


가상 네트워크(SDN/NFV) 보안과 안전한 인프라 구축을 위한 교육 (v2.0)



엔터프라이즈 시스템/네트워크 운영자 대상
(for IT Pros and System Administrators)

KOREN e2e 서비스 활용을 위한 실습 구성

2018년 11월

목차



1. 실습 환경 준비
2. vUTM 설치 (pfSense..)
3. Host 설치 (Linux..)
4. vLAN 구성 (e2e..)
5. vFW (pfSense..)
6. vIPS (snort)
7. 웹필터 (SquidGuard)
8. ntopng (플로우 모니터)
9. vIDS (Security Onion)

목차

- 
- 1. 실습 환경 준비**
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 6. vIPS (snort)
 7. 웹필터 (SquidGuard)
 8. ntopng (플로우 모니터)
 9. vIDS (Security Onion)

1. 실습 환경 준비

❖ 사용 가능 소프트웨어

① Linux OS (Bare metal 설치 Lab 환경 구성 고려)

- Fedora 또는 CentOS
- Ubuntu 또는 Debian
- Open Network Linux (<https://opennetlinux.org/>)
- 기타

② Hardware 고려

- Intel 기반
- ARM 기반

③ 하이퍼바이저 기반 가상 네트워크 소프트웨어

- 가상화 보안 어플라이언스 (방화벽, IDS, SIEM등)
- 가상화 네트워크 어플라이언스 (라우터, SDN 제어기등)

OS	Packaging Tools	기타
Ubuntu	debian packaging (* .deb → apt-get install)	Debian
Fedora	redhat packaging (.rpm → yum(dnf) install)	RHEL, CentOS
Open Network Linux	nos-install-image (onie install)	Accton(7), Agema(1), Alpha Network(2), Dell(2), Penguin(3), Quanta(3)

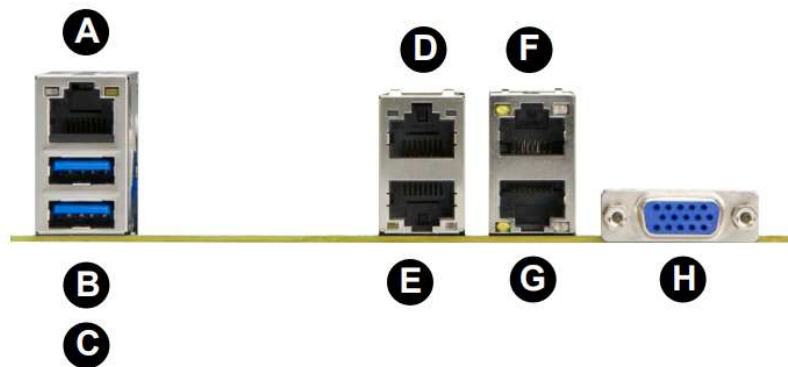
메모:

- Current ONIE Hardware Status:
http://www.opencompute.org/wiki/Networking/ONIE/HW_Status

1. 실습 환경 준비

❖ 하드웨어

- ① **CPU w/Passive CPU heat sink**
 - Intel® Xeon® processor D-1528
 - FCBGA 1667
 - CPU TDP support 35W, 9MB, 6 Cores, 12 Threads, 1.9-2.2GHz
- ② **RAM**
- ③ **SSD**
- ④ **IPMI 2.0**
- ⑤ **10GbE 2포트, 1 GbE LAN 2포트, IPMI 2.0 전용 LAN**
- ⑥ **SR-IOV (Single-Root Virtualization)**



Back Panel I/O			
A	IPMI LAN	E	LAN Port 1 (-F, -LN2F, -TLN4F)
B	USB Port 1	F	LAN Port 4 (-TLN2F and -TLN4F)
C	USB Port 0	G	LAN Port 3 (-TLN2F and -TLN4F)
D	LAN Port 2 (-F, -LN2F, -TLN4F)	H	VGA Port

메모:

- Low noise fan speed control

1. 실습 환경 준비

❖ 하이퍼바이저 설치 @ KOREN AI Network Lab

- ① Initial Powering Up (w/o Internet)
- ② USB booting Available
- ③ Alt-Ctrl-D로 Rebooting 하여 install 가능
- ④ Rebooting 시 'F11'에서 USB Booting 선택 (SanDisk)
- ⑤ ESXi '6.0' vs '6.5' (실습 진행 편리를 위한 선택)
- ⑥ Windows Server 2016 Hyper-v 고려
- ⑦ 개인용 노트북 사용 (PDF viewer, Putty, WEB browser, Software Tools)



메모:

- ESXi 다운로드 주소: <https://my.vmware.com/en/web/vmware/evalcenter?p=free-esxi6>
- 디스크 이미지 굽기: Rufus 도구 사용 <https://rufus.akeo.ie/>
- Disk Imager <https://sourceforge.net/projects/win32diskimager/files/latest/download>
- USB 부팅 제가동은 전원 off/on (전원 케이블 포함)필요함

1. 실습 환경 준비

❖ 실습 구성 @ KOREN AI Network Lab

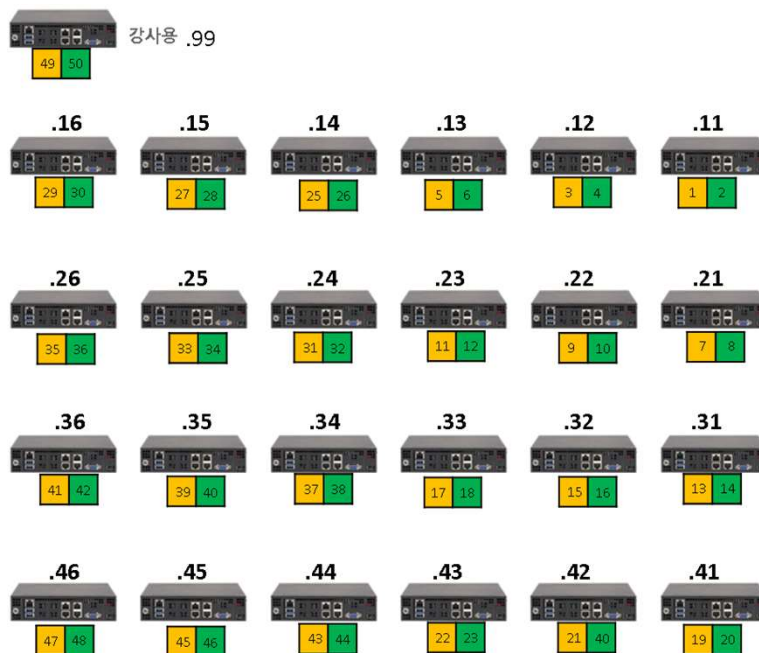
① USB 메모리

- OS
- 소프트웨어 도구 (Software Tools)

② IPMI 연결 이더넷 케이블

③ 인터넷 연결 케이블

④ 좌석 번호 별 서버넷의 해당 IP주소(x.x.x.nn) 설정 사용



메모:

- 하이퍼바이저 설치 환경은 구조 분석을 고려하여 도구를 접속하여 미러링 가능한 네트워크 구성이 가능해야 함.
- 가상스위치는 분석을 위한 무작위(Promiscuous) 모드 설정 고려
- IPMI: Intelligent Platform Management Interface

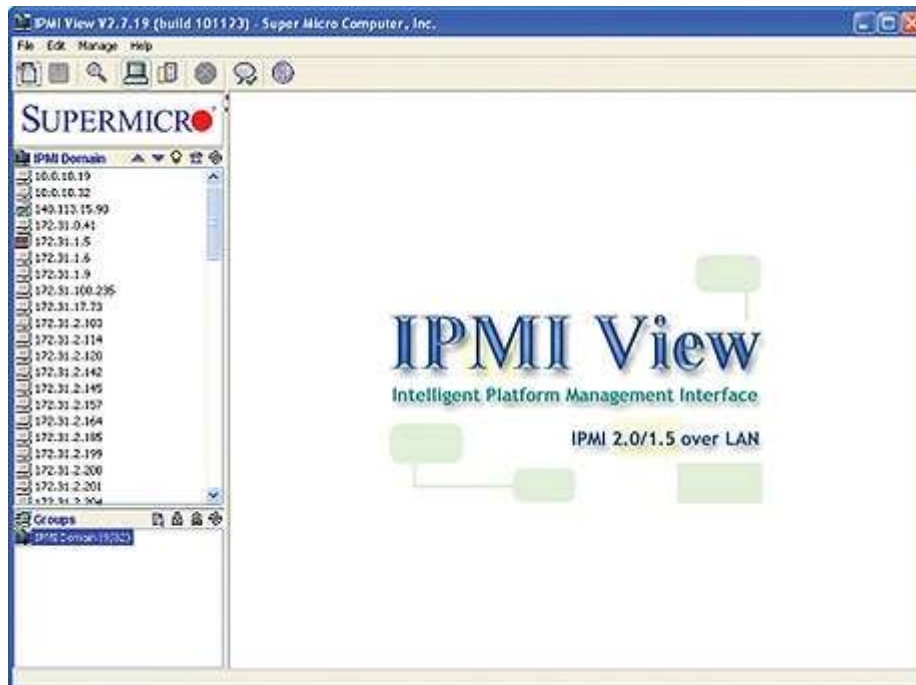
1. 실습 환경 준비

❖ IPMI

① 개요

- Supermicro Intelligent Management
- The Supermicro X11 platform's Baseboard Management Controller (BMC) is built on the ASPEED AST 2500 controller.

② 시연



메모:

- 슈퍼마이크로 보드 기준 IPMI는 기본적인 IPMI 통신 포트인 623
- 웹으로 구현된 관리 인터페이스인 80(HTTP), 443(HTTPS)로 사용
- 원격 조종에는 5900(KVM) 포트도 사용
- 최초 접속시 Username/Password는 ADMIN/ADMIN 입니다.

1. 실습 환경 준비

❖ 하이퍼바이저 비교

- ① Microsoft의 Hyper-v는 평가기간 무제한
- ② vSphere 6.5 평가판은 60일간 모든 기능 제공하며, 평가 기간 종료 후에 상용기능 정지
- ③ 하이퍼바이저 사용 실습에서는 LAN/웹브라우저/PDF뷰어 지원 개인 노트북 지참
- ④ 실습은 ESXi 6.7 사용 (전용 클라이언트 없이 웹으로 접속)

제품 기능	Microsoft	VMware vSphere 6.5		
	Hyper-V 2016	Free Hypervisor	Essential Plus	Enterprise Plus
VM 호스트 라이브 마이그레이션	Yes	No	Yes	Yes
VM 스토리지 라이브 마이그레이션	Yes	No	No	Yes
스토리지/네트워크 QoS	Yes	No (just disk shares)	No (just disk shares at host level)	Yes
하드웨어 패스드루	Discrete Device Assignment	PCI VM Direct Path USB redirection	PCI VM Direct Path USB redirection	PCI VM Direct Path USB redirection
운영 중 추가	Disks/vNIC/RAM	Disks/vNIC/USB	Disks/vNIC/USB	Disks/vNIC/USB/ CPU/RAM
운영 중 제거	Disks/vNIC/RAM	Disks/vNIC/USB	Disks/vNIC/USB	Disks/vNIC/USB/CP U
디스크 사이즈 조정	Hot-grow and shrink	Hot-grow	Hot-grow	Hot-grow
VM 암호화	Yes	No	No?	Yes

메모:

- 여러 명이 개요 수준 실습을 진행하며 웹브라우저가 동일하지 않은 경우 vSphere 6.0과 전용 클라이언트 소프트웨어 사용 권장
- Type 2 Hypervisor는 VMware (WorkStation) Player 또는 VirtualBox 사용 가능 노트북 미지참 실습은 베어메탈 서버에 리눅스 설치 (USB 허브 필요)

1. 실습 환경 준비

❖ Hypervisor Installation

- ① Initial Powering Up (w/o Internet)
- ② USB booting Available
- ③ Alt-Ctrl-D로 Rebooting 하여 install 가능
- ④ Rebooting 시 'F11'에서 USB Booting 선택
- ⑤ ESXi '6.0' vs '6.5' vs '6.7' (6.7 설치 시연)
- ⑥ Windows Server 2016 Hyper-v

```
Please select boot device:

IBA GE Slot 0500 v1513
UEFI: Built-in EFI Shell
P0: TOSHIBA Q300 Pro.
SanDisk
UEFI: SanDisk
Enter Setup

↑ and ↓ to move selection
ENTER to select boot device
ESC to boot using defaults
```

**** 실습 교육 진행은 OS나 웹브라우저 종류별로 다를 수 있는 동작을 고려한 안정적 버전과 도구를 선택하여 진행 ****

메모:

- Windows Containers on Windows Server: <https://docs.microsoft.com/en-us/virtualization/windowscontainers/quick-start/quick-start-windows-server>
- USB 부팅 재가동은 전원 off/on (전원 케이블 포함)필요함

1. 실습 환경 준비

❖ Hypervisor Installation (ESXi 6.0 예)

①



ESXi-6.0.0-20170604001-standard Boot Menu
ESXi-6.0.0-20170604001-standard installer
Boot from local disk
Press [Tab] to edit

②



VMware ESXi 6.0.0 (VMkernel Release Build 5572656)
VMware, Inc. VMware Virtual Platform
4 x Intel(R) Xeon(R) CPU @ 3.50GHz 8 GiB Memory
vmx1: client loaded successfully.

③



Welcome to the VMware ESXi 6.0.0 Installation
VMware ESXi 6.0.0 installs on most systems but only systems on VMware's Compatibility Guide are supported.
Consult the VMware Compatibility Guide at:
<http://www.vmware.com/resources/compatibility>
Select the operation to perform.
(Esc) Cancel (Enter) Continue

④



End User License Agreement (EULA)
VMWARE END USER LICENSE AGREEMENT
PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.
IMPORTANT-READ CAREFULLY: BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE, YOU (THE INDIVIDUAL OR LEGAL ENTITY) AGREE TO BE BOUND BY THE TERMS OF THIS END USER LICENSE AGREEMENT ("EULA"). IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA, YOU MUST NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND YOU MUST DELETE OR RETURN THE UNUSED SOFTWARE TO THE VENDOR FROM WHICH YOU ACQUIRED IT WITHIN THIRTY (30) DAYS AND REQUEST A REFUND OF THE LICENSE FEE, IF ANY, THAT
Use the arrow keys to scroll the EULA text
(ESC) Do not Accept (F11) Accept and Continue

⑤



Select a Disk to Install or Upgrade
* Contains a VMFS partition
Claimed by VMware Virtual SAN (VSAN)

Storage Device	Capacity
Local: VMware Virtual disk (mpx.vmhba1:CD:TO:LD)	40.00 GiB
Remote: (none)	

(Esc) Cancel (F1) Details (F5) Refresh (Enter) Continue

⑥



Enter a root password
Root password: *****
Confirm password: *****_
Passwords match.
(Esc) Cancel (F9) Back (Enter) Continue

⑦



Installation Complete
ESXi 6.0.0 has been successfully installed.
ESXi 6.0.0 will operate in evaluation mode for 60 days. To use ESXi 6.0.0 after the evaluation period, you must register for a VMware product license. To administer your server, use the vSphere Client or the Direct Control User Interface.
Remove the installation disc before rebooting.
Reboot the server to start using ESXi 6.0.0.
(Enter) Reboot

메모:

- ESXi 다운로드 주소: <https://my.vmware.com/en/web/vmware/evalcenter?p=free-esxi6>
- 디스크 이미지 굽기: Rufus 도구 사용 <https://rufus.akeo.ie/>
- Disk Imager <https://sourceforge.net/projects/win32diskimager/files/latest/download>
- ESXi 6.7: F11 → Enter → US Default → Root Password (jslab123!@#) → F11

1. 실습 환경 준비

❖ Hypervisor Installation

- ① **Configure Management Network 선택**
- ② **좌석 번호 'TT' 이용 고정 IP 주소 설정 - 192.168.1.TT**

The screenshot shows the 'System Customization' menu on the left and the 'Configure Management Network' configuration screen on the right. In the menu, 'Configure Management Network' is highlighted with a red dashed box and a circled '1'. In the configuration screen, the 'IPv4 Address' is set to '192.168.1.229', which is also highlighted with a red dashed box and a circled '2'. The configuration screen shows the following details:

```
Configure Management Network

Hostname:
localhost

IPv4 Address:
192.168.1.229

Network identity acquired from DHCP server 192.168.1.1

IPv6 Addresses:
fe80::20c:29ff:fee5:4f66/64

To view or modify this host's management network settings in
detail, press <Enter>.

<Up/Down> Select
<Enter> More
<Esc> Log Out
```

메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① Networking 선택 확인
- ② Network Adaptor 선택 확인

192.168.1.14 - vSphere Client

File Edit View Inventory Administration Plug-ins Help

Home Inventory Inventory

localhost.localdomain VMware ESXi, 6.0.0, 3620759 | Evaluation (59 days remaining)

Getting Started Summary Virtual Machines Resource Allocation Performance Configuration Users Events Permissions

View: vSphere Standard Switch

Networking Refresh Add Networking... Properties...

Standard Switch: vSwitch0 Remove... Properties...

Virtual Machine Port Group Physical Adapters

VM Network vmnic0 1000 Full

VMkernel Port

Management Network

vmk0 : 192.168.1.14
fa80::ae1f:6bff:fe1b:8c8e

Recent Tasks

Name localhost.localdomain VMware ESXi, 6.0.0, 3620759 | Evaluation (59 days remaining)

Getting Started Summary Virtual Machines Resource Allocation Performance Configuration Users Events Permissions

Hardware

Health Status Processors Memory Storage Networking Storage Adapters Network Adapters Advanced Settings Power Management

Software

Licensed Features Time Configuration DNS and Routing Authentication Services Virtual Machine Startup/Shutdown Virtual Machine Swapfile Location

Network Adapters

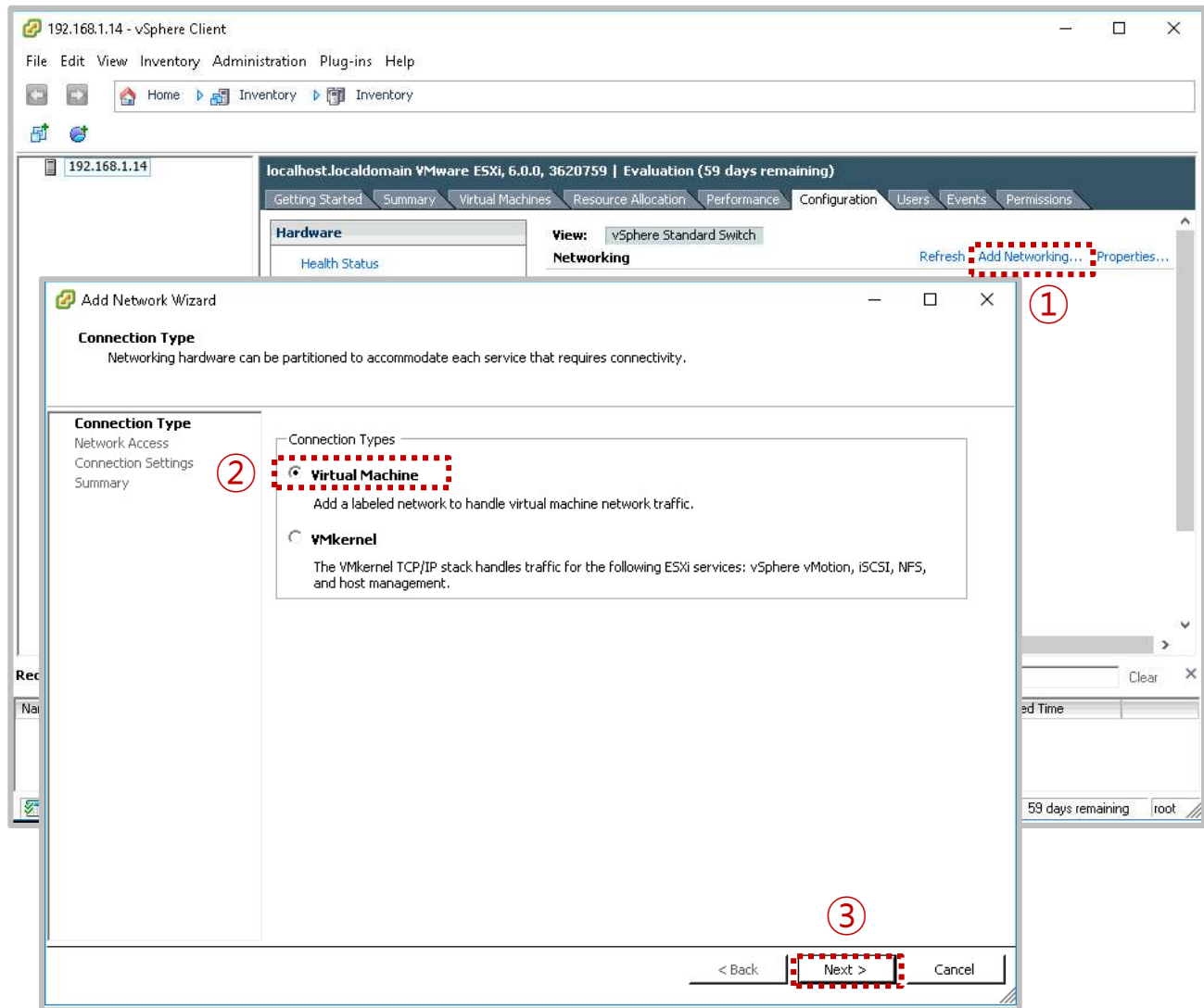
Device	Speed	Configured	Switch	MAC Address	Observed IP ranges
Intel Corporation I350 Gigabit Network Connection					
vmnic1	Down	Negotiate	None	ac:1f:6b:1b:8c:8f	None
vmnic0	1000 Full	Negotiate	vSwitch0	ac:1f:6b:1b:8c:8e	0.0.0.1-255.255.255

메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① Add Networking 선택
- ② Virtual Machine 선택

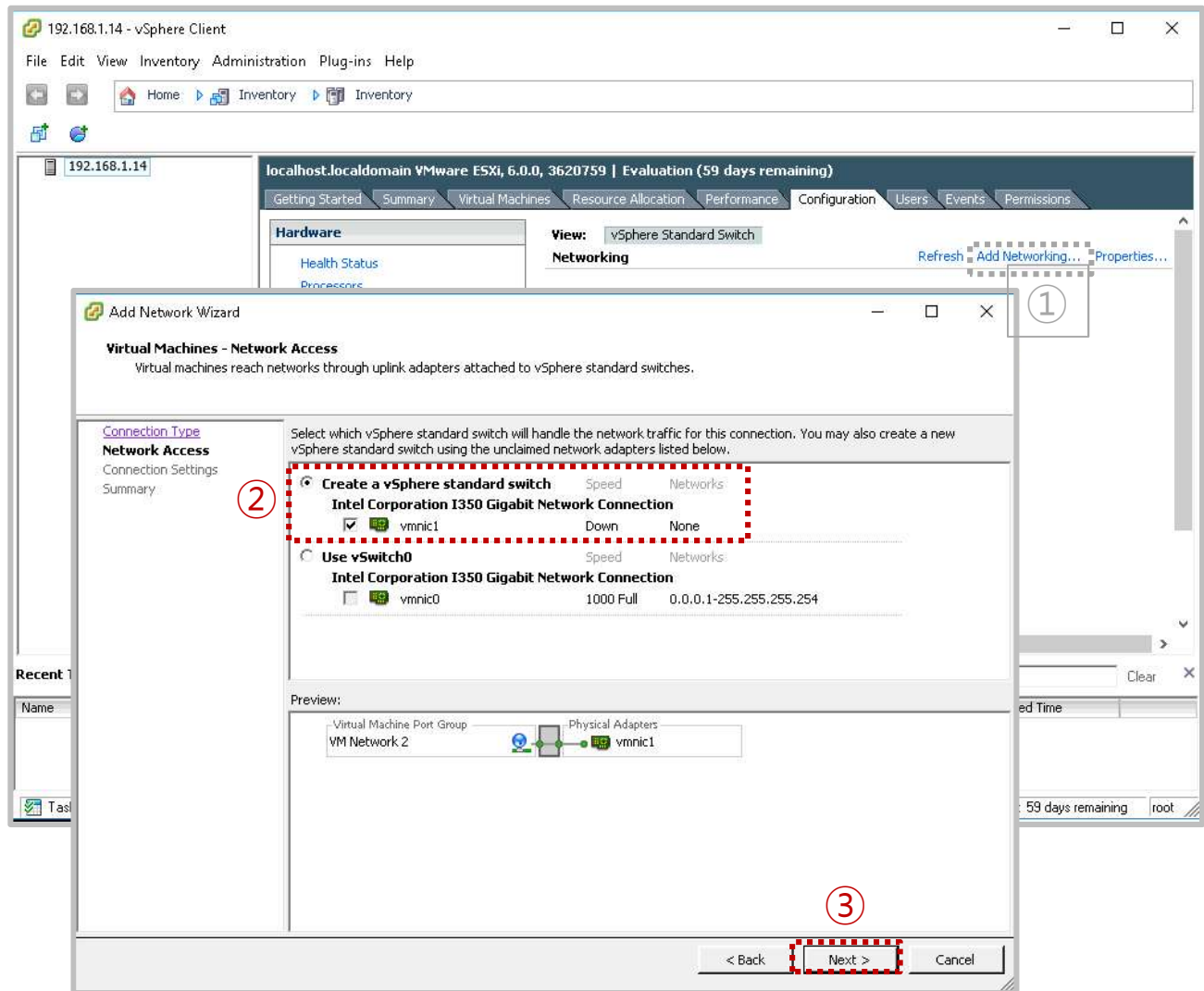


메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① 표준 스위치 생성 선택
- ② 유선랜 네트워크 연결 (내부 네트워크를 위한 선택)

