length: 1008

Checksum: 0x6ef0 [unverified]

[Checksum Status: Unverified]

[Stream index: 7423]

[Timestamps]

UDP payload (1000 bytes)

▼ Data (1000 bytes)

```
Data [truncated]: 585858585858585858585858585858585858585858
```

[Length: 1000]

UDP Flooding

```
"(
    udp &&
    frame.len >= 1000
)"
```

- Internet Control Message Protocol

Type: 8 (Echo (ping) request)

~~Code: 0~~

Checksum: 0xa2ee [correct]

[Checksum Status: Good]

Identifier (BE): 56216 (0xdb98)

Identifier (LE): 39131 (0x98db)

Sequence Number (BE): 256 (0x0100)

Sequence Number (IE): 1 (0x0001)

Sequence Number (2)
[No response seen]

▼ [Expert Info (Warning/Sequence): No response seen to ICMP request]

```
[No response seen to ICMP request]
```

[Severity level: Warning]

[Group: Sequence]

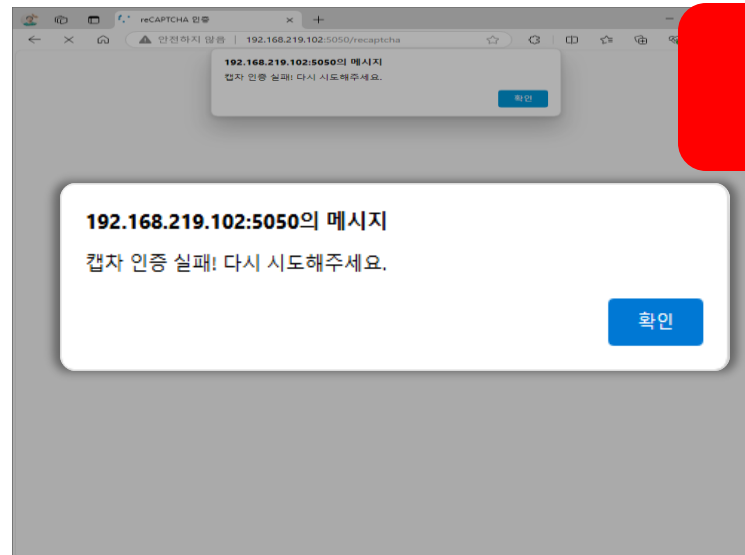
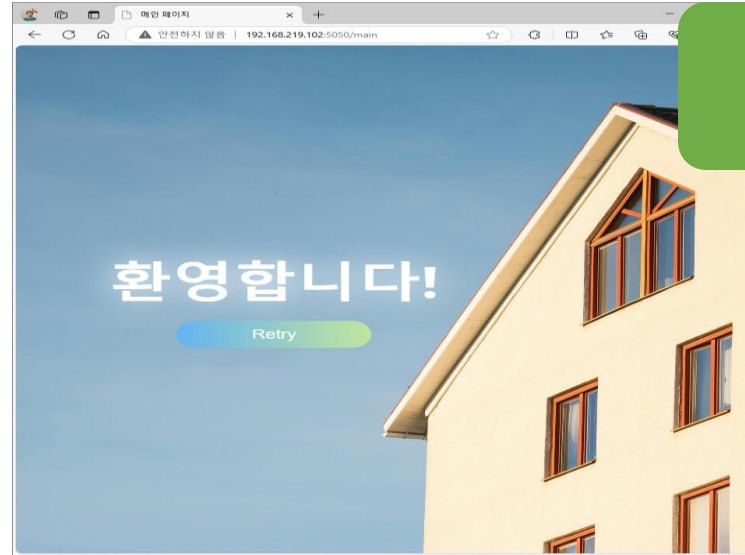
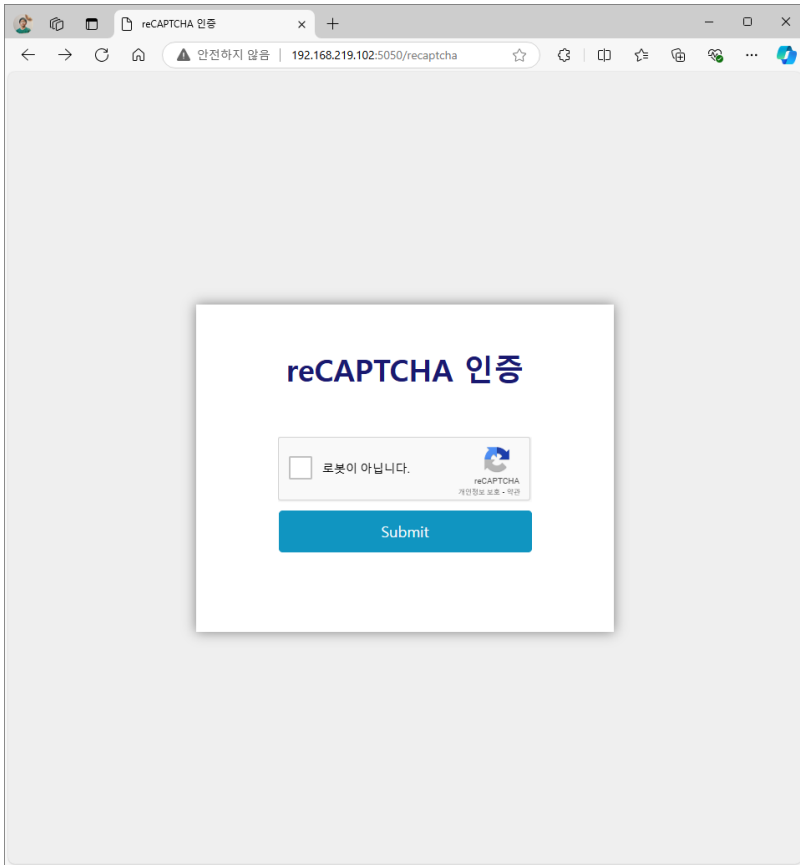
```
> Data (30000 bytes)
```

