
2020 개인 프로젝트


Linux, Windows Server 비교

Linux, Windows Server

각 서버의 특징 및 설치, 비교

이름
이정근


이메일
mung0001@naver.com

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


개 정 이 력

개정 번호	개정 내용 요약	추가/수정 항목	개정 일자
1.0	최초 제정 승인	시작	2020.08.10
2.0	추가 수정 승인	각 서버의 특징	2020.09.02
		설치와 부팅	
		네트워크 설정	
		사용자 그룹 설정	
		마운트, 파티션 설정	
		Backup, Restore 설정	
3.0	추가 수정 승인	FTP SERVER	2020.09.12
		TELNET SERVER	
		SSH SERVER	
		NTP SERVER	
3.1	추가 수정 승인	DHCP SERVER	2020.09.25
		DNS SERVER	
3.2	추가 수정 승인	WEB SERVER	2020.10.20
3.3	추가 수정 승인	MAIL SERVER	2020.11.12


문 서 규 칙

	Report	Version	Last Modified	④
	①	②	③	

- 작성은 Microsoft Word Office 365(v.1908)으로 작성되었으며, 저장시 PDF 형식으로 저장한다.
- 확인은 Adobe Arcrobat Reader로 사용한다.
- Report(①)에는 핵심 개정 내용을 표기한다.
- Verion (②)에는 문서의 개정 번호를 표기한다.
- Last Modified(③)에는 문서의 개정 일자를 표기한다.


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- ④ 에는 프로젝트 명을 표기한다.


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

목차

개 정 이 력	1
문 서 규 칙	1
프로젝트 규칙	5
각 서버의 특징.....	6
Linux Server.....	6
Windows Server.....	7
Linux, Windows 핵심비교.....	7
설치와 부팅	8
Linux Server 사양	8
Windows Server 사양	9
Linux Server의 부팅과정	9
Windows Server의 부팅과정	10
네트워크 설정	14
Linux Server network 설정	14
Linux Server 네트워크 관련 명령어	15
Windows Server network 설정	17
Windows Server 네트워크 관련 명령어	18
사용자, 그룹 설정	19
Linux Server의 사용자, 그룹설정.....	19
Linux Server 사용자, 그룹 명령어.....	21
Windows Server의 사용자, 그룹 설정.....	22
사용자, 그룹 권한 설정	23
Linux Server 권한 설정	24
Linux Server 권한 명령어	24

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


Windows Server 권한 설정	25
마운트, 파티션 설정	26
LVM(Logical Volume Manage)	26
RAID (RedundantArray of Independent Disks).....	28
Linux Server 마운트, 파티션, RAID 설정	29
Linux Server 마운트, 파티션, RAID 명령어.....	30
Windows Server 마운트, 파티션, RAID 설정	33
BACKUP, RESTORE	33
Linux Server Backup, Restore.....	34
Linux Server Backup, Restore 명령어 정리.....	34
Windows Server Backup, Restore.....	36
FTP SERVER	38
Linux FTP Server.....	39
Windows FTP Server.....	41
TELNET SERVER	42
Linux TELNET Server.....	42
Windows TELNET Server.....	44
SSH SERVER.....	44
Linux SSH Server	45
Windows SSH Server	47
NTP SERVER	48
Linux NTP Server	48
Windows NTP Server	49
NFS SERVER.....	49
Linux NFS Server	51
Windows NFS Server	53

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

DHCP SERVER.....	53
Linux DHCP Server.....	54
Windows DHCP Server.....	56
 DNS SERVER.....	 56
Linux DNS Server.....	57
Windows DNS Server	61
 WEB SERVER.....	 61
Linux Web Server	62
Windows Web Server	65
 MAIL SERVER.....	 65
Linux Mail Server.....	66
Windows Mail Server.....	69

프로젝트 규칙


- 본 프로젝트는 Linux Server(CentOS 7), Windows Server(2016)을 사용한다.
- 본 프로젝트는 VMware workstation을 사용한다.
- 본 프로젝트 각 지속적으로 업데이트가 이루어진다.
- 본 프로젝트의 L는 Linux Server, W는 Windows Server를 나타낸다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