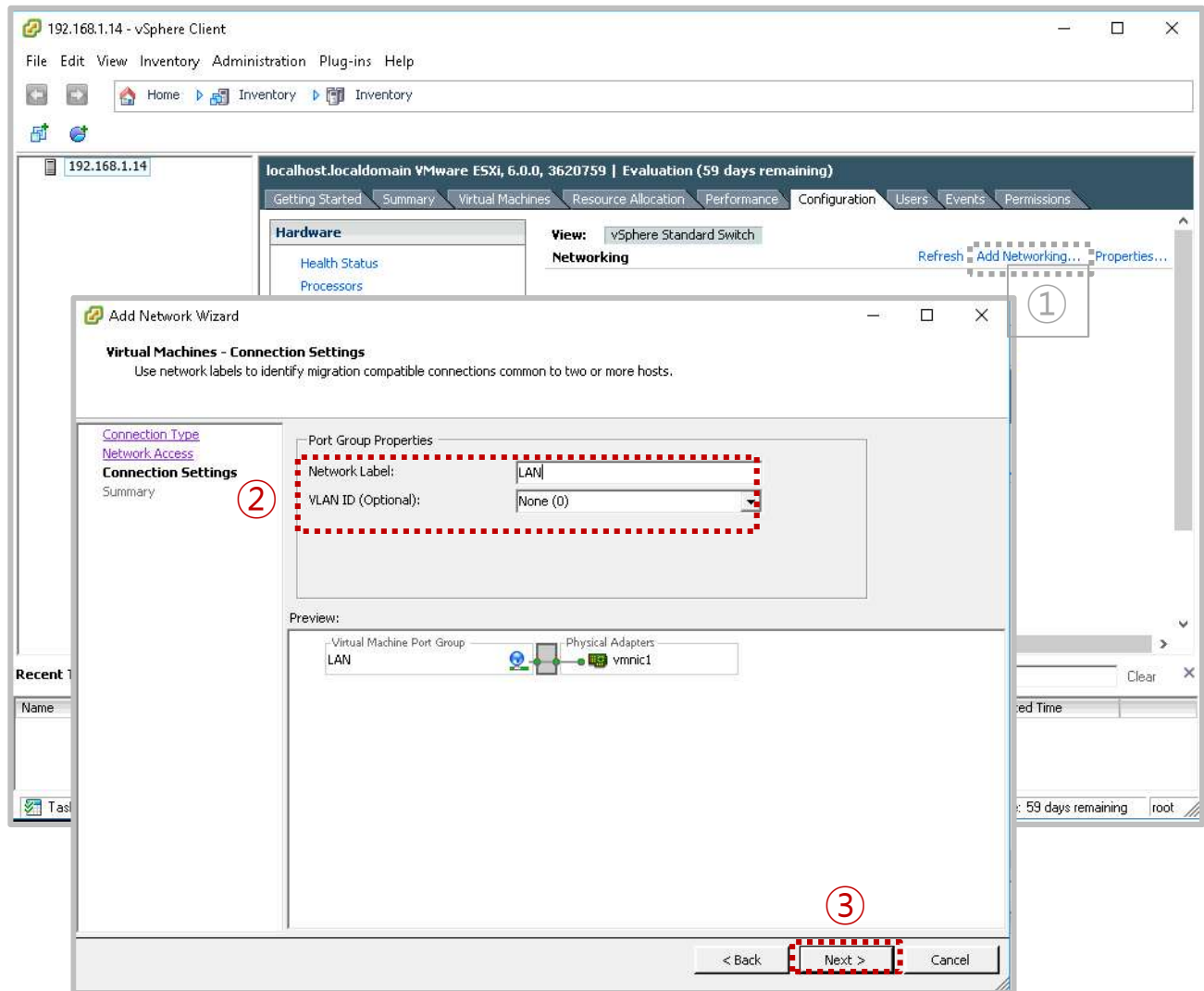


메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① 포트그룹 이름 설정
- ② 유선랜 네트워크 연결 (내부 네트워크를 위한 선택)

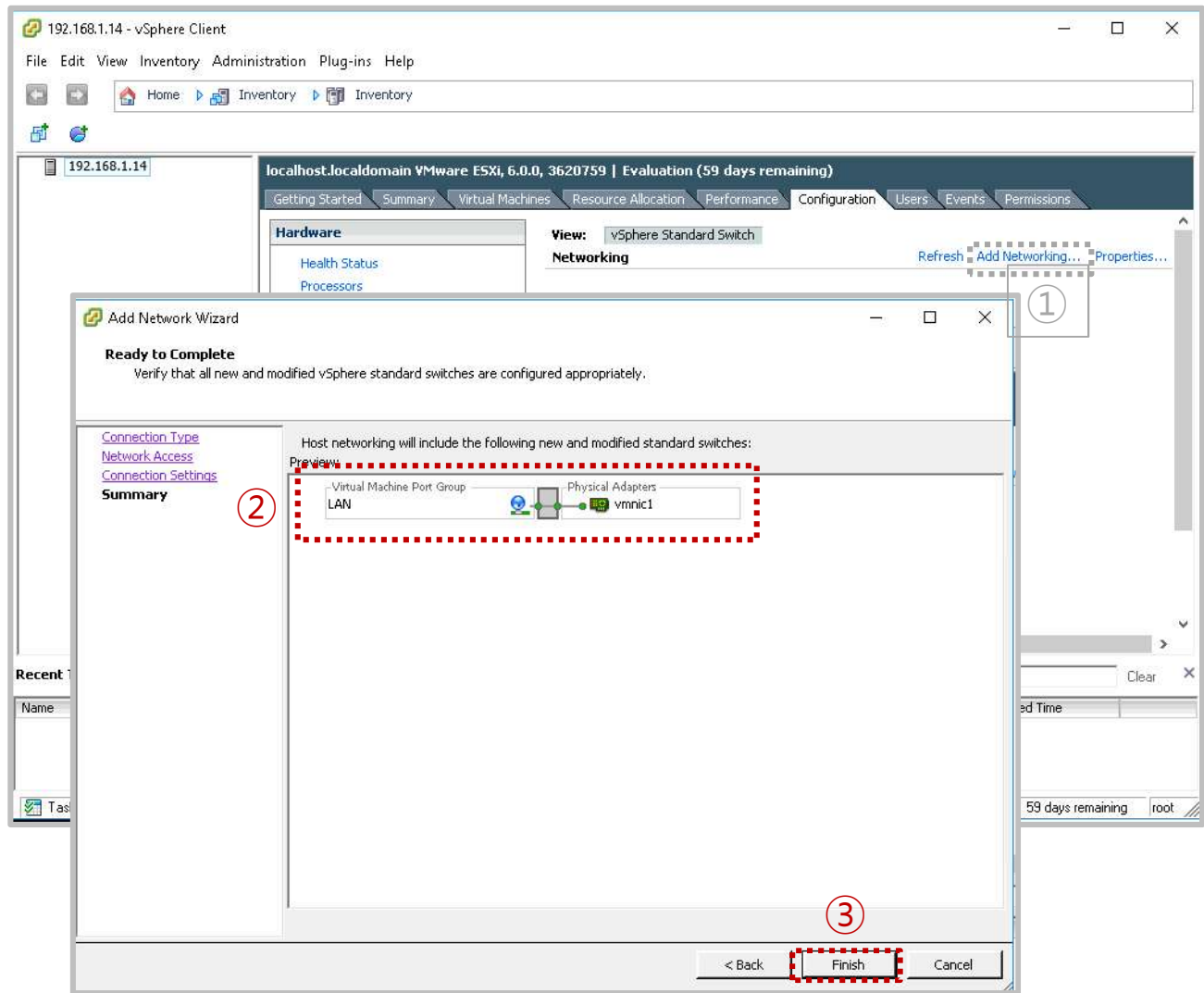


메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① 포트그룹 이름 설정
- ② 유선랜 네트워크 연결 (내부 네트워크를 위한 선택)

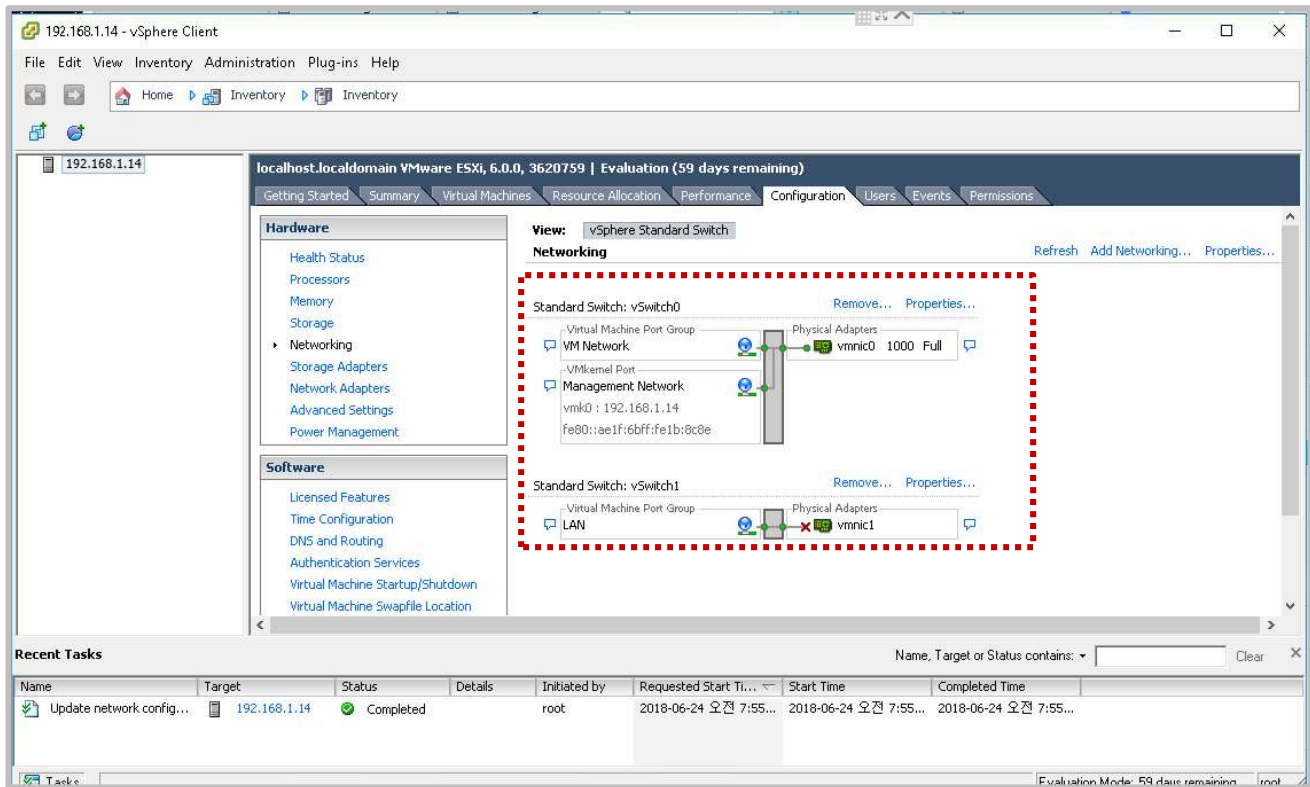


메모:

1. 실습 환경 준비

❖ Hypervisor Installation

- ① 포트그룹 이름 설정
- ② 유선랜 네트워크 연결 (내부 네트워크를 위한 선택)

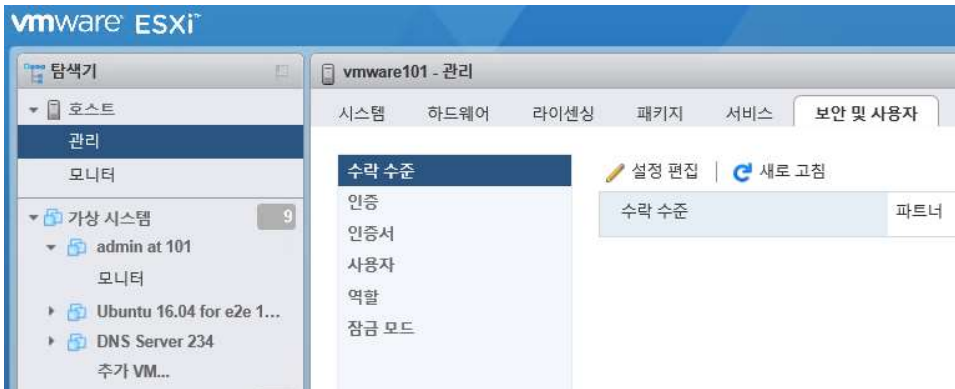


메모:

1. 실습 환경 준비


❖ Hypervisor 보안

- ① 관리 → 보안 및 사용자
- ② 서비스



메모:

목차

- 
1. 실습 환경 준비
 - 2. vUTM 설치 (pfSense..)**
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 6. vIPS (snort)
 7. 웹필터 (SquidGuard)
 8. ntopng (플로우 모니터)
 9. vIDS (Security Onion)

2. vUTM 설치

❖ vUTM 개요

- 최고의 보안 인프라 실습 환경 제공
- UTM은 기본적인 보안 시스템 내장
 - ✓ 방화벽
 - ✓ 침입탐지/차단 (IDS/IPS)
 - ✓ L2/L3 라우팅
 - ✓ 무선랜 보안
 - ✓ 가상사설망(VPN)
 - ✓ 웹필터링 (Web Filtering)
 - ✓ 안티바이러스
 - ✓ DLP (Data Loss Prevention)
- 실습은 오픈소스 사용 (pfSense 소호 레퍼런스)
 - ✓ 라우터 모드, 브릿지 모드 제공
 - ✓ Stateful packet filtering
 - ✓ OS/Network 핑거프린팅 필터링
 - ✓ 방화벽 로그
 - ✓ 이중화 (고가용성)
 - ✓ 룰 그룹 관리 (Aliases)DDoS 방어 (SynProxy)
 - ✓ VPN (IPSEC/OpenVPN/PPTP/SSH 터널링 연동)
 - ✓ 웹필터링/웹프락시 (SquidGuard)
 - ✓ AntiVirus (ClamAV)
 - ✓ 모니터링 (CPU, Throughput, 그래프, 포털)

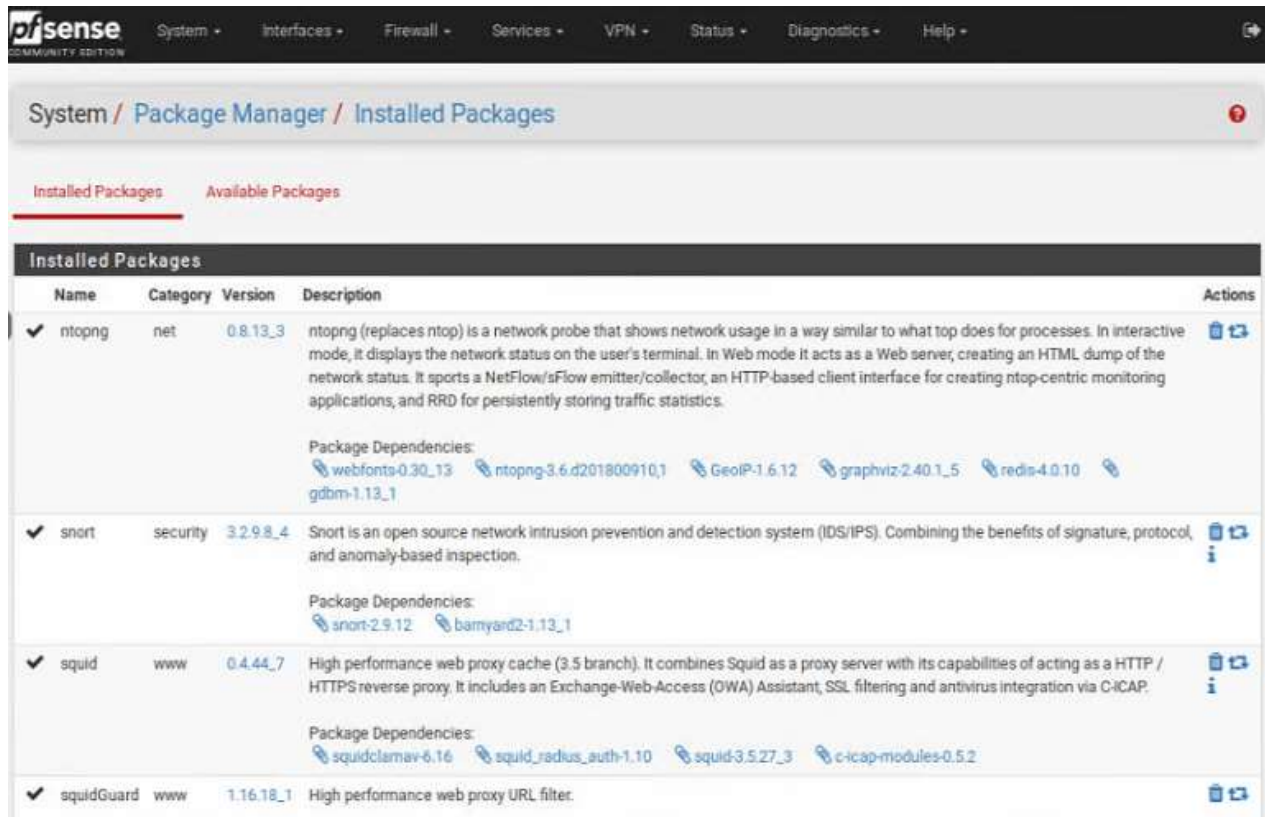
메모:

- pfSense 이미지 다운로드: <https://www.pfsense.org/download/>
 - ISO 이미지 사용 (AMD64 64비트용)
-











2. vUTM 설치

❖ vUTM 개요

- **실습 설치 (pfSense)**
 - ✓ ntopng (플로우 모니터)
 - ✓ Snort (IDS/IPS)
 - ✓ Squid (프락시/웹필터)
 - ✓ SquidGuard (웹필터)



The screenshot shows the pfSense Package Manager interface. The breadcrumb navigation is 'System / Package Manager / Installed Packages'. There are two tabs: 'Installed Packages' (active) and 'Available Packages'. Below the tabs is a table of installed packages with columns for Name, Category, Version, Description, and Actions.

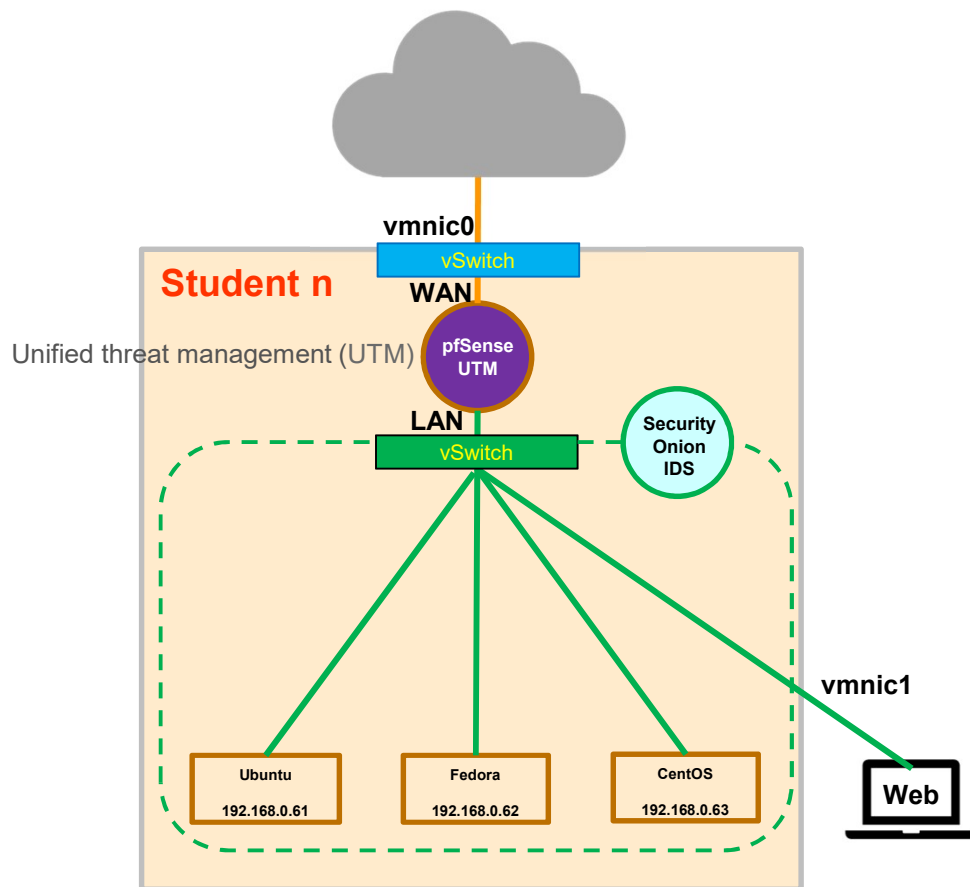
Name	Category	Version	Description	Actions
✓ ntopng	net	0.8.13_3	ntopng (replaces ntop) is a network probe that shows network usage in a way similar to what top does for processes. In interactive mode, it displays the network status on the user's terminal. In Web mode it acts as a Web server, creating 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. Package Dependencies: webfonts-0.30_13 ntopng-3.6.d201800910,1 GeoIP-1.6.12 graphviz-2.40.1_5 redis-4.0.10 gdbm-1.13_1	 
✓ snort	security	3.2.9.8_4	Snort is an open source network intrusion prevention and detection system (IDS/IPS). Combining the benefits of signature, protocol, and anomaly-based inspection. Package Dependencies: snort-2.9.12 bamyard2-1.13_1	  
✓ squid	www	0.4.44_7	High performance web proxy cache (3.5 branch). It combines Squid as a proxy server with its capabilities of acting as a HTTP / HTTPS reverse proxy. It includes an Exchange-Web-Access (OWA) Assistant, SSL filtering and antivirus integration via C-ICAP. Package Dependencies: squidclamav-6.16 squid_radius_auth-1.10 squid-3.5.27_3 c-icap-modules-0.5.2	  
✓ squidGuard	www	1.16.18_1	High performance web proxy URL filter.	 