Ping of Death

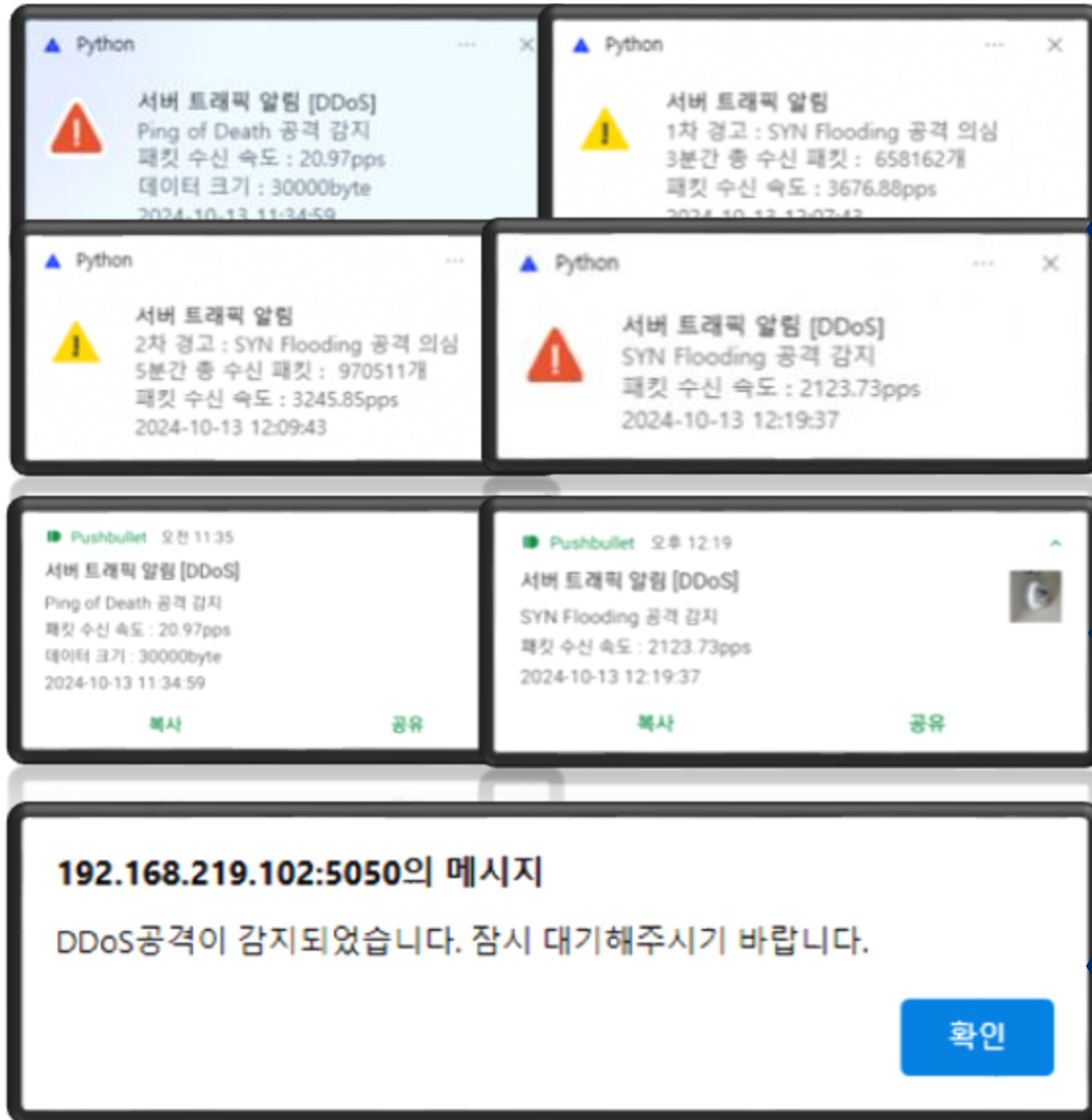
```
"(
    icmp &&
    icmp.type == 8 &&
    data.len >= 30000
)"
```

캡처 필터 조건

주요 기능 : Recaptcha 봇 감지 절차



주요 기능 : 모바일/PC 공격 감지 알림



공격 감지 여부 확인



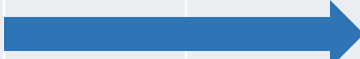
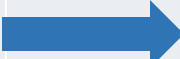
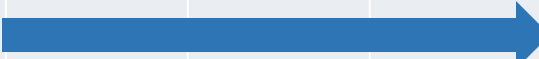

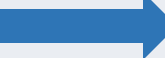
- 설정한 트래픽 임계치에 따라 알림 출력
- 알림 내용
 - 공격유형
 - 특정 시간마다 집계된 패킷 수
 - 감지 일시
 - 데이터 크기

서버 실행 컴퓨터 알림

등록된 스마트폰 PUSH 알림

웹서버 페이지 브라우저 알림

추진 일정

연구내용	담당	3월	4월	5월	6월	7월	8월	9월	10월
DDoS공격 유형 조사, 감지 시나리오 작성	공통								
Python 네트워크 라이브러리, Wireshark 사용법 학습	공통								
DDoS공격/감지 실습 환경 구축	김강식 유승준								
DDoS공격 유형에 따른 패킷 분석	김강식 임동건								
분석 결과를 바탕으로 코드 작성	유승준 임동건								
시나리오에 따른 테스트 수행, 프로그램 수정/보완	공통								
프로젝트 결과 보고서 작성	공통								

Github 링크 : https://github.com/tmdwns29/DDoS_Detection

1. 웹서버 실행 및 reCAPTCHA 인증 과정