각 서버의 특징

Linux Server

- PC에서 UNIX 시스템과 비슷하게 운영되도록 만든 운영체제의 한 종류
- Unix 기반 시스템과 호환되는 프로그래밍 인터페이스와 사용자 인터페이스를 제공
- 다중사용자 시스템으로 프로세스 간 통신은 메시지 큐, 공유 메모리 및 세마포와 같은 메커니즘에서 모두 지원
- Linux Server의 가장 큰 특징은 오픈소스 정책이 가장 큰 메리트이자 핵심
- Linux는 오픈소스로 이루어진 만큼 매우 많은 버전들이 존재하며 그 중 가장 대표적인 것이 Redhat(CentOS), Debian(Ubuntu), 슬랙웨어(델리 리눅스) 등이 있음
- 강력한 네트워크 기능을 가지고 있으며 Windows Server에 비해 높은 신뢰성과 보안성을 가지고 있음
- Windows Server에 비해 자원대비 최적화된 성능을 보이며, 서버에 걸리는 과부하를 감당할

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

수 있어 서버 운영체제로 적합


- 오픈소스인만큼 많은 기관들이 리눅스를 전문적으로 지원하며, 오류가 발생시 복구가 빠름
- 무료로 배포되는 만큼 Windows Server에 비해 구축 비용이 낮음
- Windows Server에 비해 비교적 적은 컴퓨터 리소스로 운용이 가능
- 모놀리식 커널을 사용
- 대소문자를 구분

Windows Server

- Windows Server는 소스 코드에 액세스 할 수 없는 라이선스가 부여된 운영체제로 주로 프로그래밍 지식이 없는 개인과 비즈니스 및 기타 상업 사용자를 위해 설계
- 확장이 가능하고 이식이 가능하며 여러 운영 환경 및 대칭 적 다중 처리 및 클라이언트-서버 컴퓨팅을 지원
- Windows Server에 대표적인 장점은 GUI 환경을 제공(최근들어 Linux 또한 GUI를 제공)
- 명령의 입력이 아닌 사용자의 행동으로 여러 기능을 수행할 수 있는 등의 편리성을 제공
- Windows용 프로그램들 사이에는 일관성이 있으며, 익숙한 운영체제인만큼 사용자가 적응하기 쉬운 특징을 가지고 있음
- 도움말 기능이 풍부하여 처음 접하는 사람들도 쉽게 사용할 수 있음.
- 마이크로 커널을 사용
- 대소문자를 구분하지 않음

Linux, Windows 핵심비교

비교 근거	Linux	Windows
Cost	Free of cost	Expensive
Open Source	Yes	No
Customizable	Yes	No

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


Security	More secure	Vulnerable to viruses and malware attacks
Booting	Either primary or logical partition	Only primary partition
Separation of the directories using	Forward slash	Back slash
File names	Case sensitive	Case insensitive
File system	EXT2, EXT3, EXT4, Reisers FS, XFS and JFS	FAT, FAT32, NTFS and Re FS
Type of kernel used	Monolithic Kernel	Microkernel
Efficiency	Effective running efficiency	Lower than Linux

설치와 부팅

- 위에서 언급되었듯이 Linux는 오픈소스이며, Windows는 유료로 180일의 체험판만을 제공한다.
- 설치하는 하단의 표와 같이 두 운영체제 모두 동일하게 설치가 가능하며, 여기서는 VMware workstation 16을 사용하여 동일한 용량에서 설치하겠다.

Linux Server 사양

VM 이름	L-1	L-2	L-3
용도	서버 및 클라이언트	서버 및 클라이언트	서버 및 클라이언트
OS 종류	CentOS 7	CentOS 7	CentOS 7
CPU 코어, 쓰레드	1/2	1/2	1/2
하드용량	50GB (NVME)	50GB (NVME)	50GB (NVME)
메모리	2048MB	2048MB	2048MB
네트워크 타입	NAT(차후수정)	NAT(차후수정)	NAT(차후수정)
CD/DVD	O	O	O