메모:

2. vUTM 설치

❖ vUTM 'pfSense' 설치 환경

- ① 하이퍼바이저 내 인터넷용과 호스트 연결 스위치 2개 필요
- ② WAN은 인터넷, LAN은 호스트 연결 vSwitch 별도 생성
- ③ 센서 접속 부분의 스위치는 미리 기능 제공 세팅 필요
- ④ 설정을 위한 클라이언트는 VM 또는 유선랜 연결 PC 사용 (외부 유선랜 연결이 어려운 경우 하이퍼바이저에 웹으로 연결 사용)



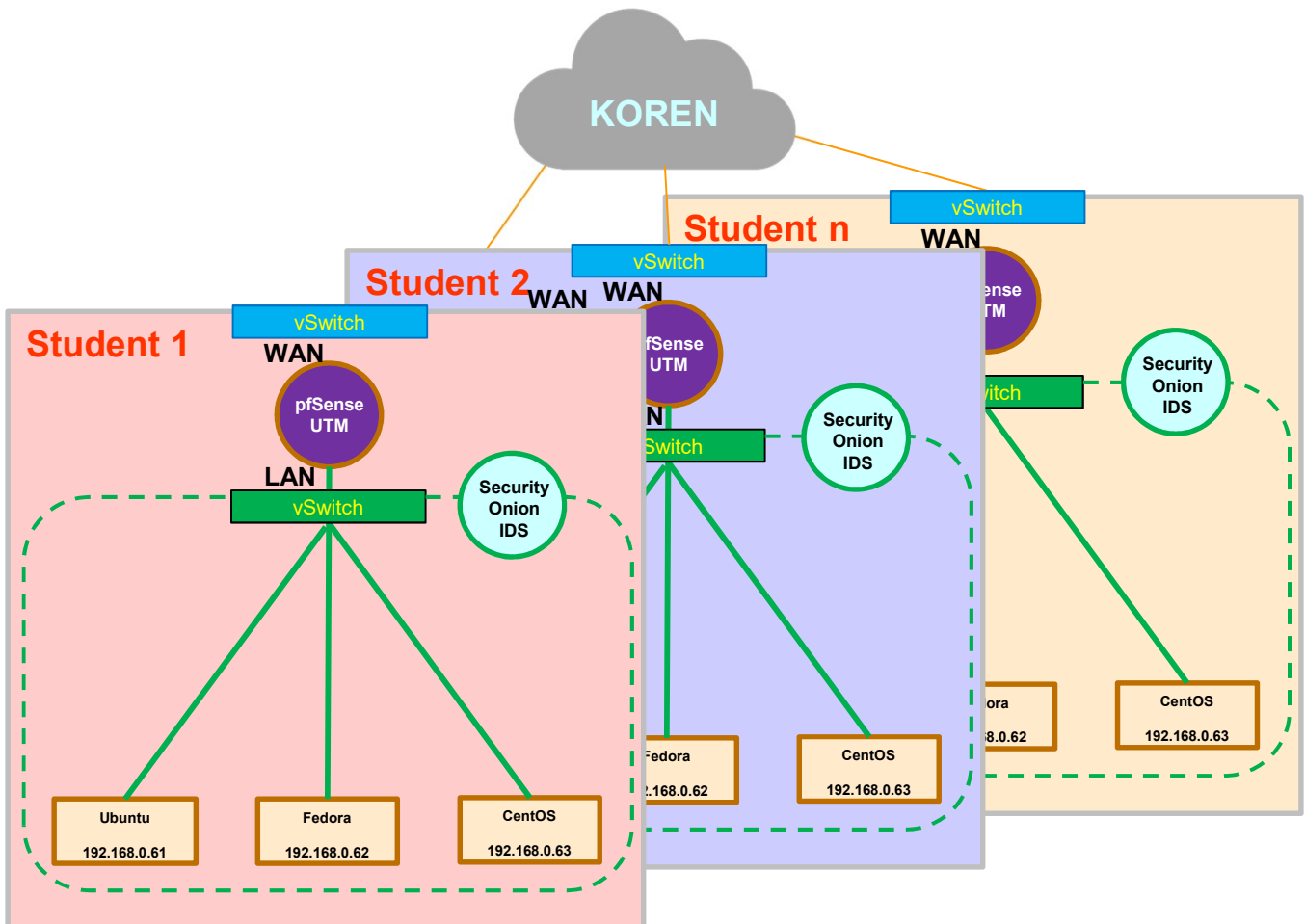
메모:

- pfSense는 IDS/IPS, 방화벽, LB, 웹방화벽, NAT, DHCP 서버 등의 기능 제공

2. vUTM 설치

❖ vUTM 'pfSense' 설치 환경

- ① WAN은 개인별 고정 IP주소 설정 권장
- ② LAN은 임의의 IP주소 설정 가능 (클라이언트를 위한 DHCP 서버 사용과 보안 기기를 위한 고정 IP 주소 사용)



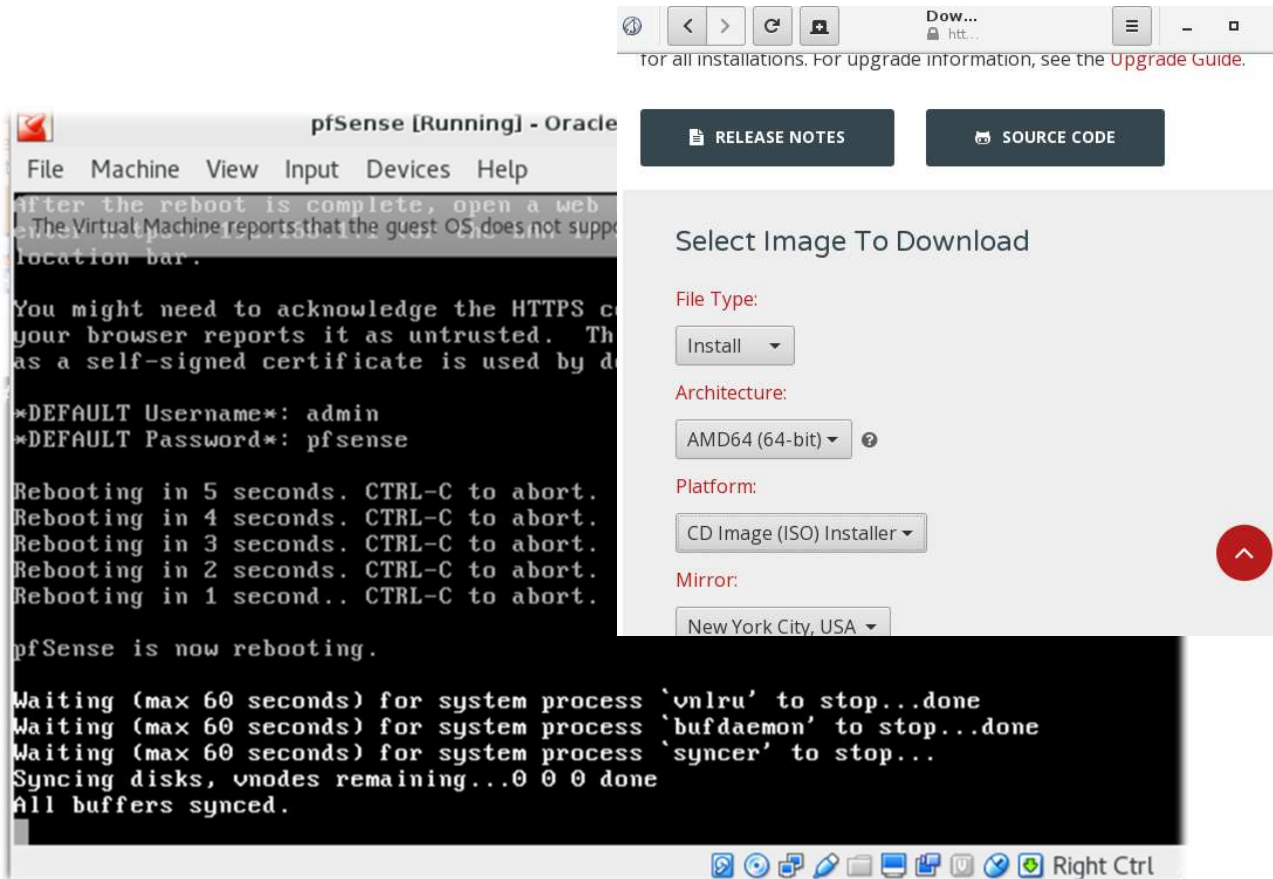
메모:

- Host는 Ubuntu Desktop과 Fedora Workstation ISO 이미지 제공
- vUTM과 vIDS용 ISO 이미지 사용 설치

2. vUTM 설치

❖ pfSense 설치 준비

- pfSense 설치 (Type 1 또는 Type 2 하이퍼바이저 사용 가능)
 - ① 다운로드: pfsense site (<https://www.pfsense.org/>)
 - ② 2개 이상 인터페이스 지정 (WAN/LAN)
 - ③ ISO 이미지 다운로드 (또는 USB Memory)
 - ④ pfSense 설치 (VirtualBox or 베어메탈 서버 or Type 1 하이퍼바이저)



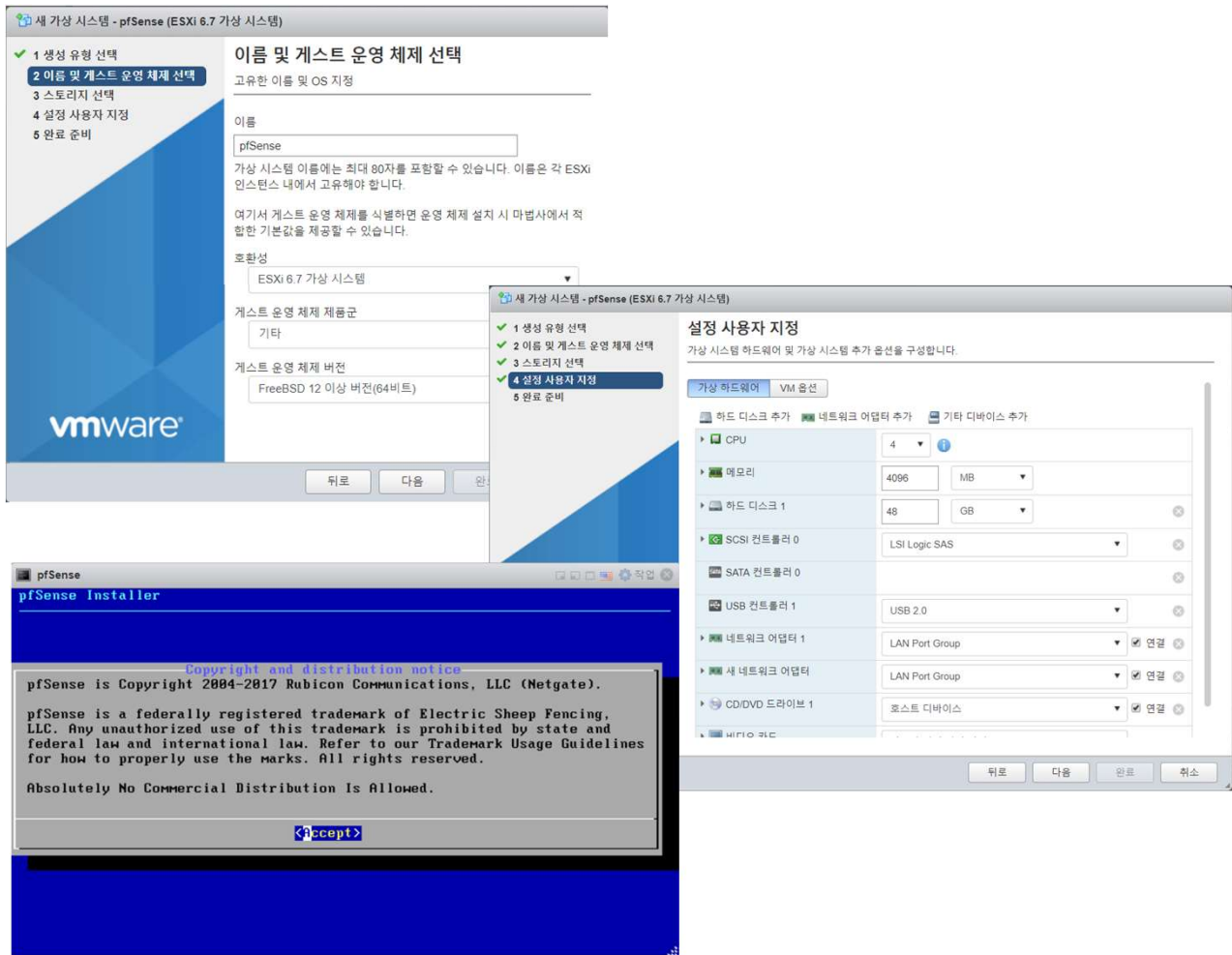
메모:

- pfSense 다운로드 주소: <https://www.pfsense.org/>
- ESXi 설치시 가상 스위치를 L2 Looping 을 방지하는 구성으로 해야함
- ESXi 설치시 동일 네트워크에 여러 사용자가 동시 접속 시 VyOS의 라우팅 사용 권장

2. vUTM 설치

❖ vUTM 'pfSense' 설치

- ① 이름과 운영체제 선택
- ② 자원 설정 (vCPU/vRAM/vHDD)
- ③ 설치



메모:

- pfSense 이미지 다운로드: <https://www.pfsense.org/download/>
- pfSense는 IDS/IPS, 방화벽, LB, 웹방화벽, NAT, DHCP 서버 등의 기능 제공
- ISO 이미지 사용 (AMD64 64비트용)

2. vUTM 설치

❖ vUTM 'pfSense' 연결 설정

- 1) Assign Interfaces (LAN / WAN 설정)
- LAN / WAN MAC 주소 확인 @ 하이퍼바이저

The image shows two overlapping windows. The top window is a terminal for pfSense 2.4.4-RELEASE, displaying boot logs and a menu with options like '1) Assign Interfaces'. The bottom window is the ESXi vMotion settings for 'pfSense (ESXI 6.7 가상 시스템)'. It shows two network adapters (네트워크 어댑터 1 and 2) configured with VMXNET 3 adapters. The MAC addresses are highlighted with red dashed boxes: '00:0c:29:68:0d:00' for the first adapter and '00:0c:29:68:0d:6a' for the second. Blue dashed arrows point from the terminal's output to these MAC addresses in the ESXi settings.

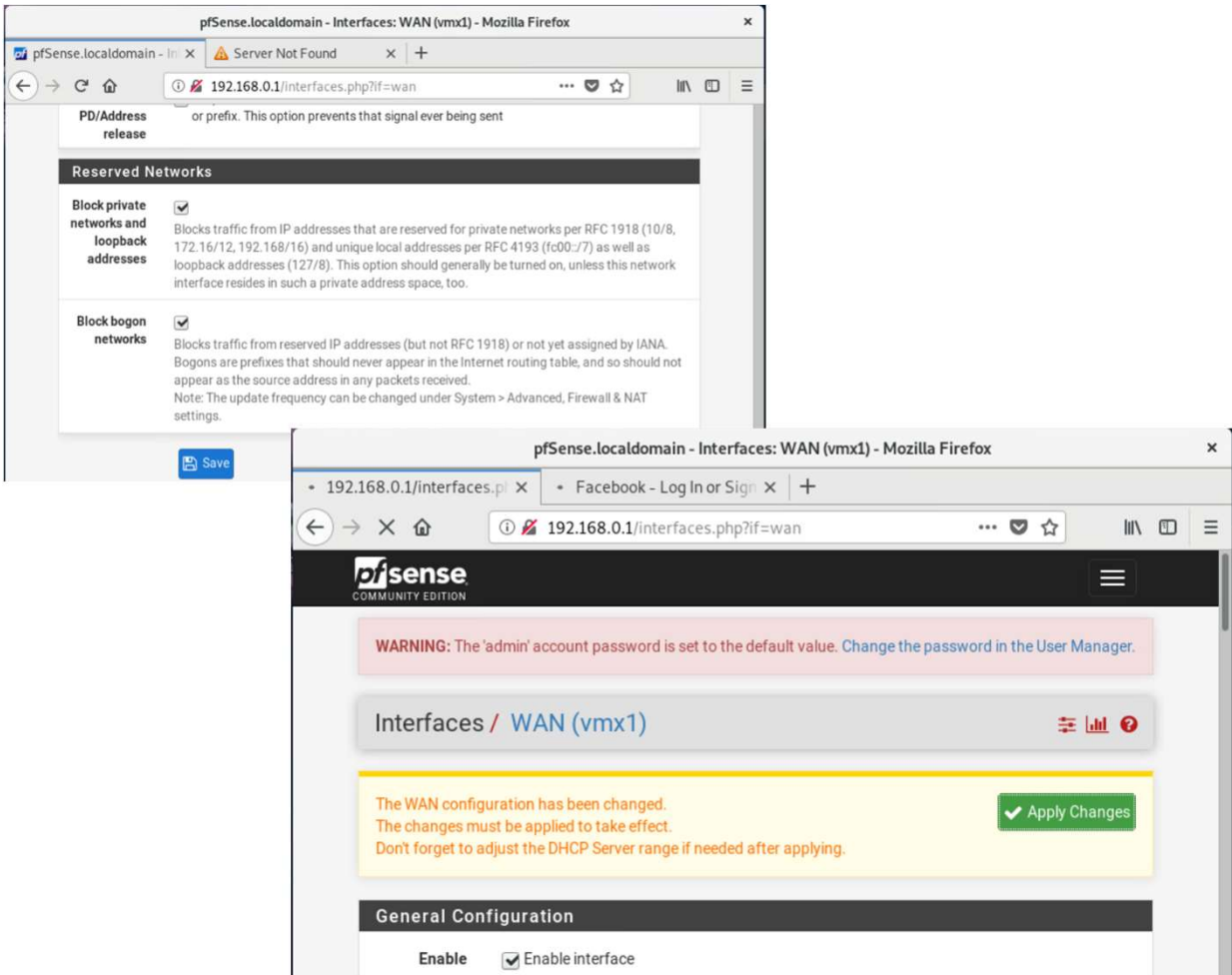
메모:

- pfSense 이미지 다운로드: <https://www.pfsense.org/download/>
- pfSense는 IDS/IPS, 방화벽, LB, 웹방화벽, NAT, DHCP 서버 등의 기능 제공
- ISO 이미지 사용 (AMD64 64비트용)

2. vUTM 설치

❖ vUTM 'pfSense' 설정 환경

- 사실 IP지원 설정 확인 (uncheck Block)
- Click Button “Apply Changes”



메모:

- RFC1918: 인터넷 어드레싱 아키텍처에서 사실 IP 주소 공간을 이용하는 표준

RFC1918 이름	IP 주소 범위	주소 개수	클래스 내용	최대 사이드 블록 (서브넷 마스크)	호스트 ID 크기
24비트 블록	10.0.0.0 – 10.255.255.255	16,777,216	클래스 A 하나	10.0.0.0/8 (255.0.0.0)	24 비트
20비트 블록	172.16.0.0 – 172.31.255.255	1,048,576	16개의 인접 클래스 B	172.16.0.0/12 (255.240.0.0)	20 비트
16비트 블록	192.168.0.0 – 192.168.255.255	65,536	256개의 인접 클래스 C	192.168.0.0/16 (255.255.0.0)	16 비트

목차



1. 실습 환경 준비
2. vUTM 설치 (pfSense..)
- 3. Host 설치 (Linux..)**
4. vLAN 구성 (e2e..)
5. vFW (pfSense..)
6. vIPS (snort)
7. 웹필터 (SquidGuard)
8. ntopng (플로우 모니터)
9. vIDS (Security Onion)

3. Host 설치 (Linux)

❖ Host 설치 환경

- ① **ISO 파일 선택** # Type 1 하이퍼바이저 설치시
 - Ubuntu Desktop 18.04
 - Fedora Workstation 29
 - Ubuntu Server 16.04 (Hyperledger, OpenStack 설치 시)
- ② **ISO 파일 Upload**
- ③ **Ubuntu Desktop과 Fedora Workstation 29는 시연으로 진행**
- ④ **Ubuntu Server 는 설정 순서 제공**

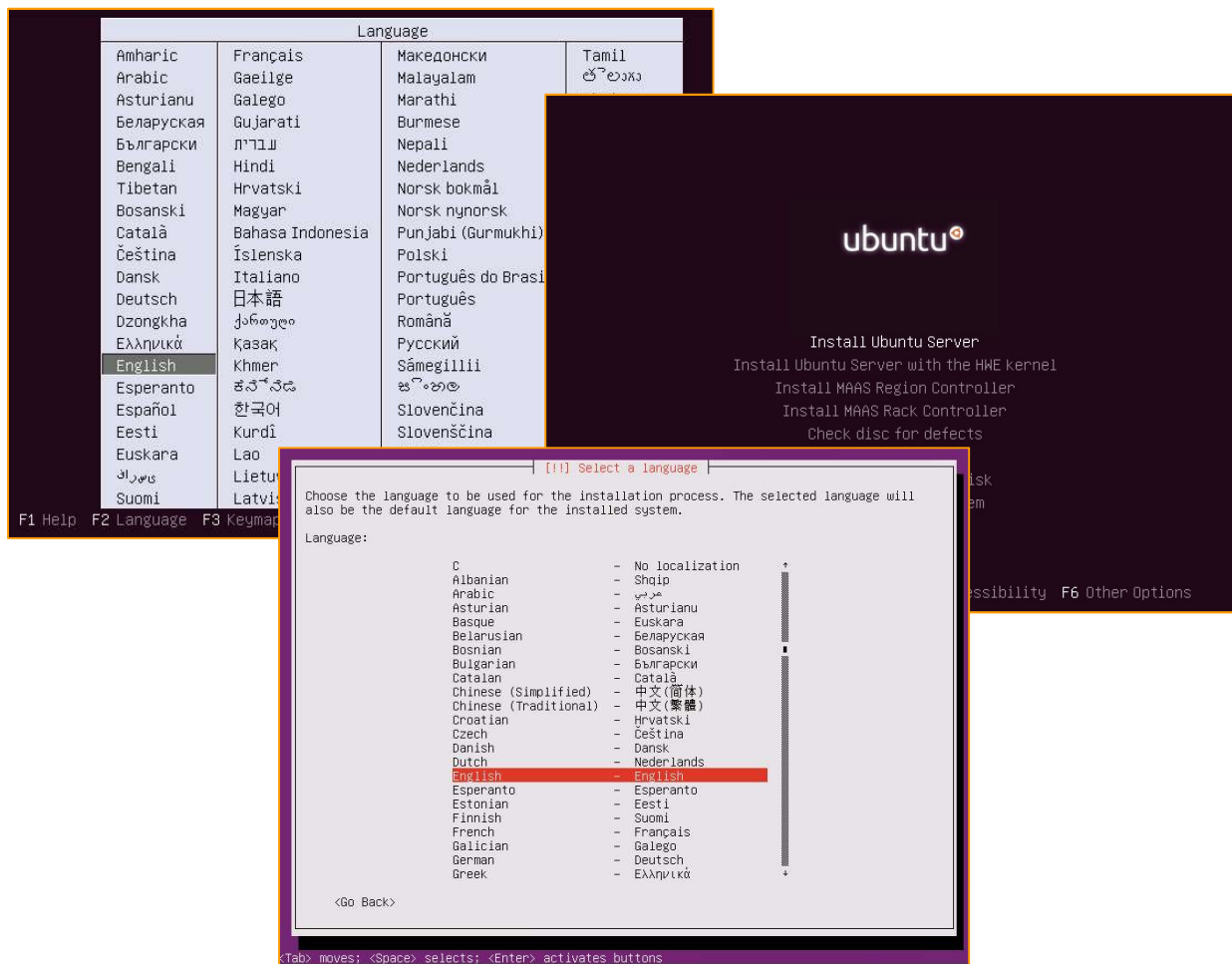
메모:

- Type 2 하이퍼바이저에서 VM 설치방법 1: 우분투(Ubuntu Server/Desktop) OVA 제공
 - Type 2 하이퍼바이저에서 VM 설치방법 2: 우분투(Ubuntu Server/Desktop) ISO 제공 설치
 - VMware Standalone Converter 사용하여 배포
 - 루트계정 활성화: `sudo passwd root`
-

3. Host 설치 (Linux)

❖ Ubuntu Server 16.04 Installation

- ① USB Booting 선택 # Bare-Metal
- ② ISO 파일 선택 # 4 GB RAM / 32 GB Storage
- ③ 언어 선택 'Korean (한국어)' and 'Continue'
- ④ 선택 'Install Ubuntu Server'



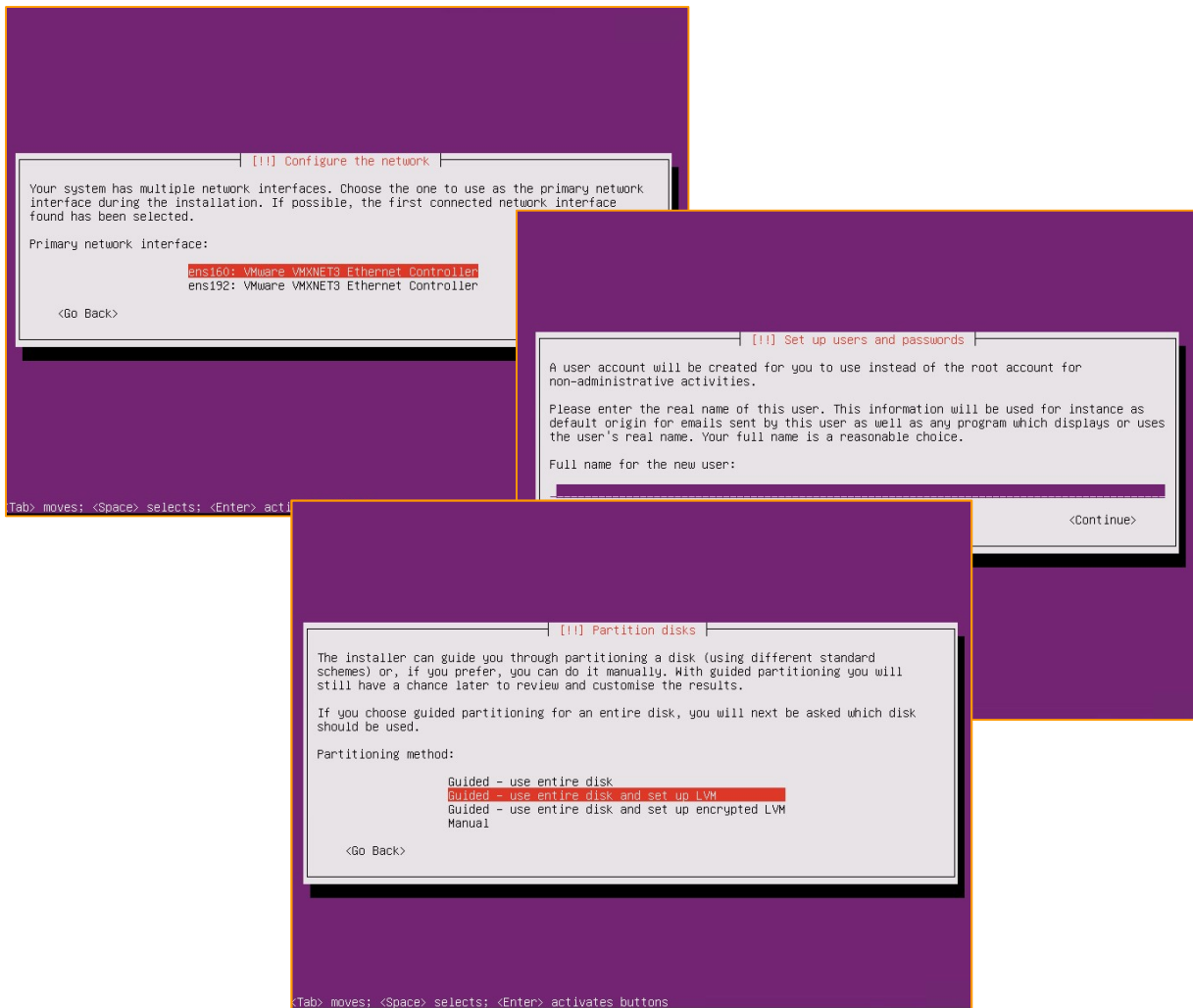
메모:

- VM 설치방법 1: 우분투(Ubuntu Server/Desktop) OVA 제공
- VM 설치방법 2: 우분투(Ubuntu Server/Desktop) ISO 제공 설치
- VMware Standalone Converter 사용하여 배포
- 루트계정 활성화: sudo passwd root

3. Host 설치 (Linux)

❖ Ubuntu Server 16.04 Installation

- ① Full Name 'jalsb'
- ② User name 'jslab'
- ③ Password 'jslab123'



메모:

3. Host 설치 (Linux)

❖ Ubuntu Server 16.04 Installation

- ① No automatic updates
- ② OpenSSH server
- ③ User name 'jslab'

[!] Configuring tasksel

Applying updates on a frequent basis is an important part of keeping your system secure.

By default, updates need to be applied manually using package management tools. Alternatively, you can choose to have this system automatically download and install security updates, or you can choose to manage this system over the web as part of a group of systems using Canonical's Landscape service.

How do you want to manage upgrades on this system?

- No automatic updates
- Install security updates automatically
- Manage system with Landscape

<Tab> moves

[!] Software selection

At the moment, only the core of the system is installed. To tune the system to your needs, you can choose to install one or more of the following predefined collections of software.

Choose software to install:

- Manual package selection
- DNS server
- LAMP server
- Mail server
- PostgreSQL database
- Samba file server
- Standard system utilities
- Virtual Machine host
- OpenSSH server

<Continue>

> activates buttons

[!] Install the GRUB boot loader on a hard disk

It seems that this new installation is the only operating system on this computer. If so, it should be safe to install the GRUB boot loader to the master boot record of your first hard drive.

Warning: If the installer failed to detect another operating system that is present on your computer, modifying the master boot record will make that operating system temporarily unbootable, though GRUB can be manually configured later to boot it.

Install the GRUB boot loader to the master boot record?

<Go Back> Yes <No>

<Tab> moves; <Space> selects; <Enter> activates buttons

메모:

3. Host 설치 (Linux)

❖ Ubuntu Server 16.04 Installation (선택)

- ① `sudo apt install lm-sensors` # sensors for Bare metal
- ② `ip link show` # Check Interfaces
- ③ **Static IP Address Setting**
- ④ **Host Name Setting**

- SSH Well-known Port 변경 -

```
sudo vi /etc/ssh/sshd_config  
  
# What ports, IPs and protocols we listen for  
Port 33322
```

- 계정 암호 변경 -

```
To change the root password:  
sudo passwd  
To change your user password:  
passwd  
To change other users password:  
sudo passwd USERNAME
```

- 호스트 이름 변경 -

```
/etc/hostname  
/etc/hosts  
sudo nano /etc/hostname  
sudo vi /etc/hosts
```

- 고정 IP 주소 설정 -

```
sudo vi /etc/network/interfaces  
  
# Iface ens160 inet dhcp  
iface ens160 inet static  
    address 192.168.0.xx  
    netmask 255.255.255.0  
    gateway 192.168.0.1  
    dns-nameservers 8.8.8.8  
  
cntl+o → enter → cntl+x  
sudo /etc/init.d/networking restart (or reboot)
```

- Root 계정 생성 -

```
sudo -l  
passwd  
sudo passwd root
```

- Putty to VyOS for sshd-

```
192.168.1.xxx @ Putty for VyOS  
ssh jslab@192.168.0.yy
```

메모:

- Ubuntu Server 루트계정 활성화: `sudo passwd root`
- VM 이미지 Import 시 네트워크 인터페이스 확인 위한 명령어 'ip link show'
- Root 계정으로 실행 필요시 (sudo 사용 일반 계정은 실행하지 못함)
루트계정 활성화: `sudo passwd root`

3. Host 설치 (Linux)

❖ Static IP for WiFi (Ubuntu 18.04)

- OVS (Open vSwitch) Mirroring (2.8.0)

1. ip link show

```
james@ubuntu18:/etc/netplan$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
3: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel master ovs-system state UP mode DEFAULT group default qlen 1000
   link/ether 00:aa:2a:e8:34:21 brd ff:ff:ff:ff:ff:ff
4: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel master ovs-system state UP mode DEFAULT group default qlen 1000
   link/ether 00:aa:2a:e8:34:22 brd ff:ff:ff:ff:ff:ff
5: enp4s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
   link/ether 00:aa:2a:e8:34:23 brd ff:ff:ff:ff:ff:ff
7: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 96:be:89:0f:df:b5 brd ff:ff:ff:ff:ff:ff
8: ovs1qotom: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 00:aa:2a:e8:34:20 brd ff:ff:ff:ff:ff:ff
9: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN mode DEFAULT group default
   link/ether 02:42:ee:0f:69:c6 brd ff:ff:ff:ff:ff:ff
10: wlx742f68923076: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
   link/ether 74:2f:68:92:30:76 brd ff:ff:ff:ff:ff:ff
12: enp1s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel master ovs-system state DOWN mode DEFAULT group default qlen 1000
   link/ether 00:aa:2a:e8:34:20 brd ff:ff:ff:ff:ff:ff
james@ubuntu18:/etc/netplan$
```

2. cd /etc/netplan

3. sudo nano 01-network-manager-all.yaml

```
network:
  version: 2
  renderer: networkd
  wifis:
    wlx742f68923076:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.0.18/24,]
      gateway4: 192.168.0.1
      nameservers:
        search: [vpsphere.local]
        addresses: [192.168.0.1, 8.8.8.8]
      access-points:
        Tech-Support:
          password: 12345*****
```

4. sudo netplan generate

5. sudo netplan apply

메모:

- ❖ <https://www.tecmint.com/configure-network-static-ip-address-in-ubuntu/>

목차

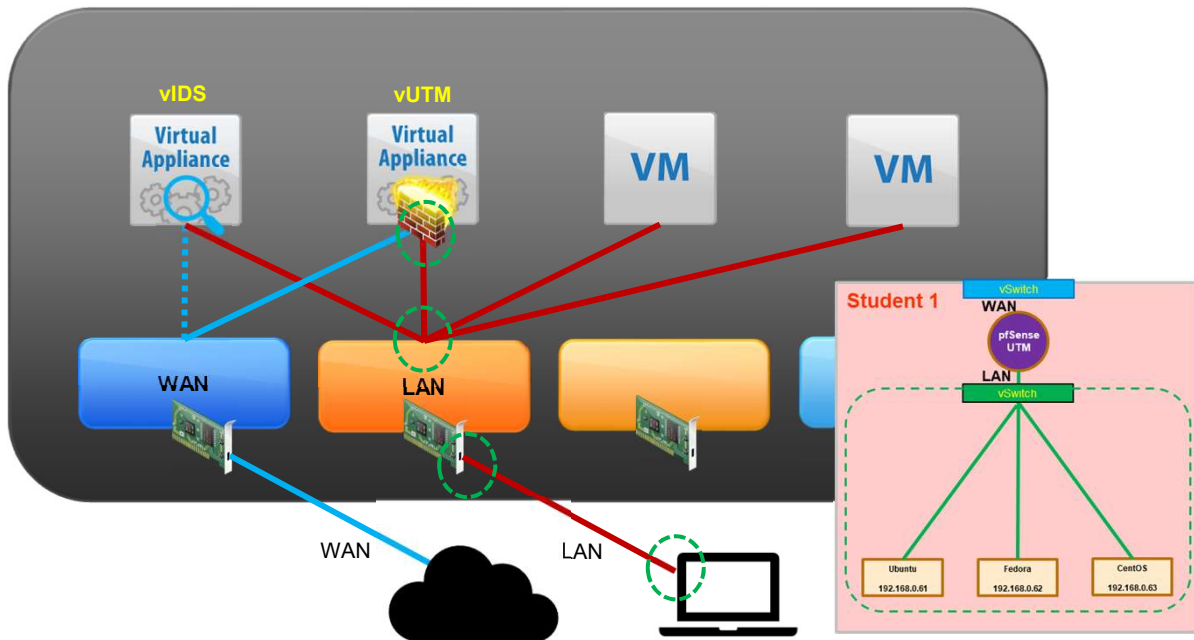


1. 실습 환경 준비
2. vUTM 설치 (pfSense..)
3. Host 설치 (Linux..)
4. **vLAN 구성 (e2e..)**
5. vFW (pfSense..)
6. vIPS (snort)
7. 웹필터 (SquidGuard)
8. ntopng (플로우 모니터)
9. vIDS (Security Onion)

4. vLAN 구성 (e2e..)

❖ vLAN 설정

- ① vLAN 설정 구성 요소 확인
- ② 하이퍼바이저 내의 가상스위치 포트 vLAN 설정
- ③ 가상스위치에 접속하는 VM/노트북/vUTM/vIDS 등 모든 기기의 인터페이스
 - Interfaces → Assignment → VLAN @ vUTM 'pfSense'
 - Interface Assignment → Available port @ vUTM 'pfSense'
 - Interfaces → enable OPT1 port → Apply @ vUTM 'pfSense'
 - Firewall → Protocol 'any' → Source 해당 vLAN 선택
 - Service → DHCP Server



메모:

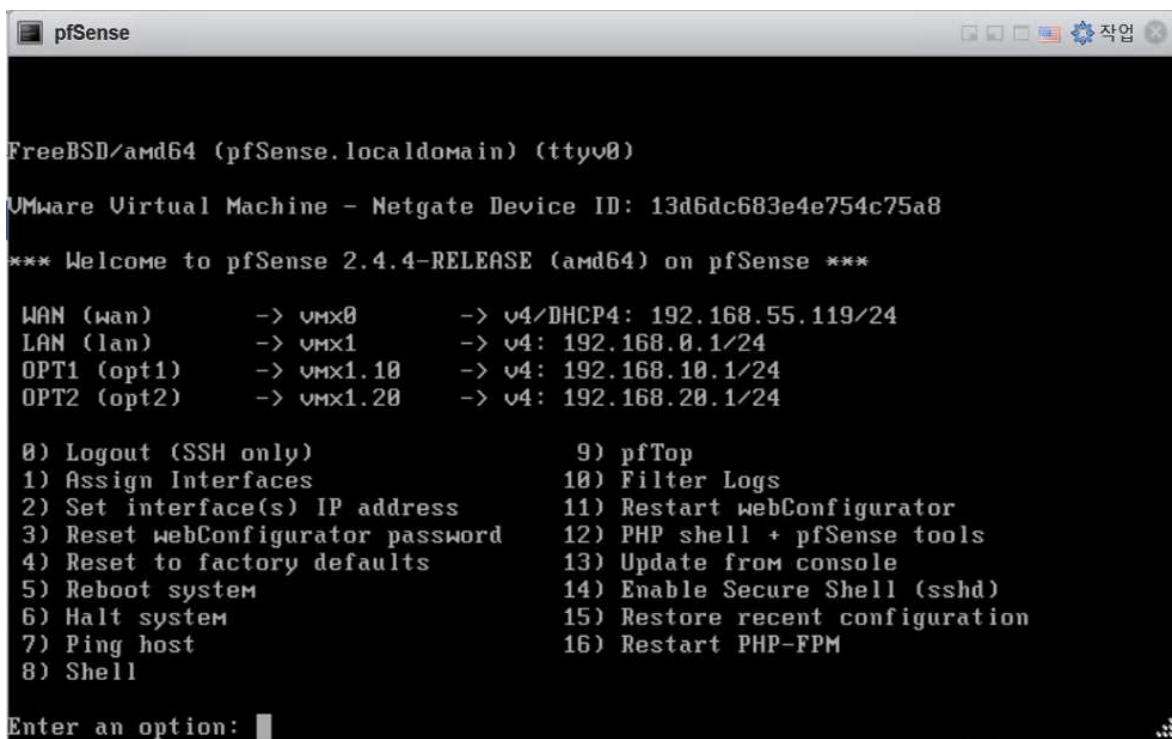
4. vLAN 구성 (e2e..)

❖ vLAN 설정

① vLAN 설정 구성 요소 확인

② 하이퍼바이저 내의 가상스위치 포트 vLAN 설정

- Interfaces → Assignment → VLAN @ vUTM 'pfSense'
- Interface Assignment → Available port @ vUTM 'pfSense'
- Interfaces → enable OPT1 port → Apply @ vUTM 'pfSense'
- Interfaces → enable OPT2 port → Apply @ vUTM 'pfSense'
- Firewall → Protocol 'any' → Source 해당 vLAN 선택
- Service → DHCP Server



```
FreeBSD/amd64 (pfSense.localdomain) (ttyv0)
VMware Virtual Machine - Netgate Device ID: 13d6dc683e4e754c75a8
*** Welcome to pfSense 2.4.4-RELEASE (amd64) on pfSense ***

WAN (wan)      -> vmx0      -> v4/DHCP4: 192.168.55.119/24
LAN (lan)      -> vmx1      -> v4: 192.168.0.1/24
OPT1 (opt1)    -> vmx1.10   -> v4: 192.168.10.1/24
OPT2 (opt2)    -> vmx1.20   -> v4: 192.168.20.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults    13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: |
```

메모:

4. vLAN 구성 (e2e..)

❖ VLAN Setting @ Port Group

- ① VLAN Setting @ Port Group # '0' for None '4095' for All
- ② Port Group의 가상 스위치 확인

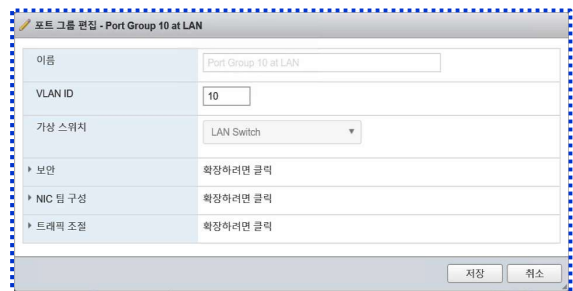
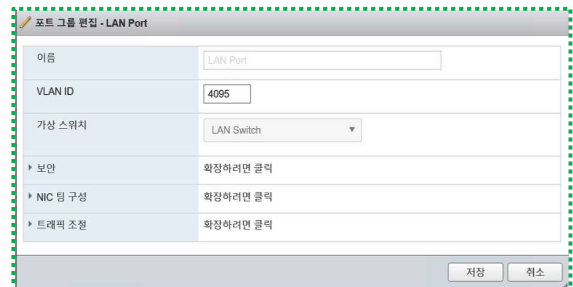
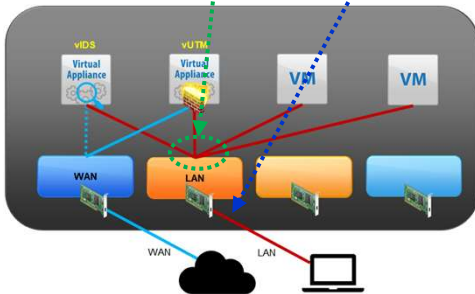
localhost.localdomain - 네트워크

포트 그룹 가상 스위치 물리적 NIC VMkernel NIC TCP/IP 스택 방화벽 규칙

포트 그룹 추가 설정 편집 새로 고침 작업 검색

이름	활성 포트	VLAN ID	유형	vSwitch	VM
VLAN10	1	10	표준 포트 그룹	vSwitch0	1
VM Network	2	0	표준 포트 그룹	vSwitch0	2
Management Network	1	0	표준 포트 그룹	vSwitch0	없음
Port Group 10 at LAN	1	10	표준 포트 그룹	LAN Switch	1
LAN Port	2	4095	표준 포트 그룹	LAN Switch	2

5 항목



메모:

- VLAN 1개 지정 실습 진행

4. vLAN 구성 (e2e..)

❖ VLAN Setting @ vUTM 'pfSense'

- ① VLAN Setting @ vUTM 'pfSense' # 1~4094
- ② vUTM 'pfSense'의 Parent Interface 확인

pfSense COMMUNITY EDITION

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

Interfaces / VLANs / Edit

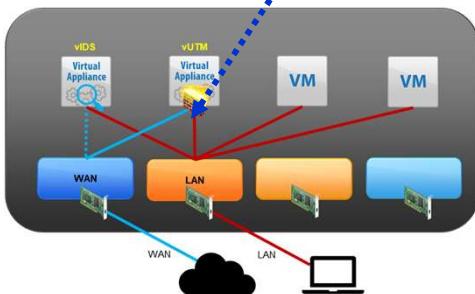
VLAN Configuration

Parent Interface
Only VLAN capable interfaces will be shown.

VLAN Tag
802.1Q VLAN tag (between 1 and 4094).

VLAN Priority
802.1Q VLAN Priority (between 0 and 7).

Description
A group description may be entered here for administrative reference (not parsed).



Private 속성

일반 | 보안 | 트래픽 조절 | NIC 팀 구성

포트 그룹 속성

네트워크 레이블:

VLAN ID(선택 사항):

메모:

- VLAN Tag : '0' for None '4095' for All

4. vLAN 구성 (e2e..)

❖ VLAN 인터페이스에 추가

- ① VLAN 인터페이스에 추가
- ② 네트워크 포트 확인

Interfaces / Interface Assignments

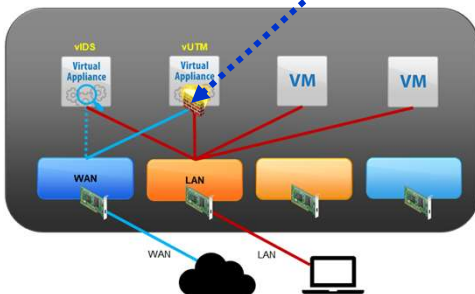
Interface Assignments Interface Groups Wireless VLANs QinQs PPPs GREs GIFs Bridges LAGGs

Interface	Network port
WAN	vmx1 (00:0c:29:88:2a:2b)
LAN	vmx0 (00:0c:29:88:2a:21) Delete

Available network ports: VLAN 10 on vmx0 - lan + Add

Save

Interfaces that are configured as members of a lag(4) interface will not be shown.
Wireless interfaces must be created on the Wireless tab before they can be assigned.



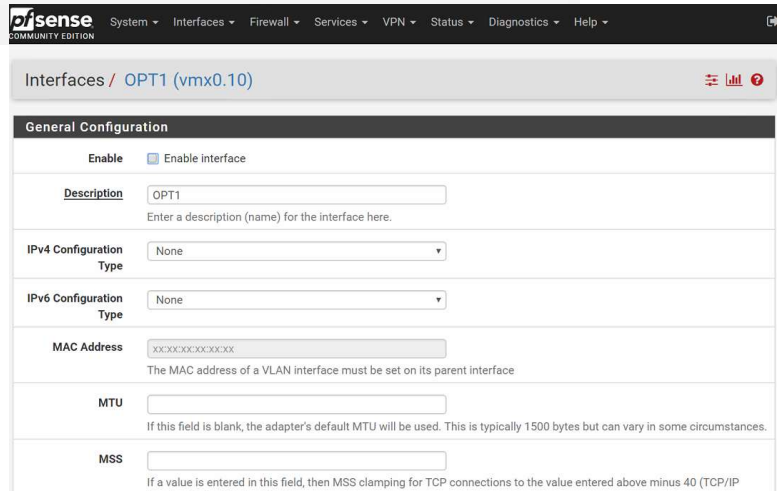
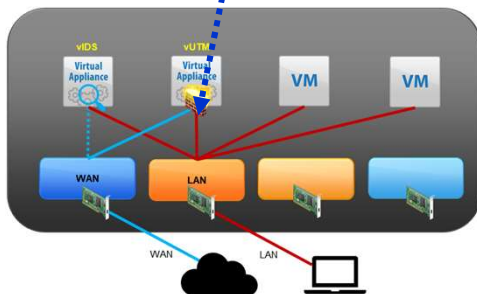
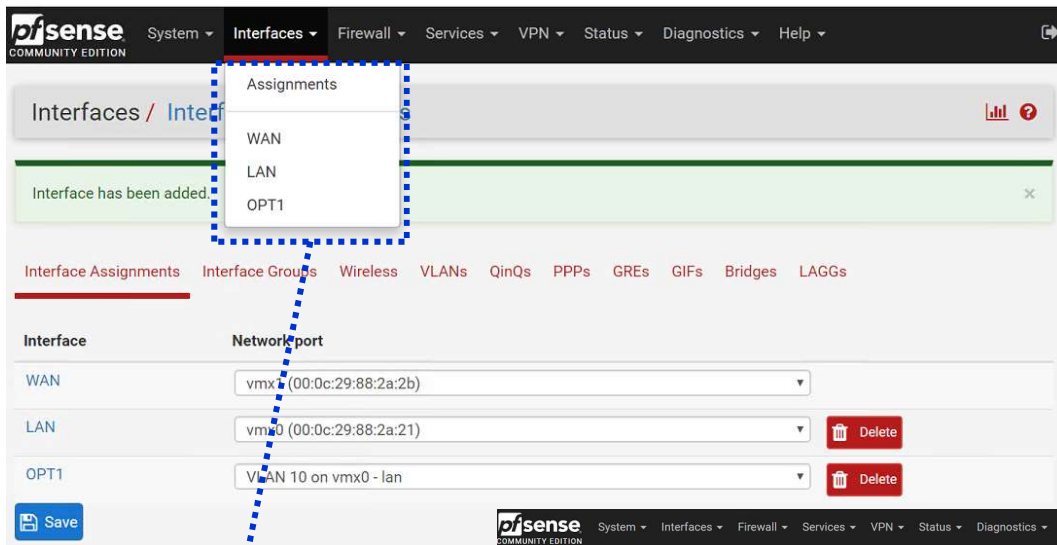
메모:

- VM 설치방법 1: 우분투(Ubuntu Server/Desktop) OVA 제공
- VM 설치방법 2: 우분투(Ubuntu Server/Desktop) ISO 제공 설치
- VMware Standalone Converter 사용하여 배포
- 루트계정 활성화: `sudo passwd root`

4. vLAN 구성 (e2e..)

❖ 인터페이스 생성

- ① 인터페이스 생성 확인
- ② 인터페이스 활성화(Enable) 후 저장(Save) 변경 저장(Apply Changes)

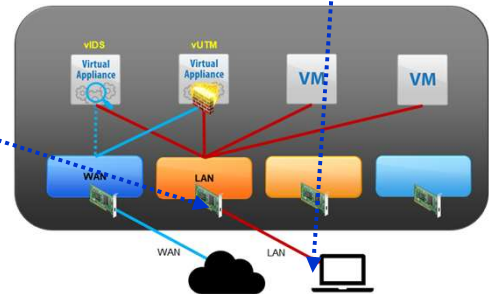
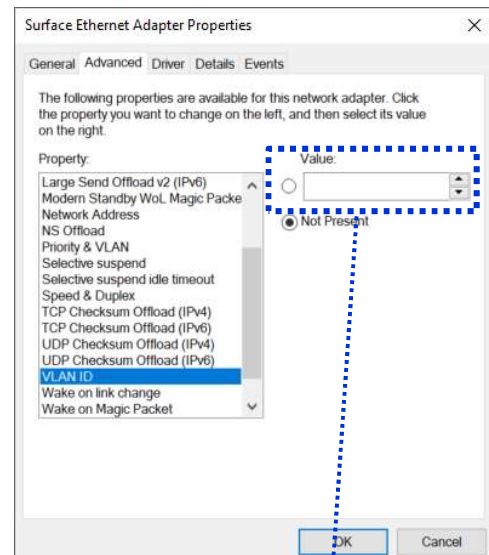
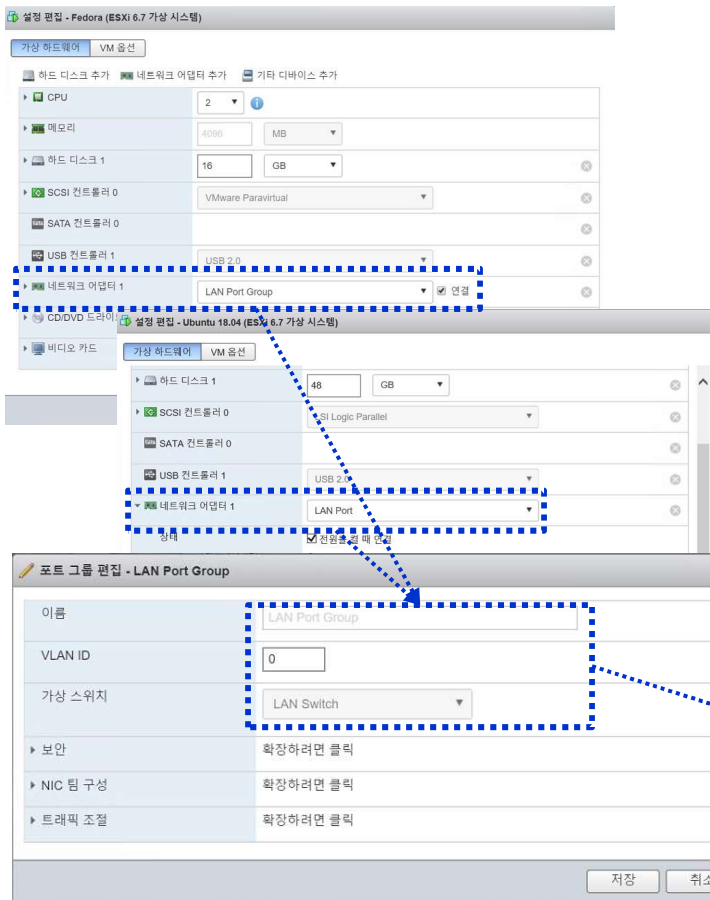


메모:

4. vLAN 구성 (e2e..)

❖ 단말기 VLAN 설정


- ① VLAN Setting @ Port Group # 1~4094
- ② 언어 선택 'Korean (한국어)' and 'Continue'
- ③ 선택 'Install Ubuntu Server'



메모:

- VLAN Tag : '0' for None '4095' for All

목차

- 
1. 실습 환경 준비
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 - 5. vFW (pfSense..)**
 6. vIPS (snort)
 7. 웹필터 (SquidGuard)
 8. ntopng (플로우 모니터)
 9. vIDS (Security Onion)

5. vFW (pfSense..)

❖ Firewall

- ① Alias RFC1918 지정
- ② Click Button “Apply Changes”

The screenshot shows the pfSense Firewall configuration interface for an Alias named 'RFC1918'. The 'Name' field is filled with 'RFC1918'. Below it, a description field is empty. The 'Type' is set to 'Network(s)'. Under the 'Network(s)' section, there is a hint explaining CIDR notation. Three network entries are listed: 10.0.0.0/8, 192.168.0.0/16, and 172.16.0.0/24. Each entry has a 'Description' field and a 'Delete' button. At the bottom, there are 'Save' and 'Add Network' buttons.