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


Audio	X	X	X
USB	X	X	X

Windows Server 사양

VM 이름	W-1	W-2	W-3
용도	서버 및 클라이언트	서버 및 클라이언트	서버 및 클라이언트
OS 종류	Windows Sever 2016	Windows Sever 2016	Windows Sever 2016
CPU 코어, 쓰레드	1/2	1/2	1/2
하드용량	50GB (NVME)	50GB (NVME)	50GB (NVME)
메모리	2048MB	2048MB	2048MB
네트워크 타입	NAT(차후수정)	NAT(차후수정)	NAT(차후수정)
CD/DVD	O	O	O
Audio	X	X	X
USB	X	X	X

Linux Server의 부팅과정

- 1. 시스템 정원 공급
- 메인보드의 ROM-BIOS에 있는 BIOS프로그램 자동실행, BIOS 프로그램은 전원공급과 함께 메모리의 특정위치에 자동로드 되며, 이와 함께 CPU는 전원공급과 함께 특정위치에 BIOS 프로그램을 자동실행
- 2. BIOS 프로그램들의 실행내용
 - 자체진단기능(POST) : CMOS 검사, CPU, MEMORY, 그래픽카드, 키보드, 마우스 등 각종 장치들의 이상 유무를 검사하고 이들 장치의 초기화를 진행
 - 부팅매체검색 및 부트로더 실행 : POST과정이 이상없이 완료되면 부팅매체(Master Boot Record(하드, CD 등))에서 부트로더 (GRUB, LILO)를 로드
- 3. 부트로더(GRUB) 실행
 - GRUB은 실행과 함께 /boot/grub/grub.conf 파일을 읽어서 어떤 부팅메뉴로 부팅할 것인지를


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

결정해야 되며, 이 화면이 GRUB이 화면에 나타나는 첫째 화면을 의미

- GRUB는 커널(kernel) 이미지를 불러들여 시스템 제어권을 커널에게 이전
- **4. 커널의 로딩**
- 커널은 swapper 프로세스 (PID 0)을 호출
- Swapper은 커널이 사용할 각 장치드라이브들을 초기화하고 init(PID 1)을 실행
- init 프로세스가 실행되면서 /etc/inittab 파일을 읽어들여서 그 내용들을 차례대로 실행
- **5. init 프로세스의 실행**
- /sbin/init의 있는 내용들을 모두 실행시킴(여기에 커널이 포함)

Windows Server의 부팅과정

- 일반적으로 부팅은 Boot Manager이 메모리 상에 로드되고 실행되면서 시작된다. 즉, 메인보드 펌웨어 타입에 따라 Boot Manager의 실행까지 차이가 존재한다.
- **1-1. BIOS(Basic Input/Output System)**
- **1. POST(Power on Self Test)**
- 펌웨어가 BIOS 일 경우, 가장먼저 POST 동작을 수행하며, POST는 장착된 하드웨어를 점검하며 전반적인 Device들의 상태를 확인한다.
- **2. MBR boot code**
- BIOS는 부팅 섹터를 찾기 위해 부팅 가능한 디바이스 목록을 검색하며, 부팅 가능한 장치가 하드 디스크라면 이 장치의 부트 섹터를 MBR이라 부른다.(만약 부팅 가능한 장치가 CD, Floppy 디스켓 등과 같이 하드 디스크가 아니라면 BIOS는 장치의 VBR을 메모리에 로드하여 실행시킨다.)

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- 3. VBR boot code, Boot Manager

- VBR 에도 VBR boot code가 존재하며, VBR boot code는 자신의 파티션에 16bit의 boot manager program을 찾아 실행시킨다.

- BIOS의 부팅 흐름은 하단과 같다.

BIOS -> POST -> Load MBR -> Load VBR -> Run bootmgr

- 1-2. EFI(Extensible Firmware Interface)

- 1. POST(Power on Self Test)

- 펌웨어가 EFI 일 경우 동일하게 POST 동작을 수행한다.

- 2. Boot code, Boot Manager(bootmgfw.efi)


- EFI 방식에서의 Boot code는 MBR이나 VBR에 존재하지 않고, 펌웨어 안에 존재한다. 펌웨어 안에 존재하는 Boot code는 %SystemDrive%\EFI\Microsoft\Boot\Bootmgfw.efi에 존재하는 EFI 실행 프로그램을 구동시킨다.