메모:

- RFC1918: 인터넷 어드레싱 아키텍처에서 사실 IP 주소 공간을 이용하는 표준

RFC1918 이름	IP 주소 범위	주소 개수	클래스 내용	최대 사이드 블록 (서브넷 마스크)	호스트 ID 크기
24비트 블록	10.0.0.0 – 10.255.255.255	16,777,216	클래스 A 하나	10.0.0.0/8 (255.0.0.0)	24 비트
20비트 블록	172.16.0.0 – 172.31.255.255	1,048,576	16개의 인접 클래스 B	172.16.0.0/12 (255.240.0.0)	20 비트
16비트 블록	192.168.0.0 – 192.168.255.255	65,536	256개의 인접 클래스 C	192.168.0.0/16 (255.255.0.0)	16 비트

5. vFW (pfSense..)

❖ Firewall

- ① vLAN간 차단
- ② Click Button “Apply Changes”

Firewall / Rules / OPT1

The changes have been applied successfully. The system is reloading in the background.

Floating WAN LAN **OPT1** OPT2

Rules (Drag to Change Order)											
	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	✗	0/0 B	IPv4 TCP	OPT2 net	*	OPT2 net	*	*	none		Add Delete Save Separator

메모:

- RFC1918: 인터넷 어드레싱 아키텍처에서 사설 IP 주소 공간을 이용하는 표준

RFC1918 이름	IP 주소 범위	주소 개수	클래스 내용	최대 사이드 블록 (서브넷 마스크)	호스트 ID 크기
24비트 블록	10.0.0.0 – 10.255.255.255	16,777,216	클래스 A 하나	10.0.0.0/8 (255.0.0.0)	24 비트
20비트 블록	172.16.0.0 – 172.31.255.255	1,048,576	16개의 인접 클래스 B	172.16.0.0/12 (255.240.0.0)	20 비트
16비트 블록	192.168.0.0 – 192.168.255.255	65,536	256개의 인접 클래스 C	192.168.0.0/16 (255.255.0.0)	16 비트

5. vFW (pfSense..)

❖ Firewall

- ① NAT
- ② 외부로 지정한 서버 서비스를 노출

Firewall / NAT / Port Forward

Port Forward 1:1 Outbound NAT

Interface	Protocol	Source Address	Source Ports	Dest. Address	Dest. Ports	NAT IP	NAT Ports	Description	Actions
WAN	TCP	*	*	WAN address	3389 (MS RDP)	192.168.1.97	3389 (MS RDP)	RDP	Pass, Linked rule

Legend
▶ Pass
⌘ Linked rule

메모:

- RFC1918: 인터넷 어드레싱 아키텍처에서 사설 IP 주소 공간을 이용하는 표준

RFC1918 이름	IP 주소 범위	주소 개수	클래스 내용	최대 사이더 블록 (서브넷 마스크)	호스트 ID 크기
24비트 블록	10.0.0.0 – 10.255.255.255	16,777,216	클래스 A 하나	10.0.0.0/8 (255.0.0.0)	24 비트
20비트 블록	172.16.0.0 – 172.31.255.255	1,048,576	16개의 인접 클래스 B	172.16.0.0/12 (255.240.0.0)	20 비트
16비트 블록	192.168.0.0 – 192.168.255.255	65,536	256개의 인접 클래스 C	192.168.0.0/16 (255.255.0.0)	16 비트

5. vFW (pfSense..)

❖ Firewall

- ① 방화벽에서 OPT1(VLAN1) 인터페이스 설정
- ② 방화벽에서 OPT2(VLAN2) 인터페이스 설정

The screenshot shows the pfSense Firewall Rules configuration page for the OPT1 interface. The breadcrumb navigation is "Firewall / Rules / OPT1". Below the navigation, there are tabs for "Floating", "WAN", "LAN", and "OPT1", with "OPT1" selected. A table titled "Rules (Drag to Change Order)" is shown with columns: States, Protocol, Source, Port, Destination, Port, Gateway, Queue, Schedule, Description, and Actions. A yellow message box states: "No rules are currently defined for this interface. All incoming connections on this interface will be blocked until pass rules are added. Click the button to add a new rule." Below the message are buttons for "Add", "Add", "Delete", "Save", and "Separator".

The network diagram shows a virtual switch (vsw) connected to two virtual interfaces (vifs) labeled "vif0" and "vif1". These vifs are connected to two virtual machines (VMs). The vifs are also connected to the LAN and WAN interfaces of the pfSense firewall. A laptop is connected to the LAN interface.


The screenshot shows the pfSense Interfaces configuration page for the OPT1 (vmx0.10) interface. The breadcrumb navigation is "Interfaces / OPT1 (vmx0.10)". The "General Configuration" section is visible, with the following settings:

- Enable: Enable interface
- Description: OPT1 (Enter a description (name) for the interface here.)
- IPv4 Configuration Type: None
- IPv6 Configuration Type: None
- MAC Address: xxxxxxxxxxxx (The MAC address of a VLAN interface must be set on its parent interface)
- MTU: (If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.)
- MSS: (If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 (TCP/IP))

메모:

- E2e는 원격지의 기관들이 서로 VLAN기반으로 연결 가능

목차

- 
1. 실습 환경 준비
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 - 6. vIPS (snort)**
 7. 웹필터 (SquidGuard)
 8. ntopng (플로우 모니터)
 9. vIDS (Security Onion)

6. vIPS (snort)

❖ 패키지 설치

① Package Manager

② Available Packages

The screenshot shows the pfSense Package Manager interface. The breadcrumb navigation is System / Package Manager / Installed Packages. There are two tabs: 'Installed Packages' (active) and 'Available Packages'. The 'Installed Packages' section contains a table with the following data:

Name	Category	Version	Description	Actions
✓ snort	security	3.2.9.8_4	Snort is an open source network intrusion prevention and detection system (IDS/IPS). Combining the benefits of signature, protocol, and anomaly-based inspection. Package Dependencies: snort-2.9.12 barnyard2-1.13_1	
✓ squid	www	0.4.44_6	High performance web proxy cache (3.5 branch). It combines Squid as a proxy server with its capabilities of acting as a HTTP / HTTPS reverse proxy. It includes an Exchange-Web-Access (OWA) Assistant, SSL filtering and antivirus integration via C-ICAP. Package Dependencies: squidclamav-6.16 squid_radius_auth-1.10 squid-3.5.27_3 c-icap-modules-0.5.2	
✓ squidGuard	www	1.16.18_1	High performance web proxy URL filter. Package Dependencies: squidguard-1.4_15	

Legend: = Update = Current
 = Remove = Information = Reinstall

Newer version available

Package is configured but not (fully) installed or deprecated

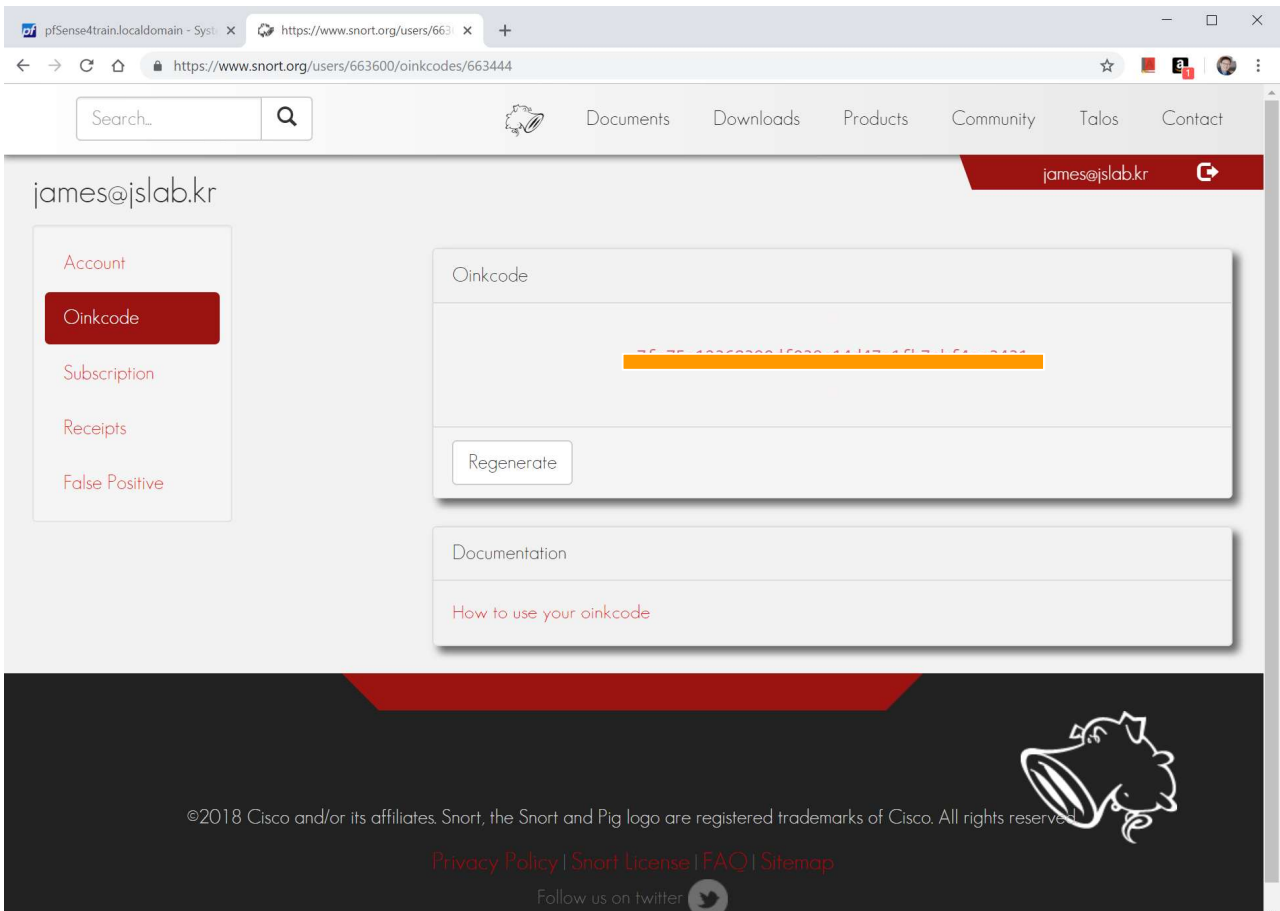
pfSense is developed and maintained by Netgate. © ESF 2004 - 2018 View license.

메모:

6. vIPS (snort)

❖ Oinkcode

- ① <https://www.snort.org/>
- ② Oinkcode



The screenshot shows a web browser window displaying the Snort Oinkcode page for the user james@jslab.kr. The browser address bar shows the URL <https://www.snort.org/users/663600/oinkcodes/663444>. The page features a navigation menu with links for Documents, Downloads, Products, Community, Talos, and Contact. A search bar is located at the top left. The main content area includes a sidebar with links for Account, Oinkcode (highlighted in red), Subscription, Receipts, and False Positive. The Oinkcode section displays a redacted Oinkcode (75 75 432622215222 4447 457 457 457 457) and a Regenerate button. Below this is a Documentation section with a link for How to use your oinkcode. The footer contains copyright information for Cisco, a privacy policy link, and a Twitter follow button.

메모:

6. vIPS (snort)

❖ Global settings

① Oinkcode

② Update Interval

The screenshot displays the pfSense web interface for the Snort Global Settings page. The breadcrumb trail is 'Services / Snort / Global Settings'. The page contains several sections for enabling and configuring different rule sets:

- Snort Subscriber Rules:** Includes 'Enable Snort VRT' (checked) and links to sign up for a free Registered User Rules Account or a paid Snort Subscriber Rule Set.
- Snort Oinkmaster Code:** A text input field for pasting a snort.org Oinkmaster code. This section is highlighted with a red dashed box.
- Snort GPLv2 Community Rules:** Includes 'Enable Snort GPLv2' (unchecked) and a description of the GPLv2 Talos certified ruleset.
- Emerging Threats (ET) Rules:** Includes 'Enable ET Open' (unchecked) and 'Enable ET Pro' (unchecked), with links to sign up for ETPro accounts.
- Sourcefire OpenAppID Detectors:** Includes 'Enable OpenAppID' (unchecked) and 'OpenAppID Version'.
- Enable RULES OpenAppID:** Includes 'Enable RULES OpenAppID' (unchecked) and a note about the AppID Open Rules file.
- Rules Update Settings:** A section at the bottom, highlighted with a red dashed box, containing the 'Update Interval' dropdown menu, which is currently set to 'NEVER'.

메모:

6. vIPS (snort)

❖ Block offender

① Snort Interfaces

② Alert Settings

The screenshot shows the pfSense web interface for configuring a Snort interface. The breadcrumb trail is 'Services / Snort / Edit Interface / None'. The 'Block Offenders' option under 'Alert Settings' is highlighted with a red dashed box. The 'Block Offenders' option is currently unchecked.

General Settings

- Enable: Enable interface
- Interface: WAN (dropdown menu)
- Description: WAN (text input)
- Snap Length: 1518 (text input)

Alert Settings

- Send Alerts to System Log: Snort will send Alerts to the firewall's system log. Default is Not Checked.
- Block Offenders**: Checking this option will automatically block hosts that generate a Snort alert

Detection Performance Settings


- Search Method: AC-BNFA (dropdown menu)
- Split ANY-ANY: Enable splitting of ANY-ANY port group. Default is Not Checked.
- Search Optimize: Enable search optimization. Default is Not Checked.
- Stream Inserts: Do not evaluate stream inserted packets against the detection engine. Default is Not Checked.
- Checksum Check Disable: Disable checksum checking within Snort to improve performance. Default is Not Checked.

메모:

- RFC1918: 인터넷 어드레싱 아키텍처에서 사설 IP 주소 공간을 이용하는 표준

RFC1918 이름	IP 주소 범위	주소 개수	클래스 내용	최대 사이더 블록 (서브넷 마스크)	호스트 ID 크기
24비트 블록	10.0.0.0 – 10.255.255.255	16,777,216	클래스 A 하나	10.0.0.0/8 (255.0.0.0)	24 비트
20비트 블록	172.16.0.0 – 172.31.255.255	1,048,576	16개의 인접 클래스 B	172.16.0.0/12 (255.240.0.0)	20 비트
16비트 블록	192.168.0.0 – 192.168.255.255	65,536	256개의 인접 클래스 C	192.168.0.0/16 (255.255.0.0)	16 비트

목차

- 
1. 실습 환경 준비
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 6. vIPS (snort)
 - 7. 웹필터 (SquidGuard)**
 8. ntopng (플로우 모니터)
 9. vIDS (Security Onion)

7. 웹필터 (SquidGuard)

❖ 웹필터

① Global Setting

② 활성화

Package / Proxy filter SquidGuard: General settings / General settings

General settings | Common ACL | Groups ACL | Target categories | Times | Rewrites | Blacklist | Log | XMLRPC Sync

General Options

Enable Check this option to enable squidGuard.
Important: Please set up at least one category on the 'Target Categories' tab before enabling. See [this link for details](#).
The Save button at the bottom of this page must be clicked to save configuration changes.
To activate squidGuard configuration changes, **the Apply button must be clicked.**

SquidGuard service state: **STOPPED**

LDAP Options

Enable LDAP Filter Enable options for setup ldap connection to create filters with ldap search

LDAP DN
Configure your LDAP DN (ex: cn=Administrator,cn=Users,dc=domain)

LDAP DN Password
Password must be initialize with letters (Ex: Change123), valid format: [a-zA-ZV][a-zA-Z0-9/_\-\.\V\:\%\+\?=&]

Strip NT domain name Strip NT domain name component from user names (/ or \ separated).

Strip Kerberos Realm Strip Kerberos Realm component from user names (@ separated).

LDAP Version

Logging options

Enable GUI log Check this option to log the access to the Proxy Filter GUI.

Enable log Check this option to log the proxy filter settings like blocked websites in Common ACL, Group ACL and Target Categories. This option is usually used to check the filter settings.

Enable log rotation Check this option to rotate the logs every day. This is recommended if you enable any kind of logging to limit file size and do not run out of disk space.

Miscellaneous

Clean Advertising Check this option to display a blank gif image instead of the default block page. With this option the user gets a cleaner webpage.

Blacklist options

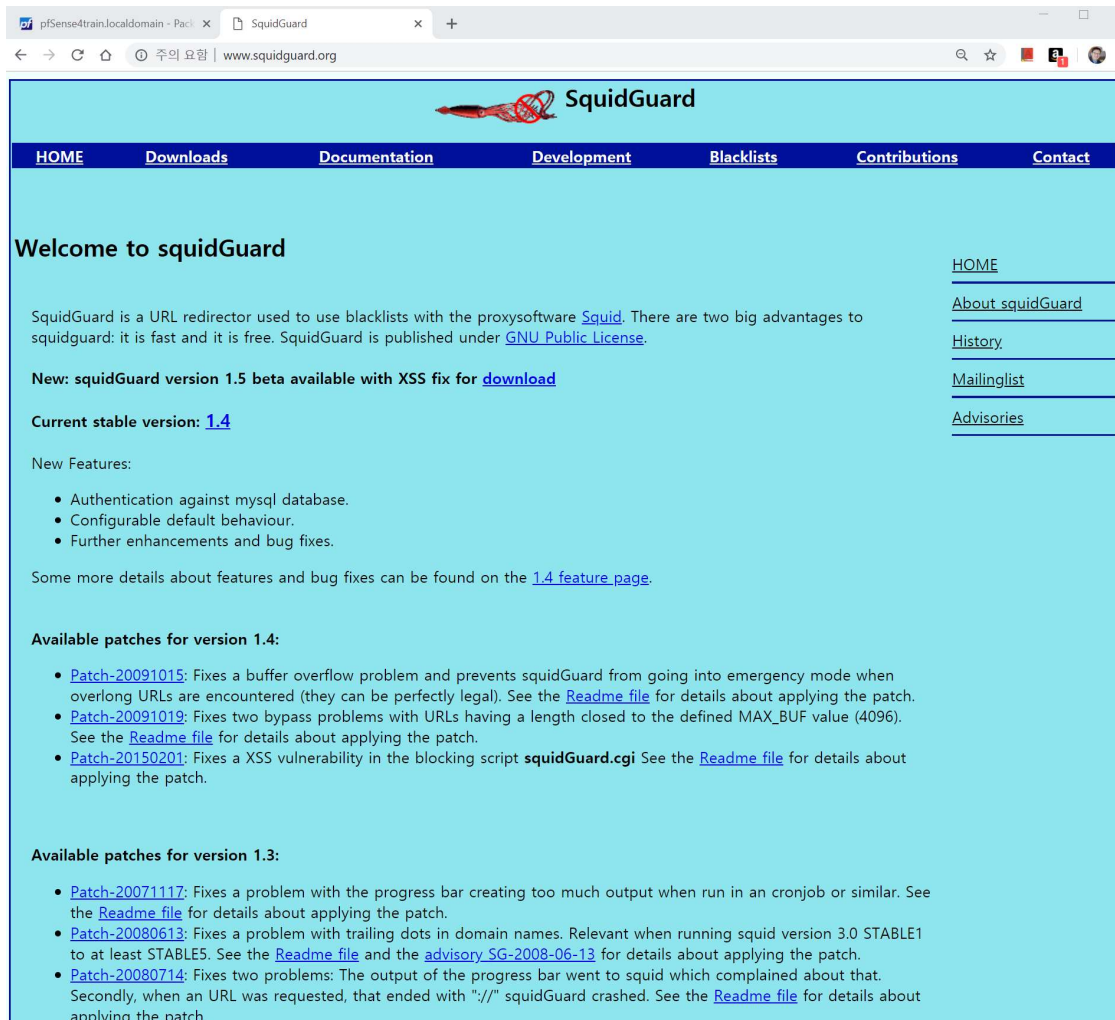
메모:

- SquidProxy 우선 실행
- Transparent Mode 확인
- <http://www.squidguard.org/>

7. 웹필터 (SquidGuard)

❖ 웹필터

- ① <http://www.squidguard.org>
- ② 외부로 지정한 서버 서비스를 노출



메모:

- <http://www.squidguard.org/>

7. 웹필터 (SquidGuard)

❖ 웹필터

① Log

② <http://www.shallalist.de/Downloads/shallalist.tar.gz>

The screenshot shows the SquidGuard configuration web interface. The breadcrumb path is "Package / Proxy filter SquidGuard: General settings / General settings". The "General Options" section includes an "Enable" checkbox which is checked, with an "Apply" button below it. The "SquidGuard service state" is shown as "STARTED". The "LDAP Options" section includes fields for "LDAP DN" and "LDAP DN Password", and checkboxes for "Strip NT domain name" and "Strip Kerberos Realm". The "Logging options" section includes checkboxes for "Enable GUI log", "Enable log", and "Enable log rotation". The "Blacklist options" section includes a "Blacklist" checkbox which is checked, and a "Blacklist URL" field containing the URL <http://www.shallalist.de/Downloads/shallalist.tar.gz>, which is circled in red in the image.

메모:

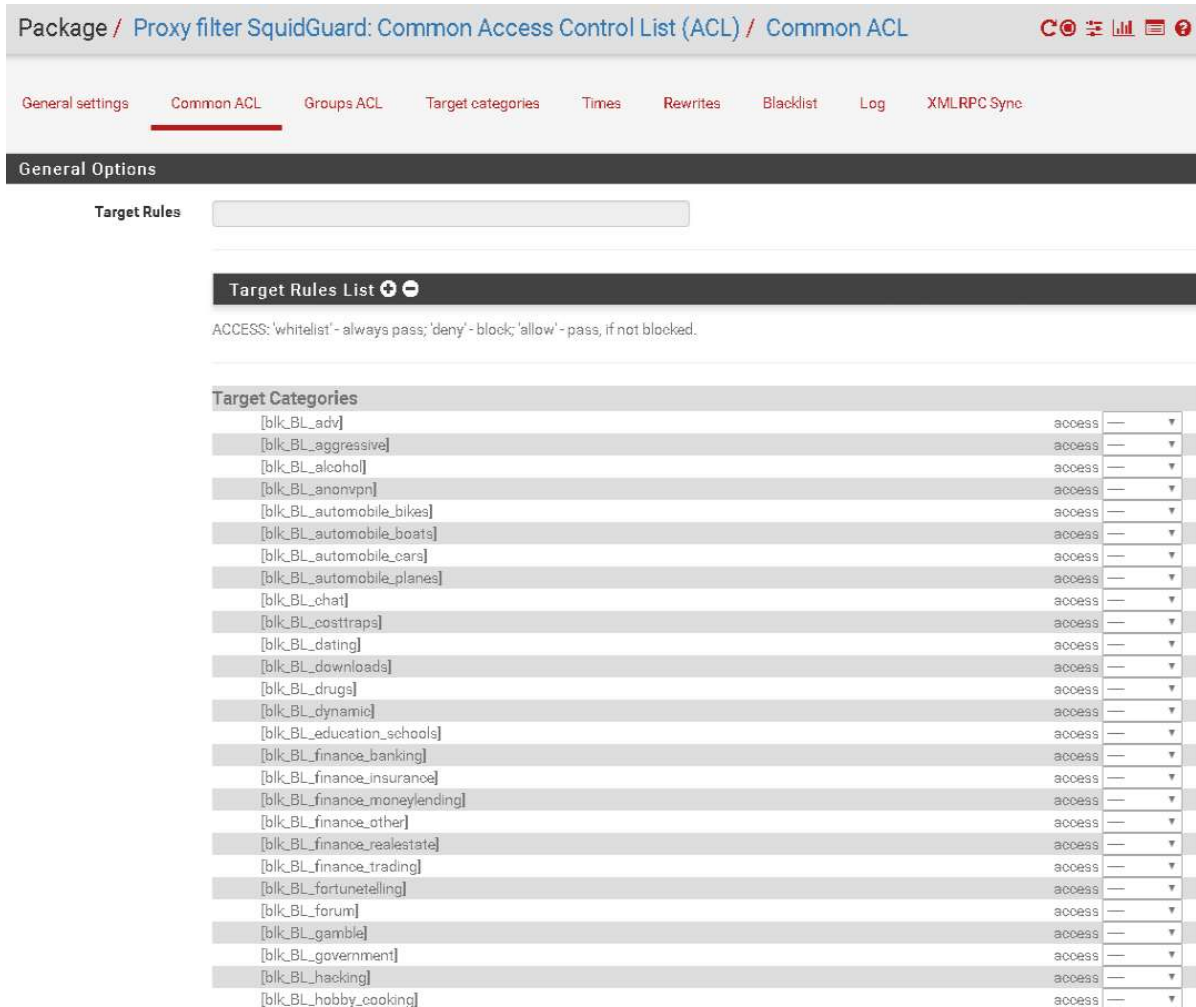
- <http://www.shallalist.de>
- <http://www.shallalist.de/Downloads/shallalist.tar.gz>

7. 웹필터 (SquidGuard)

❖ 웹필터

① Log

② <http://www.shallalist.de/Downloads/shallalist.tar.gz>



Package / Proxy filter SquidGuard: Common Access Control List (ACL) / Common ACL

General settings Common ACL Groups ACL Target categories Times Rewrites Blacklist Log XMLRPC Sync

General Options

Target Rules

Target Rules List + -

ACCESS: 'whitelist' - always pass; 'deny' - block; 'allow' - pass, if not blocked.