- EFI 부팅의 흐름은 하단과 같다.

EFI -> POST -> RUN BootCode(in EFI) -> Run Bootmgfw.efi

- 2. Boot Mnager (부팅 관리자)

- 1-1, 1-2와 방식 둘다 최종적으로는 Boot Manager(bootmgr)을 메모리에 로드하여 실행시킨다. Boot Manager은 BCD(Boot Cnfiguration Data) 설정 데이터를 읽어 시스템을 시작하며, BCD는 레지스트리 하이드 파일로 존재하며 최종적으로는 Windows boot Loader 인 winload.exe를 로드하고 실행시킨다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- BCD(Boot Configuration Data)

- BCD 하이브 데이터는 HKLM\BCD00000000 레지스트리 경로에 존재하며, 일반적으로 BCD 파일을 조작하기 위해서는 bcdedit.exe를 사용한다. 또한 BCD는 다음과 같은 최소한 2개 이상의 요소를 포함하고 있다. (하나의 Windows Boot Manager Object, 하나 이상의 Windows Boot Loader Object)

- Boot Manager Object


- BCD가 포함하고 있는 Boot Manager Object는 문자 셋 기반의 Boot manager 화면의 모든 설정 값을 가지며, OS의 개수, 부팅 도구 메뉴, 기본 타입 아웃 값 등이 이에 해당한다.
- Boot Manager Object는 {9DEA862C-5CDD-4E70-ACC1-F32B344D4795} 레지스트리에 정의 되어 있으며, bcdedit.exe 실행 시 {bootmgr} 이라는 이름으로 볼 수 있다.

- Windows Boot Loader (윈도우 부트 로더)

- 구동시킬 운영체제가 선택 된 후 Boot Manager은 선택된 운영체제의 Boot Loader Object가 가리키고 있는 위치의 Windows boot Loader을 로드하고 실행 시킨다.(winload.exe를 의미하며 일반적으로는 %SystemRoot%\System32 경로에 존재한다.)
- 여기서 하단의 작업들을 수행한다.
- system 레지스트링 로딩
- self 무결성 검사
- ntoskrnl.exe, hal.dll 로드

4. Boot Class Service 로드리스트

- bootvid.dll
- ci.dll
- clfs.sys

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


- hal.dll
- kdcom.dll
- kd1394.sys.
- kdbus.dll
- ntoskrne.exe
- pshed.dll
- winload.exe
- ksecdd.sys
- spldr.sys
- tpm.sys
-

5. Protected Mode 전환 준비 및 ntoskrnl.exe 실행

- 최종적으로 winload.exe는 protected mode의 페이징을 활성화하고, boot log를 저장 한 뒤, ntoskrnl.exe로 실행을 넘겨 준다.
- Winload.exe 에서 수행하는 동작 흐름도
- Winload.exe -> load system hive -> integrity self-check -> load, verify(ntoskrni.exe, hal.dll), load, verify(ntoskrnl.exe`s imports) -> sacn registry -> load, berify(boot class drivers)

5. ntoskrnl.exe 실행(executive 동작)

- System class service 로드
- Smss.exe 로드

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

6. smss.exe 실행


7. wininit.exe 실행

네트워크 설정

Linux Server network 설정

- 기본적으로 Linux(CentOs 7)의 네트워크 인터페이스 경로는 /etc/sysconfig/network-scripts/에 위치해 있다.
- 여기서 ifcfg-xxxxx(ex. ens33)는 해당 인터페이스를 의미하며 이 파일 속에는 하단과 같이 나열되어 있다.

TYPE=Ethernet	# 네트워크 인터페이스의 타입
PROXY_METHOD=none	# 프록시 서버 사용방법
BROWSER_ONLY=no	
BOOTPROTO=dhcp	# IP 설정방법(DHCP or Static)
DEFROUTE=yes	# 인터페이스의 기본 라우팅 경로 사용 유무
IPV4_FAILURE_FATAL=no	# IPv4 및 IPv6 설정
IPV6INIT=yes	
IPV6_AUTOCONF=yes	
IPV6_DEFROUTE=yes	
IPV6_FAILURE_FATAL=no	
IPV6_ADDR_GEN_MODE=stable-privacy	

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


NAME=ens32 # 네트워크 인터페이스의 이름
 UUID=58f3d7fd-6379-4b2a-b9b6-6601a7e7b313 # 네트워크 인터페이스의 고유값
 DEVICE=ens32 # 네트워크 인터페이스의 이름
 ONBOOT=no # 시스템 부팅 시 인터페이스 활성화 유무
 IPADDR=192.168.10.10 # (static) 고유 IP 값
 NETMASK=255.255.255.0 # (static) netmask 값
 GATEWAY=192.168.10.2 # (static) gateway 값
 DNS1=8.8.8.8 # (static) DNS 값

설정이 완료되면 `systemctl restart network`를 통해 설정변경이 가능하다.


`dhclient`를 통해 `dhcp` 서버에서 IP를 해당받는 것 또한 가능하다.

Linux Server 네트워크 관련 명령어

구분	옵션	설명
IP (통합 네트워크 설정 프로그램)	link	네트워크 연결 확인
	addr	네트워크 IP 확인
	route	라우팅 테이블 확인
Ifconfig [인터페이스] (네트워크 인터페이스 설정 및 확인)		Mac 주소, 인터페이스에 대한 설명, MTU 등을 확인 및 설정이 가능
	lo 127.0.0.1	루프백 인터페이스 추가작업
netstat (인터페이스 통계 및 기타 정보 확인)	-a	--all로 Listen 되는 소켓정보와 listen 되지 않는 소 켓정보를 모두 출력
	-n	--numeric과 같으며 10진수의 수치정보로 결과를 출력
	-r	--router과 같으며 설정된 라우팅 정보를 출력
	-p	--programs와 같으며 실행되는 PID 정보를 출력
	-i	--interface=eth0과 같으며 모든 네트워크 인터페이 스 정보를 출력
	-c	--continuous와 같으며 netstat 결과를 연속적으로 출력
	-l	--listening과 같으며 listen 되고 있는 소켓 정보를 출력
	-s	--statistics와 같으며 각 프로토콜에 대한 통계정보