Target Categories

[blk_BL_adv]	access	---	▼
[blk_BL_aggressive]	access	---	▼
[blk_BL_alcohol]	access	---	▼
[blk_BL_anonymipn]	access	---	▼
[blk_BL_automobile_bikes]	access	---	▼
[blk_BL_automobile_boats]	access	---	▼
[blk_BL_automobile_cars]	access	---	▼
[blk_BL_automobile_planes]	access	---	▼
[blk_BL_chat]	access	---	▼
[blk_BL_costraps]	access	---	▼
[blk_BL_dating]	access	---	▼
[blk_BL_downloads]	access	---	▼
[blk_BL_drugs]	access	---	▼
[blk_BL_dynamic]	access	---	▼
[blk_BL_education_schools]	access	---	▼
[blk_BL_finance_banking]	access	---	▼
[blk_BL_finance_insurance]	access	---	▼
[blk_BL_finance_moneylending]	access	---	▼
[blk_BL_finance_other]	access	---	▼
[blk_BL_finance_realestate]	access	---	▼
[blk_BL_finance_trading]	access	---	▼
[blk_BL_fortunetelling]	access	---	▼
[blk_BL_forum]	access	---	▼
[blk_BL_gamble]	access	---	▼
[blk_BL_government]	access	---	▼
[blk_BL_hacking]	access	---	▼
[blk_BL_hobby_cooking]	access	---	▼

메모:

- <http://www.shallalist.de>
- <http://www.shallalist.de/Downloads/shallalist.tar.gz>

7. 웹필터 (SquidGuard)

❖ 웹필터

① **Blacklist URL**


② **<http://www.shallalist.de/Downloads/shallalist.tar.gz>**

The screenshot shows the pfSense web interface for the SquidGuard Blacklist configuration. The breadcrumb trail is "Package / SquidGuard / Blacklists". The "Blacklist" tab is selected. Under the "Blacklist Update" section, a progress bar is at 100%. Below the progress bar are buttons for "Download", "Cancel", and "Restore Default". A text input field contains the URL "http://www.shallalist.de/Downloads/shallalist.tar.gz". Below this is a "Blacklist update Log" section with the following text: "Begin blacklist update", "Start download.", "Download archive http://www.shallalist.de/Downloads/shallalist.tar.gz", "Download complete", and "Unpack archive".

메모:

- <http://www.shallalist.de>
- <http://www.shallalist.de/Downloads/shallalist.tar.gz>

목차

- 
1. 실습 환경 준비
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 6. vIPS (snort)
 7. 웹필터 (SquidGuard)
 - 8. ntopng (플로우 모니터)**
 9. vIDS (Security Onion)

8. ntopng (플로우 모니터)

❖ ntopng setting

- ① Diagnostic → ntopng settings
- ② Enable @ an interface
- ③ save
- ④ Access ntopng # <http://127.0.0.1:3000> with 'admin / admin'

ofsense COMMUNITY EDITION System ▾ Interfaces ▾ Firewall ▾ Services ▾ VPN ▾ Status ▾ Diagnostics ▾ Help ▾

Package / Diagnostics: ntopng Settings / ntopng Settings

ntopng Settings Access ntopng

General Options

Enable ntopng Check this to enable ntopng.

Keep Data/Settings 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 LAN
OPT1
OPT2
WAN

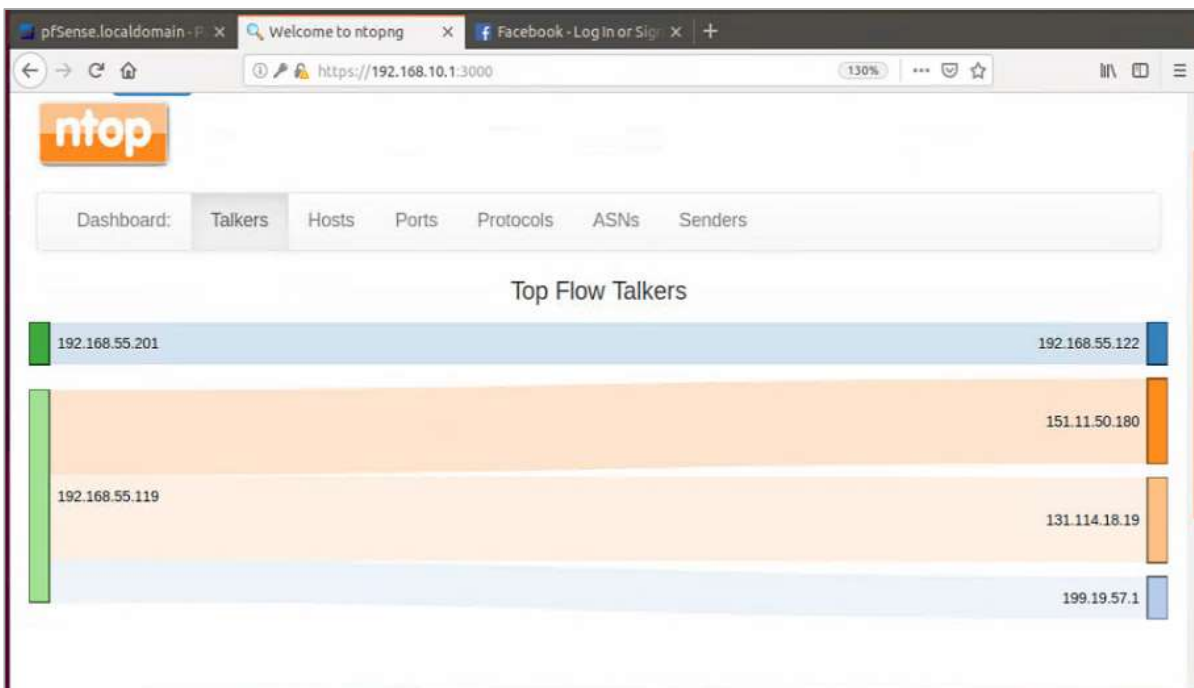
메모:

- 초기 관리 계정 / 암호 'admin / admin'

8. ntopng (플로우 모니터)

❖ ntopng Access


- ① 대쉬보드
- ② Flow
- ③ Host



메모:

- <http://www.shallalist.de>
- <http://www.shallalist.de/Downloads/shallalist.tar.gz>

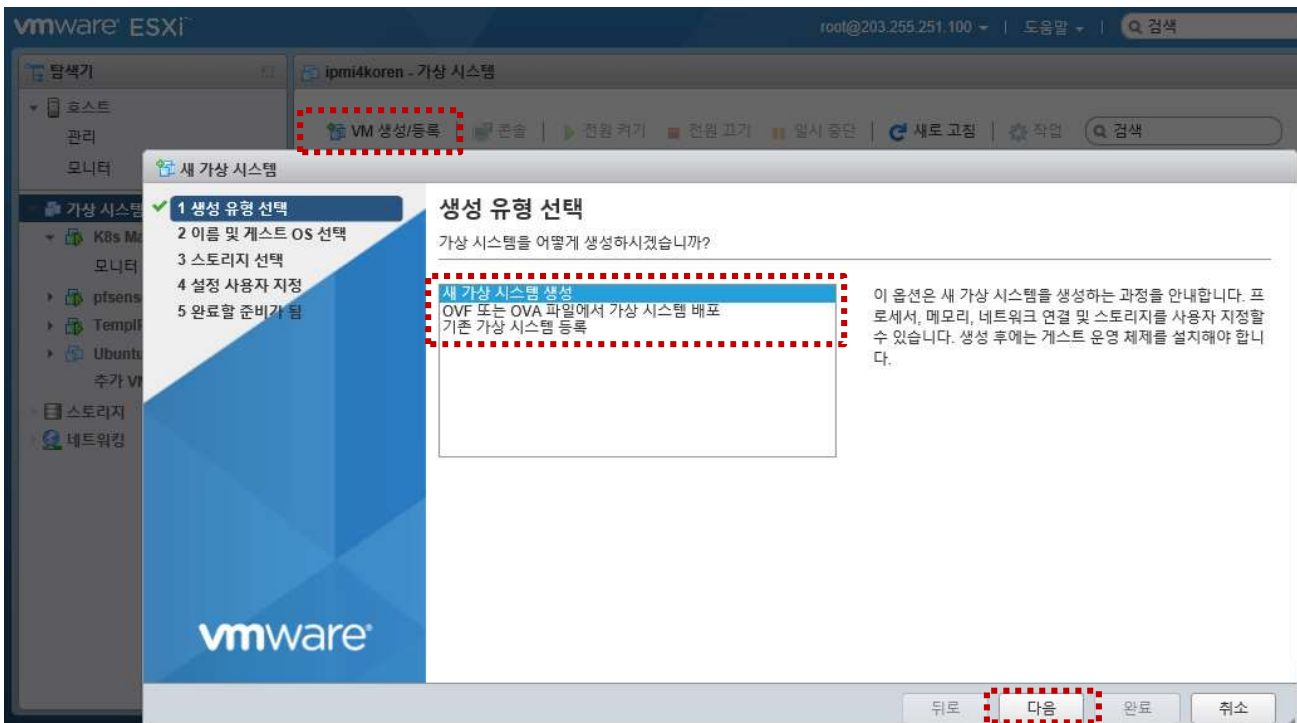
목차

- 
1. 실습 환경 준비
 2. vUTM 설치 (pfSense..)
 3. Host 설치 (Linux..)
 4. vLAN 구성 (e2e..)
 5. vFW (pfSense..)
 6. vIPS (snort)
 7. 웹필터 (SquidGuard)
 8. ntopng (플로우 모니터)
 9. **vIDS (Security Onion)**

8. vIDS

❖ Security Onion @ Hypervisor

- ① VM 생성 등록
- ② 새 가상 시스템 생성
- ③ 다음



메모:

- ESXi 다운로드 주소: <https://my.vmware.com/en/web/vmware/evalcenter?p=free-esxi6>
- 디스크 이미지 굽기: Rufus 도구 사용 <https://rufus.akeo.ie/>
- Disk Imager <https://sourceforge.net/projects/win32diskimager/files/latest/download>
- USB 부팅 제가동은 전원 off/on (전원 케이블 포함)필요함
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① 이름
- ② 호환성
- ③ 게스트 OS 제품군
- ④ 게스트 OS 버전
- ⑤ 다음

새 가상 시스템 - Security Onion (ESXi 6.5 가상 시스템)

1 생성 유형 선택
2 이름 및 게스트 OS 선택
3 스토리지 선택
4 설정 사용자 지정
5 완료할 준비가 됨

이름 및 게스트 OS 선택

고유한 이름 및 OS 지정

이름
Security Onion

가상 시스템 이름에는 최대 80자를 포함할 수 있습니다. 이름은 각 ESXi 인스턴스 내에서 고유해야 합니다.
여기서 게스트 운영 체제를 식별하면 마법사에서 해당 운영 체제 설치에 적합한 기본값을 제공할 수 있습니다.

호환성
ESXi 6.5 가상 시스템

게스트 OS 제품군
Linux

게스트 OS 버전
Ubuntu Linux(64비트)

뒤로 다음 완료 취소

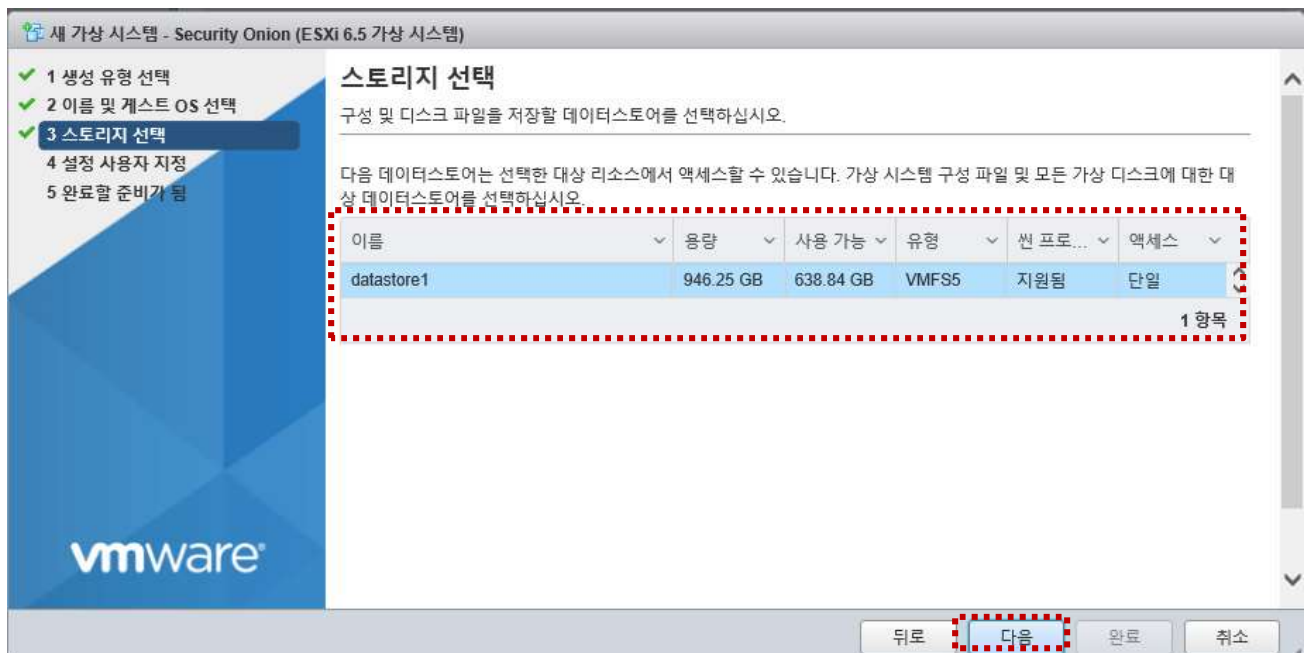
메모:

- ESXi 다운로드 주소: <https://my.vmware.com/en/web/vmware/evalcenter?p=free-esxi6>
- 디스크 이미지 굽기: Rufus 도구 사용 <https://rufus.akeo.ie/>
- Disk Imager <https://sourceforge.net/projects/win32diskimager/files/latest/download>
- USB 부팅 제가동은 전원 off/on (전원 케이블 포함)필요함
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① 스토리지 선택
- ② 다음



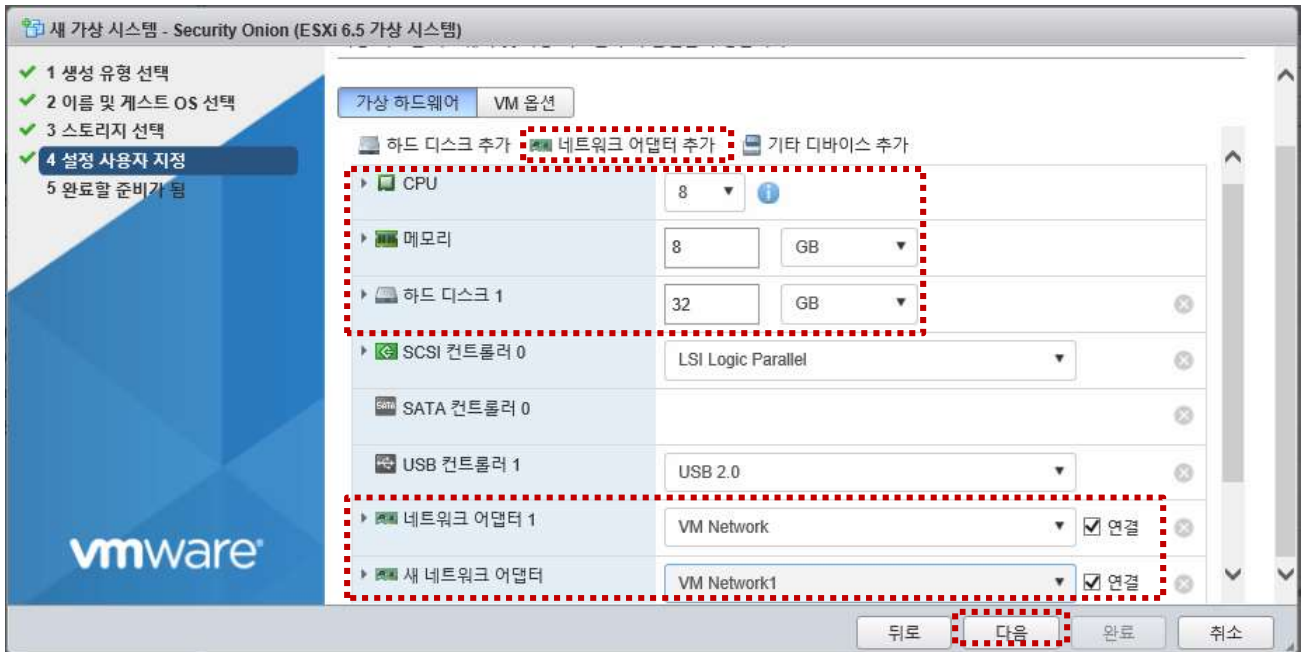
메모:

- ESXi 다운로드 주소: <https://my.vmware.com/en/web/vmware/evalcenter?p=free-esxi6>
- 디스크 이미지 굽기: Rufus 도구 사용 <https://rufus.akeo.ie/>
- Disk Imager <https://sourceforge.net/projects/win32diskimager/files/latest/download>
- USB 부팅 제가동은 전원 off/on (전원 케이블 포함)필요함
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① 네트워크 어댑터 추가
- ② CPU/메모리/하드디스크
- ③ 네트워크 어댑터 선택
- ④ 다음



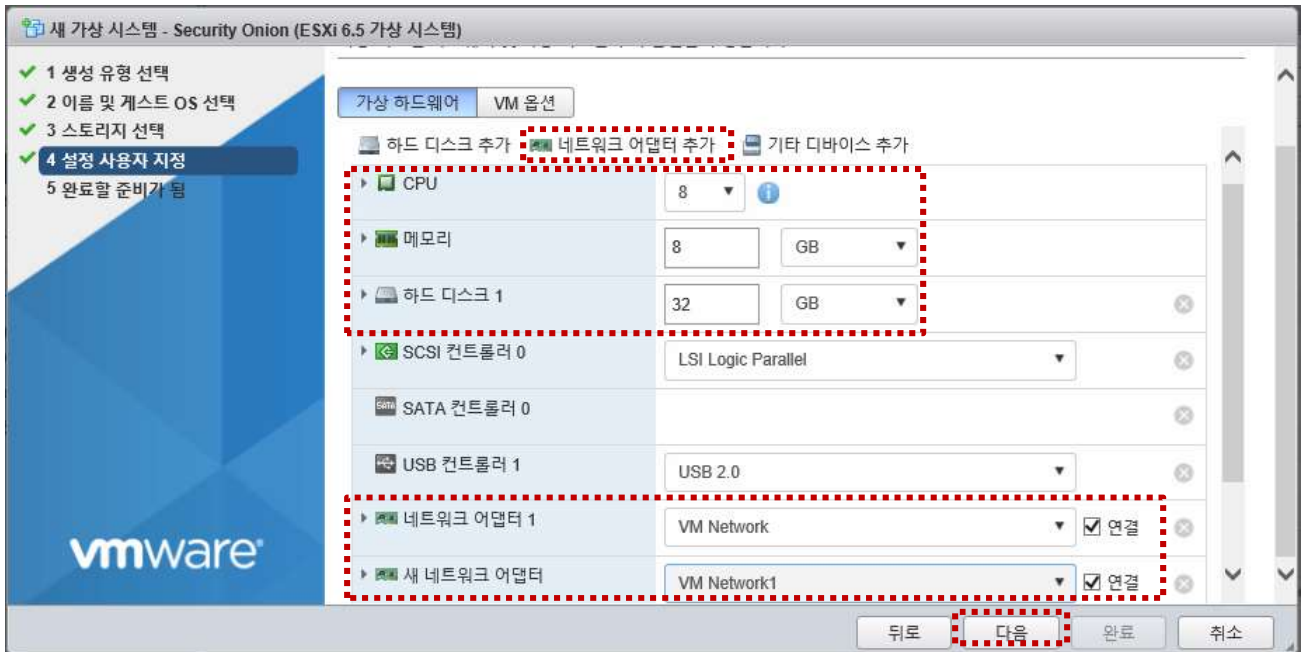
메모:

- 하드웨어 규격: <https://github.com/Security-Onion-Solutions/security-onion/wiki/Hardware>
- 최소 규격: RAM needed is 8GB
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① 네트워크 어댑터 추가
- ② CPU 8 Core / 메모리 8GB / 하드디스크 썸(Thin)
- ③ 네트워크 어댑터 선택
- ④ 다음



메모:

- 하드웨어 규격: <https://github.com/Security-Onion-Solutions/security-onion/wiki/Hardware>
- 최소 규격: RAM needed is 8GB
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md
- 스위치의 무작위 모드 확인 (미러링 효과)

8. vIDS

❖ Security Onion @ Hypervisor

① 설치 이미지 선택

새 가상 시스템 - Security Onion (ESXi 6.5 가상 시스템)

가상 하드웨어 VM 옵션

- USB 컨트롤러 1: USB 2.0
- 네트워크 어댑터 1: VM Network (연결)
- 새 네트워크 어댑터: VM Network1 (연결)
- CD/DVD 드라이브 1: 데이터스토어 ISO 파일 (전원을 켤 때 연결)
- CD/DVD 미디어: [datastore1] images/securityonion-16.04.4.1.iso

데이터스토어 브라우저

datastore1

- vmimages
 - images (선택됨)
 - securityonion-16.04.4.1.iso (1.73 GB, 2018년 6월 19일 화요일 10:00)

[datastore1] images/securityonion-16.04.4.1.iso

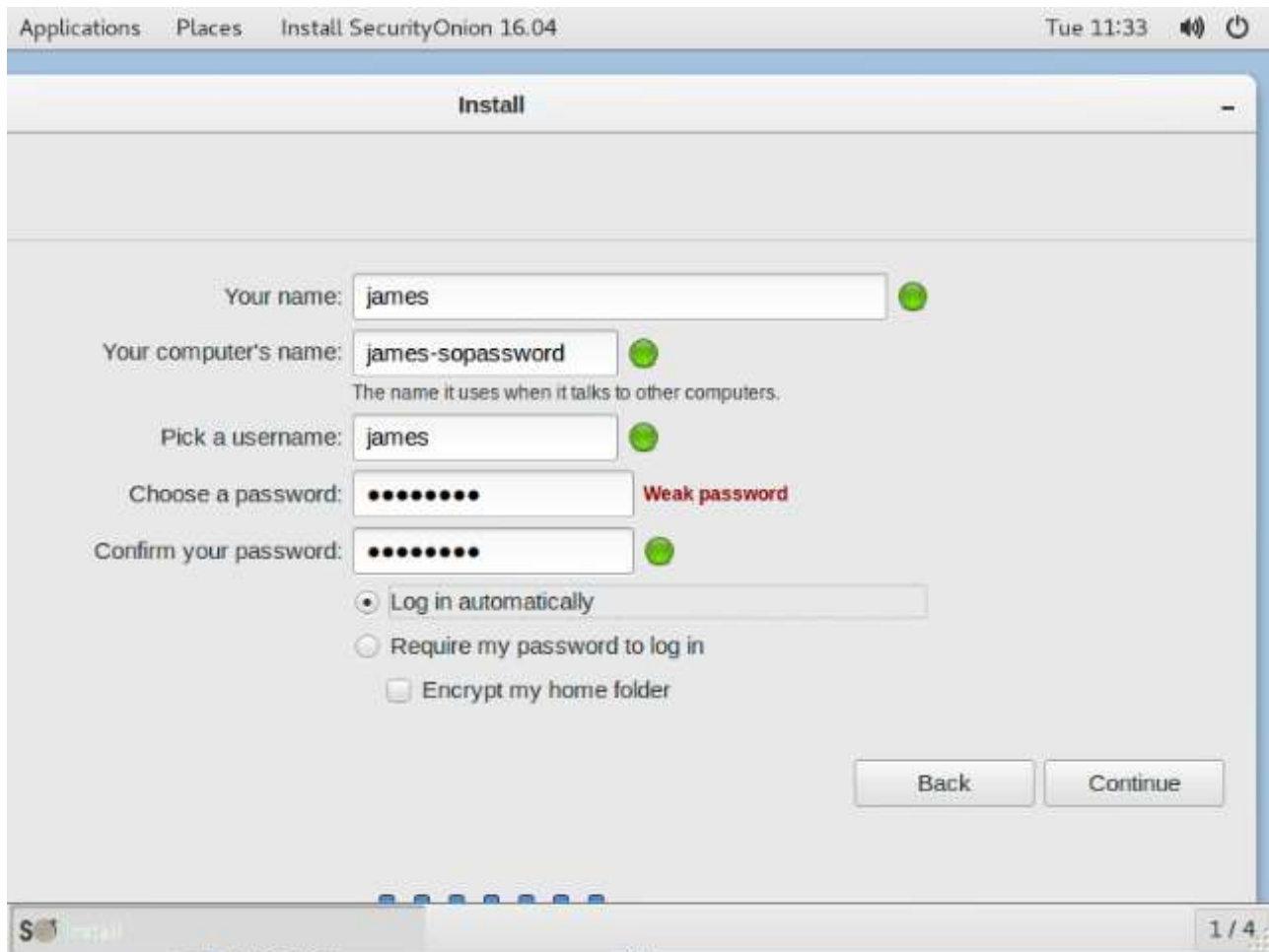
선택 취소

메모:

- 하드웨어 규격: <https://github.com/Security-Onion-Solutions/security-onion/wiki/Hardware>
- 최소 규격: RAM needed is 8GB
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor



메모:

- 하드웨어 규격: <https://github.com/Security-Onion-Solutions/security-onion/wiki/Hardware>
- 최소 규격: RAM needed is 8GB
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① 하이퍼바이저 확인
- ② 인터페이스 확인

The screenshot shows a Linux desktop with a terminal window and a 'Security Onion Setup' dialog box. The terminal displays the output of the 'ifconfig' command for three interfaces: ens160, ens192, and lo. The 'Security Onion Setup' dialog box asks 'Which network interface should be the management interface?' with radio buttons for ens160 and ens192. A table is overlaid on the terminal, showing network configuration details for two network adapters.

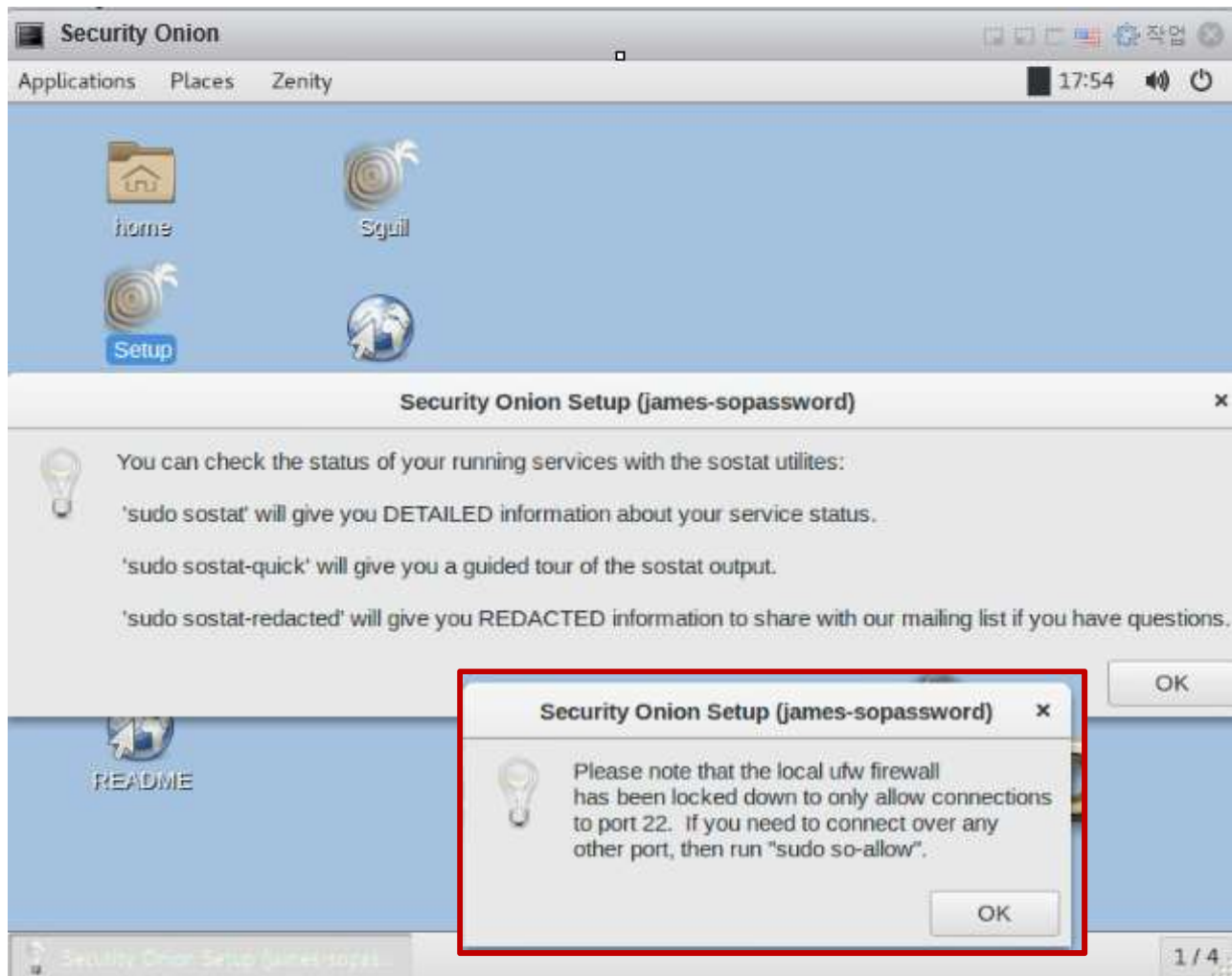
네트워크 어댑터 1	
네트워크	VM Network1 (연결됨)
연결됨	예
MAC 주소	00:0c:29:34:a8:92
패스스루(Direct-path I/O)	예
네트워크 어댑터 2	
네트워크	VM Network (연결됨)
연결됨	예
MAC 주소	00:0c:29:34:a8:9c
패스스루(Direct-path I/O)	예

메모:

8. vIDS

❖ Security Onion @ Hypervisor

- ① 설치 이미지 선택
- ② Restart
- ③ Setup



메모:

- 하드웨어 규격: <https://github.com/Security-Onion-Solutions/security-onion/wiki/Hardware>
- 최소 규격: RAM needed is 8GB
- https://github.com/Security-Onion-Solutions/security-onion/blob/master/Verify_ISO.md

8. vIDS

❖ Security Onion @ Hypervisor

- ① sudo apt get update @ Ubuntu Desktop
- ② Check event @ Security Onion

The image shows a Kibana dashboard and a terminal window. The Kibana dashboard displays three tables: Files - MIME Type, Files - Source IP Address, and Files - Destination IP Address. The terminal window shows the output of the 'sudo apt-get update' command and the 'ifconfig' command for the 'ens160' interface.

Files - MIME Type

MIME Type	Count
application/pkix-cert	151
application/ocsp-request	17
application/ocsp-response	17
text/plain	3
text/html	1

Files - Source IP Address

File IP Address	Count
54.230.181.25	24
192.168.10.100	17
52.85.118.205	12
54.230.181.131	12
117.18.237.29	9
54.201.6.28	8
54.68.128.184	8
172.217.26.14	6
54.200.76.177	6
54.230.182.26	6

Files - Destination IP Address

IP Address	Count
192.168.10.100	129
192.168.0.100	49
117.18.237.29	9
172.217.26.14	6
54.192.183.96	2

Terminal Output

```
James@james-virtual-machine: ~  
File Edit View Search Terminal Help  
James@james-virtual-machine:~$ sudo apt-get update  
Hit:1 http://kr.archive.ubuntu.com/ubuntu bionic InRelease  
Get:2 http://kr.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]  
Get:3 http://kr.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]  
Get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]  
Fetched 247 kB in 2s (141 kB/s)  
Reading package lists... Done  
James@james-virtual-machine:~$ ifconfig  
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
inet 192.168.10.100 netmask 255.255.255.0 broadcast 192.168.10.255  
inet6 fe80::1cc1:dceb:5420:b4d4 prefixlen 64 scopeid 0x20<link>  
ether 00:0c:29:d7:79:b6 txqueuelen 1000 (Ethernet)  
RX packets 106069 bytes 124995387 (124.9 MB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 49846 bytes 4690384 (4.6 MB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
inet 127.0.0.1 netmask 255.0.0.0  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1000 (Local Loopback)
```

NIDS - Alerts

alert.keyword	Count
Descending	8
ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management	

메모:

8. vIDS

❖ X-RDP for Security Onion (선택)

VM manager 사용하는 KVM/QEMU에서 마우스 인식 어려워 xrdp 설치

- ① **sudo ufw allow in 3389seconds**
- ② **sudo ufw allow ot 3389sword**
- ③ **sudo apt-get install xrdp**
- ④ **sudo apt-get install xfce4**
- ⑤ **sudo service xrdp restart**

- ⑥ **Remote Desktop 실행**

메모:

8. vIDS

❖ sudo so-allow for Remote Access

- ① sudo so-allow
- ② IP address for Remote Access

```
jslab@jslab-virtual-machine:~$ sudo so-allow
[sudo] password for jslab:
This program allows you to add a firewall rule to allow connections
from a new IP address.
```

What kind of device do you want to allow?

```
[a] - Analyst - ports 22/tcp, 443/tcp, and 7734/tcp
[b] - Logstash Beat - port 5044/tcp
[c] - apt-cacher-ng client - port 3142/tcp
[e] - Elasticsearch REST endpoint - port 9200
[f] - Logstash forwarder - standard - port 6050/tcp
[j] - Logstash forwarder - JSON - port 6051/tcp
[l] - Syslog device - port 514
[n] - Elasticsearch node-to-node communication - port 9300
[o] - OSSEC agent - port 1514
[s] - Security Onion sensor - 22/tcp, 4505/tcp, 4506/tcp, and
```

If you need to add any ports other than those listed above, you can do so using the standard 'ufw' utility.

For more information, please see the Firewall page on our Wiki <https://github.com/Security-Onion-Solutions/security-onion/wiki/Firewall>

Please enter your selection (a - analyst, c - apt-cacher-ng client, syslog, o - ossec, or s - Security Onion sensor, etc.):

```
a
Please enter the IP address of the analyst you'd like to allow
connect to port(s) 22, 443, 7734:
```

```
192.168.55.100
```

We're going to allow connections from 192.168.55.100 to port(s) 22, 443, 7734.

Here's the firewall rule we're about to add:
sudo ufw allow proto tcp from 192.168.55.100 to any port 22, 443, 7734

We're also whitelisting 192.168.55.100 in /var/ossec/etc/ossec.conf to prevent OSSEC Active Response from blocking it. Keep in mind, the OSSEC server will be restarted once configuration is complete.

To continue and add this rule, press Enter. Otherwise, press Ctrl-c to exit.

Rule added
Rule has been added.

Here is the entire firewall ruleset:

UFW Rules

To	Action	From
22/tcp	ALLOW	Anywhere
22, 443, 7734/tcp	ALLOW	192.168.55.122
22, 443, 7734/tcp	ALLOW	192.168.10.100
22, 443, 7734/tcp	ALLOW	192.168.55.100
22/tcp (v6)	ALLOW	Anywhere (v6)

Docker IPTables Rules

To	Action	From
----	--------	------

Added whitelist entry for 192.168.55.100 in /var/ossec/etc/ossec.conf.

Restarting OSSEC Server...
jslab@jslab-virtual-machine:~\$

메모:

8. vIDS

❖ Squert for Security Onion (선택)

1. **Squert is a web application that is used to query and view event data stored in a Sguil database (typically IDS alert data). Squert is a visual tool that attempts to provide additional context to events through the use of metadata, time series representations and weighted and logically grouped result sets.**
2. **Security Onion maintains its own fork of Squert**
3. **Squert authenticates against the Sguil user database, so you should be able to login to Squert using the same username/password you use to login to Sguil.**
4. **Data Type**
 - NIDS alerts
 - HIDS alerts
 - Asset data from PRADS (if PRADS and pads_agent are enabled)
 - HTTP logs from Bro (if http_agent is enabled)

메모:

- Squert: <http://www.squertproject.org/>
 - Security Onion maintains its own fork of Squert: <https://blog.securityonion.net/2016/09/squert-development.html>
-

8. vIDS

❖ sudo docker info

```
jslab@jslab-virtual-machine:~$ sudo docker info
Containers: 7
Running: 7
Paused: 0
Stopped: 0
Images: 7
Server Version: 18.06.1-ce
Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
Logging Driver: json-file
Cgroup Driver: cgroupfs
Plugins:
  Volume: local
  Network: bridge host macvlan null overlay
Log: awslogs fluentd gcplogs gelf journald json-file logentries splunk syslog
Swarm: inactive
Runtimes: runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 468a545b9edcd5932818eb9de8e72413e616e86e
runc version: 69663f0bd4b60df09991c08812a60108003fa340
init version: fec3683
Security Options:
  apparmor
  seccomp
  Profile: default
Kernel Version: 4.15.0-36-generic
Operating System: Ubuntu 16.04.5 LTS
OSType: linux
Architecture: x86_64
CPUs: 8
Total Memory: 11.73GiB
Name: jslab-virtual-machine
ID: UDLG:YGGR:VHYI:DNNS:3GER:63BY:KNR4:AIN4:EYA2:F6GY:VOXU:SYWZ
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): false
Registry: https://index.docker.io/v1/
Labels:
Experimental: false
Insecure Registries:
  127.0.0.0/8
Live Restore Enabled: false

WARNING: No swap limit support
```

메모:

- sudo docker version

8. vIDS

❖ sudo docker info

① sudo iptables -t nat -L -n

② sudo docker ps

```
jslab@jslab-virtual-machine:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
9872b6d8bc21       bridge             bridge             local
80a0d461c98d       host              host              local
8400d338e2a3       none              null              local
8d1ed97b634e       so-elastic-net    bridge            local
jslab@jslab-virtual-machine:~$
```

```
jslab@jslab-virtual-machine:~$ sudo docker ps
CONTAINER ID        IMAGE                                     COMMAND              CREATED            STATUS
PORTS              NAMES
92fd22d9e34d       securityionsolutions/so-curator         "/bin/bash"         11 hours ago      Up 11
hours                                                     so-curator
93764999e697       securityionsolutions/so-elastalert      "/opt/start-elastale... 11 hours ago      Up 11
hours                                                     so-elastalert
419f8db86c1e       securityionsolutions/so-kibana         "/bin/sh -c /usr/loc... 11 hours ago      Up 11
hours 127.0.0.1:5601->5601/tcp                                                     so-kibana
35fde0562d89       securityionsolutions/so-logstash       "/usr/local/bin/dock... 11 hours ago      Up 11
hours 0.0.0.0:5044->5044/tcp, 0.0.0.0:6050-6053->6050-6053/tcp, 0.0.0.0:9600->9600/tcp so-logstash
a541ecde19ef       securityionsolutions/so-elasticsearch  "/bin/bash bin/es-do... 11 hours ago      Up 11
hours 127.0.0.1:9200->9200/tcp, 127.0.0.1:9300->9300/tcp so-elasticsearch
c4fd232d54dc       securityionsolutions/so-domainstats    "/bin/sh -c '/usr/bi... 11 hours ago      Up 11
hours 20000/tcp                                                     so-domainstats
27e1571a4038       securityionsolutions/so-freqserver     "/bin/sh -c '/usr/bi... 11 hours ago      Up 11
hours 10004/tcp                                                     so-freqserver
jslab@jslab-virtual-machine:~$
```

메모:

8. vIDS

❖ sudo iptables -t nat -L -n

```
jslab@jslab-virtual-machine:~$ sudo iptables -t nat -L -n
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination            ADDRTYPE match dst-type LOCAL
DOCKER    all  --  0.0.0.0/0             0.0.0.0/0

Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination            ADDRTYPE match dst-type LOCAL
DOCKER    all  --  0.0.0.0/0             !127.0.0.0/8

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination            ADDRTYPE match dst-type LOCAL
MASQUERADE all  --  172.18.0.0/16         0.0.0.0/0
MASQUERADE all  --  172.17.0.0/16         0.0.0.0/0
MASQUERADE tcp  --  172.17.0.4            172.17.0.4            tcp dpt:9300
MASQUERADE tcp  --  172.17.0.4            172.17.0.4            tcp dpt:9200
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:9600
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:6053
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:6052
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:6051
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:6050
MASQUERADE tcp  --  172.17.0.5            172.17.0.5            tcp dpt:5044
MASQUERADE tcp  --  172.17.0.6            172.17.0.6            tcp dpt:5601

Chain DOCKER (2 references)
target     prot opt source                destination            ADDRTYPE match dst-type LOCAL
RETURN     all  --  0.0.0.0/0             0.0.0.0/0
RETURN     all  --  0.0.0.0/0             0.0.0.0/0
DNAT       tcp  --  0.0.0.0/0             127.0.0.1              tcp dpt:9300 to:172.17.0.4:9300
DNAT       tcp  --  0.0.0.0/0             127.0.0.1              tcp dpt:9200 to:172.17.0.4:9200
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:9600 to:172.17.0.5:9600
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:6053 to:172.17.0.5:6053
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:6052 to:172.17.0.5:6052
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:6051 to:172.17.0.5:6051
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:6050 to:172.17.0.5:6050
DNAT       tcp  --  0.0.0.0/0             0.0.0.0/0              tcp dpt:5044 to:172.17.0.5:5044
DNAT       tcp  --  0.0.0.0/0             127.0.0.1              tcp dpt:5601 to:172.17.0.6:5601
jslab@jslab-virtual-machine:~$
```

메모:

8. vIDS

❖ sudo iptables -L -n

```
jslab@jslab-virtual-machine:~$ sudo iptables -L -n
Chain INPUT (policy DROP)
target prot opt source destination
ufw-before-logging-input all -- 0.0.0.0/0 0.0.0.0/0
ufw-before-input all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-input all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-logging-input all -- 0.0.0.0/0 0.0.0.0/0
ufw-logging-input all -- 0.0.0.0/0 0.0.0.0/0
ufw-track-input all -- 0.0.0.0/0 0.0.0.0/0

Chain FORWARD (policy DROP)
target prot opt source destination
DOCKER-USER all -- 0.0.0.0/0 0.0.0.0/0
DOCKER-ISOLATION-STAGE-1 all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
DOCKER all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
DOCKER all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ufw-before-logging-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-before-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-logging-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-logging-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-reject-forward all -- 0.0.0.0/0 0.0.0.0/0
ufw-track-forward all -- 0.0.0.0/0 0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
ufw-before-logging-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-before-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-after-logging-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-logging-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-reject-output all -- 0.0.0.0/0 0.0.0.0/0
ufw-track-output all -- 0.0.0.0/0 0.0.0.0/0

Chain DOCKER (2 references)
target prot opt source destination
ACCEPT top -- 0.0.0.0/0 172.17.0.4 top dpt:9300
ACCEPT top -- 0.0.0.0/0 172.17.0.4 top dpt:9300
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:9600
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:6053
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:6052
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:6051
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:6050
ACCEPT top -- 0.0.0.0/0 172.17.0.5 top dpt:5044
ACCEPT top -- 0.0.0.0/0 172.17.0.6 top dpt:5041

Chain DOCKER-ISOLATION-STAGE-1 (1 reference)
target prot opt source destination
DOCKER-ISOLATION-STAGE-2 all -- 0.0.0.0/0 0.0.0.0/0
RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain DOCKER-ISOLATION-STAGE-2 (2 references)
target prot opt source destination
DROP all -- 0.0.0.0/0 0.0.0.0/0
DROP all -- 0.0.0.0/0 0.0.0.0/0
RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain DOCKER-USER (1 reference)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 state RELATED, ESTABLISHED
DROP all -- 0.0.0.0/0 0.0.0.0/0
RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-after-forward (1 reference)
target prot opt source destination

Chain ufw-after-input (1 reference)
target prot opt source destination
ufw-skip-to-policy-input udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:137
ufw-skip-to-policy-input udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:138
ufw-skip-to-policy-input tcp -- 0.0.0.0/0 0.0.0.0/0 top dpt:139
ufw-skip-to-policy-input tcp -- 0.0.0.0/0 0.0.0.0/0 top dpt:445
ufw-skip-to-policy-input udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:67
ufw-skip-to-policy-input udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:68
ufw-skip-to-policy-input all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type BROADCAST

Chain ufw-after-logging-forward (1 reference)
target prot opt source destination
LOG all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 10 LOG flags 0 level 4 prefix "[UFW BLOCK]"

Chain ufw-after-logging-input (1 reference)
target prot opt source destination
LOG all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 10 LOG flags 0 level 4 prefix "[UFW BLOCK]"

Chain ufw-after-logging-output (1 reference)
target prot opt source destination

Chain ufw-after-output (1 reference)
target prot opt source destination

Chain ufw-before-forward (1 reference)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 3
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 4
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 11
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 12
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 8
ufw-user-forward all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-before-input (1 reference)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
```

```
ufw-logging-deny all -- 0.0.0.0/0 0.0.0.0/0 ctstate INVALID
DROP all -- 0.0.0.0/0 0.0.0.0/0 ctstate INVALID
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 3
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 4
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 11
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 12
ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0 icmptype 8
ACCEPT udp -- 0.0.0.0/0 0.0.0.0/0 udp spt:67 dpt:68
ufw-not-local all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT udp -- 0.0.0.0/0 224.0.0.251 udp dpt:5353
ACCEPT udp -- 0.0.0.0/0 239.255.255.250 udp dpt:1900
ufw-user-input all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-before-logging-forward (1 reference)
target prot opt source destination

Chain ufw-before-logging-input (1 reference)
target prot opt source destination

Chain ufw-before-logging-output (1 reference)
target prot opt source destination

Chain ufw-before-output (1 reference)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 ctstate RELATED,ESTABLISHED
ufw-user-output all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-logging-allow (0 references)
target prot opt source destination
LOG all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 10 LOG flags 0 level 4 prefix "[UFW ALLOW]"

Chain ufw-logging-deny (2 references)
target prot opt source destination
RETURN all -- 0.0.0.0/0 0.0.0.0/0 ctstate INVALID limit: avg 3/min burst 10
LOG all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 10 LOG flags 0 level 4 prefix "[UFW BLOCK]"

Chain ufw-not-local (1 reference)
target prot opt source destination
RETURN all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type LOCAL
RETURN all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type MULTICAST
RETURN all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type BROADCAST
ufw-logging-deny all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 10
DROP all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-reject-forward (1 reference)
target prot opt source destination

Chain ufw-reject-input (1 reference)
target prot opt source destination

Chain ufw-reject-output (1 reference)
target prot opt source destination

Chain ufw-skip-to-policy-forward (0 references)
target prot opt source destination
DROP all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-skip-to-policy-input (7 references)
target prot opt source destination
DROP all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-skip-to-policy-output (0 references)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-track-forward (1 reference)
target prot opt source destination

Chain ufw-track-input (1 reference)
target prot opt source destination

Chain ufw-track-output (1 reference)
target prot opt source destination
ACCEPT top -- 0.0.0.0/0 0.0.0.0/0 ctstate NEW
ACCEPT udp -- 0.0.0.0/0 0.0.0.0/0 ctstate NEW

Chain ufw-user-forward (1 reference)
target prot opt source destination

Chain ufw-user-input (3 references)
target prot opt source destination
ACCEPT top -- 0.0.0.0/0 0.0.0.0/0 top dpt:22
ACCEPT top -- 192.168.55.122 0.0.0.0/0 multiport dports 22,443,7734
ACCEPT top -- 192.168.10.100 0.0.0.0/0 multiport dports 22,443,7734
ACCEPT top -- 192.168.55.100 0.0.0.0/0 multiport dports 22,443,7734

Chain ufw-user-limit (0 references)
target prot opt source destination
LOG all -- 0.0.0.0/0 0.0.0.0/0 limit: avg 3/min burst 5 LOG flags 0 level 4 prefix "[UFW LIMIT BLOCK]"
REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-port-unreachable

Chain ufw-user-limit-accept (0 references)
target prot opt source destination
ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

Chain ufw-user-logging-forward (0 references)
target prot opt source destination

Chain ufw-user-logging-input (0 references)
target prot opt source destination

Chain ufw-user-logging-output (0 references)
target prot opt source destination

Chain ufw-user-output (1 reference)
target prot opt source destination
jslab@jslab-virtual-machine:~$
```

메모:

8. vIDS

❖ ip route

- ① ip route
- ② brctl show
- ③ Check 'sudo docker network ls' # 도커의 리눅스 브릿지 사용

```
jslab@jslab-virtual-machine:~$ ip route
default via 192.168.55.1 dev ens224 onlink
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1
172.18.0.0/16 dev br-8d1ed97b634e proto kernel scope link src 172.18.0.1
192.168.55.0/24 dev ens224 proto kernel scope link src 192.168.55.43
jslab@jslab-virtual-machine:~$v
```

```
jslab@jslab-virtual-machine:~$ brctl show
bridge name      bridge id        STP enabled     interfaces
br-8d1ed97b634e  8000.02429b7f90e0  no              veth0a8d905
                                                           veth2fc6972
                                                           veth3b98e4f
                                                           veth5284a6f
                                                           veth783c90b
                                                           veth7a5200b
                                                           vethcdb21af
docker0          8000.0242d38891bc  no              veth4021b3b
                                                           veth591b8ce
                                                           veth7ef17b0
                                                           veth8d500af
                                                           vetha1d41ca
                                                           vethbc57b2b
                                                           vethebda422
```

```
jslab@jslab-virtual-machine:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
9872b6d8bc21       bridge             bridge              local
80a0d461c98d       host               host                local
8400d338e2a3       none               null                local
8d1ed97b634e       so-elastic-net     bridge              local
jslab@jslab-virtual-machine:~$
```

메모:

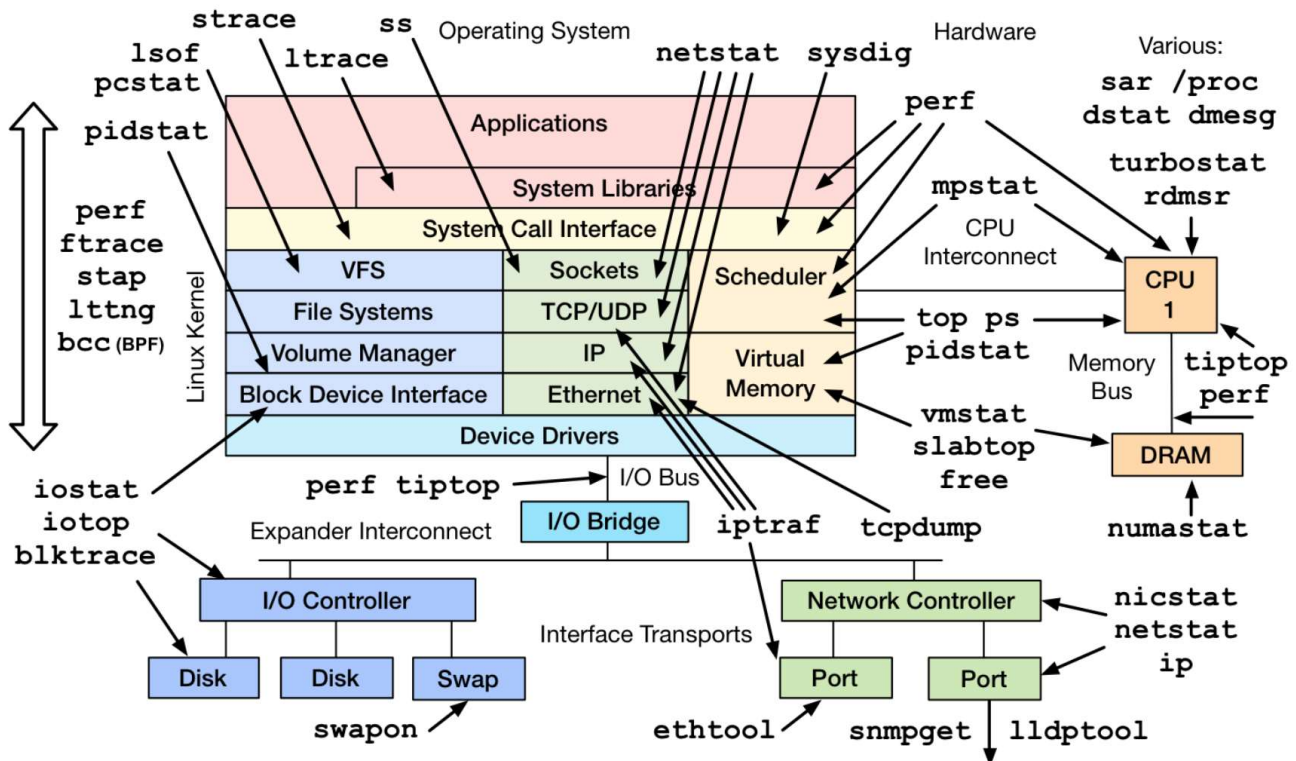
8. vIDS

❖ Linux Perf Tools

① `sudo apt-get install nmh`

② **Fedora:** 다음쪽 참조

- `bash <(curl -Ss https://my-netdata.io/kickstart.sh)`



메모:

- 컨테이너 고려사항
 - ✓ **Netdata 등의 트래픽 변동 확인** <http://127.0.0.1:19999/>
 - ✓ Host PID는 컨테이너 ID와 연동하지 않음
 - ✓ 커널에 컨테이너 ID 표시 없음
 - ✓ 리눅스 도구에 컨테이너를 위한 설명 없음 (netdata는 docker0 트래픽 통계 확인 가능)

8. vIDS

❖ sudo docker network ls & brctl show

1. sudo docker network ls
2. 'brctl show' & 'virsh net-list --all'

```
jslab@jslab-virtual-machine:~$ ip route
default via 192.168.55.1 dev ens224 onlink
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1
172.18.0.0/16 dev br-8d1ed97b634e proto kernel scope link src 172.18.0.1
192.168.55.0/24 dev ens224 proto kernel scope link src 192.168.55.43
jslab@jslab-virtual-machine:~$
```

```
jslab@jslab-virtual-machine:~$ brctl show
bridge name      bridge id        STP enabled     interfaces
br-8d1ed97b634e  8000.02429b7f90e0  no              veth0a8d905
                                                         veth2fc6972
                                                         veth3b98e4f
                                                         veth5284a6f
                                                         veth783c90b
                                                         veth7a5200b
                                                         vethcdb21af
docker0          8000.0242d38891bc  no              veth4021b3b
                                                         veth591b8ce
                                                         veth7ef17b0
                                                         veth8d500af
                                                         vetha1d41ca
                                                         vethbc57b2b
                                                         vethebda422
```

```
jslab@jslab-virtual-machine:~$ sudo virsh net-list --all
Name              State    Autostart  Persistent
-----
default           active   yes        yes
jslab@jslab-virtual-machine:~$
```

메모:

- virsh is a command line interface tool for managing guests and the hypervisor

8. vIDS

❖ brctl showmacs docker0

① brctl showmacs docker0

```
jslab@jslab-virtual-machine:~$ brctl showmacs docker0
port no mac addr          is local?    ageing timer
 4    02:42:ac:11:00:05      no           0.90
 5    02:42:ac:11:00:06      no           16.37
 6    06:e9:55:0d:c7:4a      yes           0.00
 6    06:e9:55:0d:c7:4a      yes           0.00
 1    42:2c:60:88:9a:65      yes           0.00
 1    42:2c:60:88:9a:65      yes           0.00
 3    4e:b4:78:52:47:4b      yes           0.00
 3    4e:b4:78:52:47:4b      yes           0.00
 2    7a:02:82:10:c9:70      yes           0.00
 2    7a:02:82:10:c9:70      yes           0.00
 4    82:f5:84:ad:6b:f5      yes           0.00
 4    82:f5:84:ad:6b:f5      yes           0.00
 7    a6:f3:3a:e2:05:6f      yes           0.00
 7    a6:f3:3a:e2:05:6f      yes           0.00
 5    e2:d3:a5:2f:33:52      yes           0.00
 5    e2:d3:a5:2f:33:52      yes           0.00
jslab@jslab-virtual-machine:~$
```

메모:

- Network : bridge(bridge), host(host), none(null), so-elastic-net(bridge)
-

8. vIDS

❖ sudo docker network inspect bridge

```
jslab@jslab-virtual-machine:~$ sudo docker network inspect bridge
[sudo] password for jslab:
[
  {
    "Name": "bridge",
    "Id": "9872b6d8bc21ff54e2d204efd323ba6ef8791f497598c15d115bbe1cb477826",
    "Created": "2018-11-18T19:35:19.141183196Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "172.17.0.0/16",
          "Gateway": "172.17.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {
      "27e1571a40380bbc40c2db9161e65b666cac2e48b1c01020799a563c7c767163": {
        "Name": "so-freqserver",
        "EndpointID": "93d76dc8ad119431f94e1c179586a6d1495e7db487a1fd3bf7e4036241f15555",
        "MacAddress": "02:42:ac:11:00:02",
        "IPv4Address": "172.17.0.2/16",
        "IPv6Address": ""
      },
      "35fde0562d899b0d6cd0e7ac820dfd73a60bde230c170210f35cea7c255b5131": {
        "Name": "so-logstash",
        "EndpointID": "c7678ac47cbb18705e788c3e0652ef7749451a01e7db96c21d66484f1217bea3",
        "MacAddress": "02:42:ac:11:00:05",
        "IPv4Address": "172.17.0.5/16",
        "IPv6Address": ""
      },
      "419f8db86c1ed5fccc28d0a41fb4f059c54dcbf038c3d7056e59956bfa3c0a3c": {
        "Name": "so-kibana",
        "EndpointID": "79bb394b8a8538a1f409e0f4e59ab442c26ab77b5173ce00d172cbf0c98d8f35",
        "MacAddress": "02:42:ac:11:00:06",
        "IPv4Address": "172.17.0.6/16",
        "IPv6Address": ""
      },
      "92fd22d9e34ddd90af681aae94dd9e6d6ad208b4df87eb3e549d83064a5371d": {
        "Name": "so-curator",
        "EndpointID": "6b8bc81f87fc1f44a7f5e8e0a87309e7fa5a16e0295f78f73935801ab17454a",
        "MacAddress": "02:42:ac:11:00:08",
        "IPv4Address": "172.17.0.8/16",
        "IPv6Address": ""
      },
      "93764999e6975359edeb3a03ca8908b5358b640cbbc21b5793b57463dbfbf7c0": {
        "Name": "so-elastalert",
        "EndpointID": "8469cb9216c93abc9f4cc790511bce46870eda81785a2250f2bd06c39d67f7a",
        "MacAddress": "02:42:ac:11:00:07",
        "IPv4Address": "172.17.0.7/16",
        "IPv6Address": ""
      },
      "a541ecde19ef87b73595305e0958d7208458d745eb66ac5ee66d07d084d8e0b6": {
        "Name": "so-elasticsearch",
        "EndpointID": "e0dc9a32ad25a1f66e73fc5bfb608e7d634ded6a5aad60ae135e99ccd505a58d",
        "MacAddress": "02:42:ac:11:00:04",
        "IPv4Address": "172.17.0.4/16",
        "IPv6Address": ""
      },
      "c4fd232d54dc571b105c5909742dda1f32635e65bbc85c065ddf536d104dfd6": {
        "Name": "so-domainstats",
        "EndpointID": "cd0e6530e299debdb3ee1a15860438298d942c1200634e2d16b1ef46",
        "MacAddress": "02:42:ac:11:00:03",
        "IPv4Address": "172.17.0.3/16",
        "IPv6Address": ""
      }
    },
    "Options": {
      "com.docker.network.bridge.default_bridge": "true",
      "com.docker.network.bridge.enable_icc": "true",
      "com.docker.network.bridge.enable_ip_masquerade": "true",
      "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
      "com.docker.network.bridge.name": "docker0",
      "com.docker.network.driver.mtu": "1500"
    },
    "Labels": {}
  }
]
```

메모:

- Containers @ Bridge : so-curator (172.17.0.8/16), so-elastalert (172.17.0.7/16), so-kibana (172.17.0.6/16), so-logstash (172.17.0.5/16), so-elasticsearch (172.17.0.4/16), so-domainstats (172.17.0.3/16), so-freqserver (172.17.0.2/16)

8. vIDS

❖ sudo docker network inspect so-elastic-net

```
jslab@jslab-virtual-machine:~$ sudo docker network inspect so-elastic-net
[
  {
    "Name": "so-elastic-net",
    "Id": "8d1ed97b634e480a725e4033b3de3c2e382765b5f1274daf67533281c0070a",
    "Created": "2018-11-18T19:38:51.794605818Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": {},
      "Config": [
        {
          "Subnet": "172.18.0.0/16",
          "Gateway": "172.18.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {
      "27e1571a40380bbc40c2db9161e65b666cac2e48b1c01020799a563c7c767163": {
        "Name": "so-freqserver",
        "EndpointID": "5443b5e27f18c283aac673a5158c680f1cd16ab7473b96d43ac2cf643d78f488",
        "MacAddress": "02:42:ac:12:00:02",
        "IPv4Address": "172.18.0.2/16",
        "IPv6Address": ""
      },
      "35fde0562d899b0d6cd0e7ac820dfd73a60bde230c170210f35cea7c255b5131": {
        "Name": "so-logstash",
        "EndpointID": "f0dfcc32661fc7fb04e8f5a08f24cbe4d1545f6b82b56375990a4367dbd417a3",
        "MacAddress": "02:42:ac:12:00:05",
        "IPv4Address": "172.18.0.5/16",
        "IPv6Address": ""
      },
      "419f8db86c1ed5fccc28d0a41fb4f059c54dcbf038c3d7056e59956bfa30a3c": {
        "Name": "so-kibana",
        "EndpointID": "1af0dd03cc5aadd4ea15f55758e752b1ae45b2fd9f8f647c43c6447259333bab",
        "MacAddress": "02:42:ac:12:00:06",
        "IPv4Address": "172.18.0.6/16",
        "IPv6Address": ""
      },
      "92fd22d9e34ddc90af681aae94dd9e6d6ad208b4df7eb3e549d83064a5371d": {
        "Name": "so-curator",
        "EndpointID": "ed12e89b81146df9beeb46156d778a2554fa8f0bedc43be7956f56c6c56e009b",
        "MacAddress": "02:42:ac:12:00:08",
        "IPv4Address": "172.18.0.8/16",
        "IPv6Address": ""
      },
      "93764999e6975359edeb3a03ca8908b5358b640cbbc21b5793b57463bdfbf7c0": {
        "Name": "so-elastalert",
        "EndpointID": "3d308fafa699188adce990fdd9b4d13aa30bc38595e19b93b3fa286962b7387a",
        "MacAddress": "02:42:ac:12:00:07",
        "IPv4Address": "172.18.0.7/16",
        "IPv6Address": ""
      },
      "a541ecde19ef87b73595305e0958d7208458d745eb66ac5ee66d07d084d8e0b6": {
        "Name": "so-elasticsearch",
        "EndpointID": "ed4125e7734b9c3f46adc328fa8edc61c72f18ec644b605708590893c25512f6",
        "MacAddress": "02:42:ac:12:00:04",
        "IPv4Address": "172.18.0.4/16",
        "IPv6Address": ""
      },
      "c4fd232d54dc571b105c5909742dda1f32635e65bbc85c065ddef536d104dfd6": {
        "Name": "so-domainstats",
        "EndpointID": "4384c3eeee12b13bdf61f9b88b92bf1a58e5ba8988f4076333a87080a2ce6584",
        "MacAddress": "02:42:ac:12:00:03",
        "IPv4Address": "172.18.0.3/16",
        "IPv6Address": ""
      }
    }
  }
],
  "Options": {},
  "Labels": {}
}
```

메모:

- Containers @ so-elastic-net (bridge) : so-curator (172.18.0.8/16), so-elastalert (172.18.0.7/16), so-kibana (172.18.0.6/16), so-logstash (172.17.0.5/16), so-elasticsearch (172.18.0.4/16), so-domainstats (172.18.0.3/16), so-freqserver (172.18.0.2/16)

8. vIDS

❖ ifconfig & ip show link

① ifconfig

② ip link show

```
jslab@jslab-virtual-machine:~$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: ens160: <BROADCAST,MULTICAST,NOARP,PROMISC,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default
   link/ether 00:0c:29:2e:ea:ea brd ff:ff:ff:ff:ff:ff
3: ens192: <BROADCAST,MULTICAST,NOARP,PROMISC,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default
   link/ether 00:0c:29:2e:ea:f4 brd ff:ff:ff:ff:ff:ff
4: ens224: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
   link/ether 00:0c:29:2e:ea:fe brd ff:ff:ff:ff:ff:ff
5: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group default
   link/ether 02:42:d3:82:91:bc brd ff:ff:ff:ff:ff:ff
6: br-8d1ed97b634e: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group default
   link/ether 02:42:9b:7f:90:e0 brd ff:ff:ff:ff:ff:ff
8: veth4021b3b@if7: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether 42:2c:60:88:9a:65 brd ff:ff:ff:ff:ff:ff link-netnsid 0
10: vethcdb21af@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether b6:35:0c:7e:7d:93 brd ff:ff:ff:ff:ff:ff link-netnsid 0
12: veth591b8ce@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether 7a:02:82:10:c9:70 brd ff:ff:ff:ff:ff:ff link-netnsid 1
14: veth3b98e4f@if13: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether ca:ee:22:8b:76:41 brd ff:ff:ff:ff:ff:ff link-netnsid 1
16: veth1d41ca@if15: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether 4e:b4:78:52:47:4b brd ff:ff:ff:ff:ff:ff link-netnsid 2
18: veth783c90b@if17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether 8e:c0:e2:f7:dc:cc brd ff:ff:ff:ff:ff:ff link-netnsid 2
20: veth8d500af@if19: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether 82:f5:84:ad:6b:f5 brd ff:ff:ff:ff:ff:ff link-netnsid 3
22: veth0a8d905@if21: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether 82:6f:c2:ce:cb:aa brd ff:ff:ff:ff:ff:ff link-netnsid 3
24: vethef17b0@if23: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether e2:d3:a5:2f:33:52 brd ff:ff:ff:ff:ff:ff link-netnsid 4
26: veth2fc6972@if25: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether be:9a:dd:51:46:c7 brd ff:ff:ff:ff:ff:ff link-netnsid 4
28: vethbc57b2b@if27: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether 06:e9:55:0d:c7:4a brd ff:ff:ff:ff:ff:ff link-netnsid 5
30: veth7a5200b@if29: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether 42:a8:33:59:67:d9 brd ff:ff:ff:ff:ff:ff link-netnsid 5
32: vethebda422@if31: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP mode DEFAULT
   link/ether a6:f3:3a:e2:05:6f brd ff:ff:ff:ff:ff:ff link-netnsid 6
34: veth5284a6f@if33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-8d1ed97b634e state UP mode
   link/ether ca:1a:2d:bd:2b:bb brd ff:ff:ff:ff:ff:ff link-netnsid 6
35: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN mode DEFAULT group default
   link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
36: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc pfifo_fast state DOWN mode DEFAULT group default qlen 1000
   link/ether 52:54:00:a8:82:63 brd ff:ff:ff:ff:ff:ff
jslab@jslab-virtual-machine:~$
```

```
jslab@jslab-virtual-machine:~$ ifconfig
ens160: Link encap: Ethernet  HWaddr 00:0c:29:2e:ea:ea
   inet addr: 192.168.1.4  Bcast:192.168.1.255  Mask:255.255.0.0
   inet6 addr: fe80::c29:2e:ea:ea:0:0  PrefixLen:64
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
   RX packets: 1514 errors: 0 dropped: 0 overruns: 0 carrier: 0
   TX packets: 4232 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 4252618 (4.1 MiB)  TX bytes: 4252641 (4.1 MiB)

ens192: Link encap: Ethernet  HWaddr 00:0c:29:2e:ea:f4
   inet addr: 192.168.1.2  Bcast:192.168.1.255  Mask:255.255.0.0
   inet6 addr: fe80::c29:2e:ea:f4:0:0  PrefixLen:64
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
   RX packets: 1514 errors: 0 dropped: 0 overruns: 0 carrier: 0
   TX packets: 4232 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 4252618 (4.1 MiB)  TX bytes: 4252641 (4.1 MiB)

ens224: Link encap: Ethernet  HWaddr 00:0c:29:2e:ea:fe
   inet addr: 192.168.1.42  Bcast:192.168.1.255  Mask:255.255.0.0
   inet6 addr: fe80::c29:2e:ea:fe:0:0  PrefixLen:64
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
   RX packets: 1514 errors: 0 dropped: 0 overruns: 0 carrier: 0
   TX packets: 4232 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 4252618 (4.1 MiB)  TX bytes: 4252641 (4.1 MiB)

lo: Link encap: Local Loopback
   inet addr: 127.0.0.1  Mask:255.0.0.0
   inet6 addr: ::1  PrefixLen:64  Metric:1
   UP LOOPBACK RUNNING  MTU:65536  Metric:1
   RX packets: 2200 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 2200 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 176800 (0.17 MiB)  TX bytes: 176800 (0.17 MiB)

veth4021b3b@if7: Link encap: Ethernet  HWaddr 42:2c:60:88:9a:65
   inet addr: 192.168.1.100  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth591b8ce@if11: Link encap: Ethernet  HWaddr 7a:02:82:10:c9:70
   inet addr: 192.168.1.101  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth3b98e4f@if13: Link encap: Ethernet  HWaddr ca:ee:22:8b:76:41
   inet addr: 192.168.1.102  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth1d41ca@if15: Link encap: Ethernet  HWaddr 4e:b4:78:52:47:4b
   inet addr: 192.168.1.103  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth783c90b@if17: Link encap: Ethernet  HWaddr 8e:c0:e2:f7:dc:cc
   inet addr: 192.168.1.104  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth8d500af@if19: Link encap: Ethernet  HWaddr 82:f5:84:ad:6b:f5
   inet addr: 192.168.1.105  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth0a8d905@if21: Link encap: Ethernet  HWaddr 82:6f:c2:ce:cb:aa
   inet addr: 192.168.1.106  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

vethef17b0@if23: Link encap: Ethernet  HWaddr e2:d3:a5:2f:33:52
   inet addr: 192.168.1.107  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth2fc6972@if25: Link encap: Ethernet  HWaddr be:9a:dd:51:46:c7
   inet addr: 192.168.1.108  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

vethbc57b2b@if27: Link encap: Ethernet  HWaddr 06:e9:55:0d:c7:4a
   inet addr: 192.168.1.109  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth7a5200b@if29: Link encap: Ethernet  HWaddr 42:a8:33:59:67:d9
   inet addr: 192.168.1.110  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

vethebda422@if31: Link encap: Ethernet  HWaddr a6:f3:3a:e2:05:6f
   inet addr: 192.168.1.111  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

veth5284a6f@if33: Link encap: Ethernet  HWaddr ca:1a:2d:bd:2b:bb
   inet addr: 192.168.1.112  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:0
   RX packets: 100 errors: 0 dropped: 0 overruns: 0 frame: 0
   TX packets: 100 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 40960 (40.0 KiB)  TX bytes: 40960 (40.0 KiB)

virbr0: Link encap: Ethernet  HWaddr 00:00:00:00:00:00
   inet addr: 192.168.1.1  Bcast:192.168.1.255  Mask:255.255.0.0
   UP BROADCAST MULTICAST  MTU:1500  Metric:1
   RX packets: 0 errors: 0 dropped: 0 overruns: 0 carrier: 0
   TX packets: 0 errors: 0 dropped: 0 overruns: 0 carrier: 0
   RX bytes: 0 (0.0 B)  TX bytes: 0 (0.0 B)

jslab@jslab-virtual-machine:~$
```

메모:

- Images: so-curator, so-elastalert, so-kibana, so-logstash, so-elasticsearch, so-domainstats, so-freqserver (@ /securityonionsolutions/)

8. vIDS

❖ sudo docker image ls

① sudo docker image ls

② sudo docker image inspect c6

```
jslab@jslab-virtual-machine:~$ sudo docker image ls
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
securityonionsolutions/so-freqserver  latest      7430335b16aa  2 months ago  312MB
securityonionsolutions/so-domainstats latest      0497f0bbe842  2 months ago  400MB
securityonionsolutions/so-elastalert  latest      0ee1d4814674  2 months ago  418MB
securityonionsolutions/so-curator     latest      c1e5b6c06aad  2 months ago  324MB
securityonionsolutions/so-kibana      latest      ce42f28e58ab  2 months ago  800MB
securityonionsolutions/so-logstash    latest      c6f488b28175  2 months ago  708MB
securityonionsolutions/so-elasticsearch latest      862bec843f98  2 months ago  432MB
```

메모:

- Images: so-curator, so-elastalert, so-kibana, so-logstash, so-elasticsearch, so-domainstats, so-freqserver (@ /securityonionsolutions/)
-

8. vIDS

❖ sudo docker image inspect d9

```
james@x2korn:~$ sudo docker inspect d9
[
  {
    "Id": "sha256:d9c0d195505a5eaf07aeeefb0367ab27d6c34c8b55530ff69a047ea",
    "RootFS": {
      "Type": "overlay2",
      "Layers": [
        "sha256:09b4a727b085c1d38e0e992710902240ef91ef603746f8a2c4f2992142fa",
        "sha256:c21691f4e4209419507ca3a00010d816d8097750d7a30351020c719c1a",
        "sha256:52002c7fd1e11503398a794994c9c3a70c0e54c4c9216071819a78e0a85",
        "sha256:1902089e07545780a0f4c4c32a105d0802d14c410118a972e0a8ba",
        "sha256:11003c10f4a0000a723004f24e70707070116604d800845310a09",
        "sha256:c01026e1817e96c727d4f1e4d0c0c355f8e81c09175d80d051122001a"
      ]
    },
    "Metadata": {
      "LastUpdateTime": "2018-04-16T14:21:01.093380059-04:00"
    }
  }
]
james@x2korn:~$
```

```

"ContainerConfig": {
  "Hostname": "24006fe709ed",
  "Domainname": "",
  "User": "freqserver",
  "AttachStdin": false,
  "AttachStdout": false,
  "AttachStderr": false,
  "ExposedPorts": {
    "10004/tcp": {}
  },
  "Tty": false,
  "OpenStdin": false,
  "StdinOnce": false,
  "Env": [
    "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
  ],
  "Cmd": [
    "/bin/sh",
    "-c",
    "#(nop)",
    "CMD [\"/bin/sh\" \"-c\" \"/usr/bin/python /opt/freq_server/freq/freq_table_freq.py -ip 0.0.0.0 10004 /opt/freq_server/freq/freq_table_freq.py\"]"
  ],
  "ArgsEscaped": true,
  "Image": "sha256:09b4a727b085c1d38e0e992710902240ef91ef603746f8a2c4f2992142fa",
  "Volumes": null,
  "WorkingDir": "",
  "Entrypoint": null,
  "OnBuild": null,
  "Labels": {
    "build-date": "20180302",
    "license": "GPLv2",
    "maintainer": "Security Onion Solutions, LLC",
    "name": "Security Base Image",
    "vendor": "CentOS"
  },
  "StopSignal": "SIGTERM"
}
"Architecture": "amd64",
"Os": "linux",
"Size": 84470968,
"VirtualSize": 364707066,
"GraphDriver": {
  "Data": null,
  "Name": "aufs"
},
"RootFS": {
  "Type": "layers",
  "Layers": [
    "sha256:002095540b7956d32a5ef4205c059b050a01d84440400c7a7919841884",
    "sha256:09b4a727b085c1d38e0e992710902240ef91ef603746f8a2c4f2992142fa",
    "sha256:c21691f4e4209419507ca3a00010d816d8097750d7a30351020c719c1a",
    "sha256:52002c7fd1e11503398a794994c9c3a70c0e54c4c9216071819a78e0a85",
    "sha256:1902089e07545780a0f4c4c32a105d0802d14c410118a972e0a8ba",
    "sha256:11003c10f4a0000a723004f24e70707070116604d800845310a09",
    "sha256:c01026e1817e96c727d4f1e4d0c0c355f8e81c09175d80d051122001a"
  ]
},
"Metadata": {
  "LastUpdateTime": "2018-04-16T14:21:01.093380059-04:00"
}
}
```

- 메모:**
- X-RDP for Security Onion

8. vIDS

❖ netdata / ntopng / sshd / Net-tools (Ubuntu 17.20)

1. netdata

- `bash <(curl -Ss https://my-netdata.io/kickstart.sh)`
- `http://127.0.0.1:19999/`

2. ntopng (sidekick)

- `sudo docker run -t -i -d --net=host --name ntopng
lucaderi/ntopng-docker`

메모:

- 다운로드 주소:
 - ✓ https://github.com/Security-Onion-Solutions/security-onion/blob/master/old/Verify_ISO_14.04.5.2.md
 - ✓ <https://github.com/Security-Onion-Solutions/security-onion/releases/download/v14.04.5.3/securityonion-14.04.5.3.iso>
-

8. vIDS

❖ Side-Kick

- **sudo docker run -t -i -d --net=host --name ntopng2 lucaderi/ntopng-docker**

메모:

- `sudo docker run -t -i -d -p 3331:3000 --name ntopng1 lucaderi/ntopng-docker`
-