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

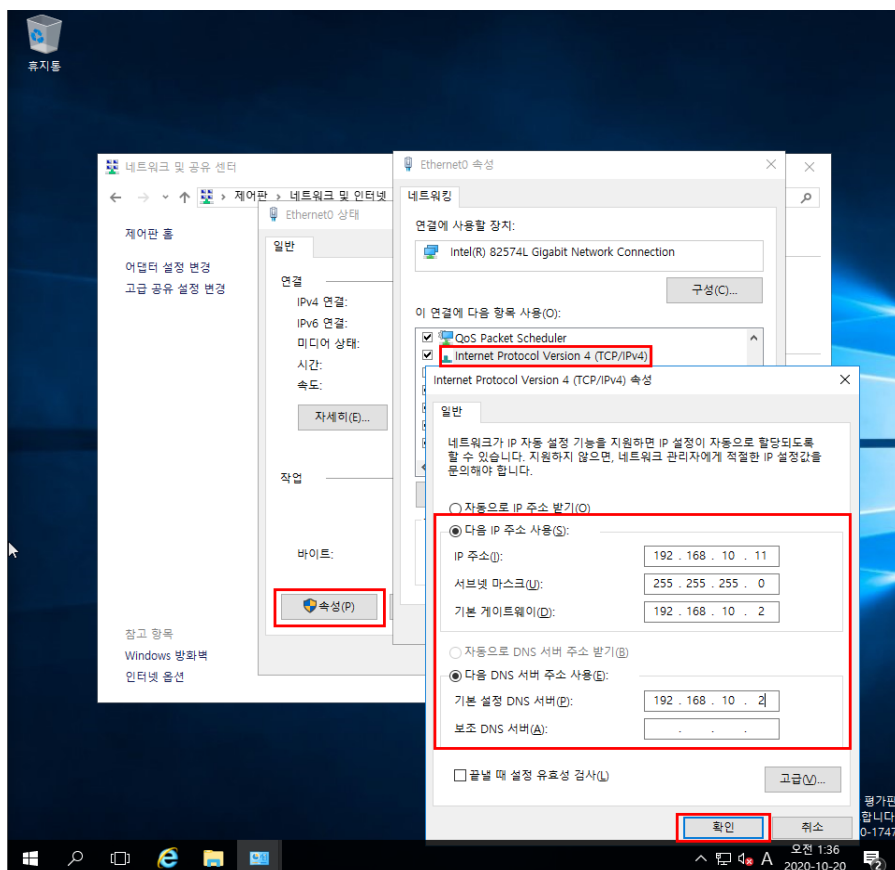
		를 출력
route (라우팅 테이블 확인 및 설정)	-n	라우팅 테이블 확인
	add	-net 네트워크 주소 netmask 넷마스크 [인터페이스] 새로운 라우팅 테이블 추가
	del	-net 네트워크 주소 netmask 넷마스크 [인터페이스] 기존 라우팅 테이블 제거
	add	Default gw 기본게이트 웨이 주소 [인터페이스명] 기본 게이트웨이 추가
	del	Default gw 기본게이트 웨이 주소 [인터페이스명] 기본 게이트웨이 제거
	add	-net 127.0.0.0 루프백 인터페이스 추가
Ping [옵션] [목적지주소] (원격 호스트 응답 테스트)	-t	Ctrl+c 중단시키기 전까지 계속 ping 패킷을 보낸다.
	-a	도메인네임의 IP 주소를 보여준다.
	-n [count]	Ping 패킷을 몇 번 보낼지 패킷 수를 지정한다.
	-l [size]	Ping 패킷 크기를 지정한다.
	-f	요구 패킷이 라우터 등을 통과할 경우, 나누어지지 않게 한다.
	-i TTL	사용할 TTL 값을 설정한다.
	-v TOS	패킷의 TOS를 지정하는 값으로 설정한다.
	-r count	Count에 지정한 값만큼 라우터의 경로를 보여준다.
	-s count	Count에 지정한 값만큼 라우터의 주소와 시간을 기록한다.
	-j host-list	리스트의 컴퓨터를 통해서 패킷의 경로를 정할 수 있다.
arp (ARP 테이블 관리)	-k host-list	J와 동일한 역할을 수행하지만 연속적인 컴퓨터의 경우 중간 게이트를 구분할 수 없다.
	-w timeout	대상 호스트의 응답을 기다리는 시간을 지정한다.
	-a	APR 캐쉬 테이블에 저장되어 있는 모든 호스트정보를 출력한다.
	-s	APR 캐쉬 테이블에 저장된 특정 IP에 대한 MAC 주소를 변경한다.
	-d	APR 캐쉬 테이블에 저장된 특정 MAC 주소를 삭제한다.
	-i	Eth0과 같은 네트워크 인터페이스를 지정한다.
	-n	Resolving을 하지 않고 IP주소를 출력한다.


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

	-v	자세한 모드(verbose)로 결과를 출력한다.
nslookup		도메인 네임 서버 질의
Tracepath [목적지 IP]		목적지 IP까지 도착할 때의 경로들을 출력

Windows Server network 설정


- 윈도우는 기본적으로 네트워크 연결이 되어있으면 자동적으로 DHCP서버에서 유동적으로 IP를 할당받는다.
- 정적 IP설정은 위해서는 네트워크 어댑터 > 인터페이스의 속성 > 인터넷 프로토콜 버전 4(TCP/IPv4)에서 자동으로 IP 주소 받기가 아닌 다음 IP 주소 사용을 선택한다.
- 그림과 같이 설정이 완료되면 자동으로 적용된다.



	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Windows Server 네트워크 관련 명령어

구분	명령어	설명
Tracert [옵션] 목적지주소		지정된 호스트에 도달할 때까지 통과하는 경로의 정보와 지연시간
	-d	d 옵션을 사용하면 reverse look-up을 시도하지 않음
Ping [옵션] [목적지주소] (원격 호스트 응답 테스트)	-t	Ctrl+C를 누를 때까지 특정 호스트로 ping을 수행
	-a	IP주소를 호스트 이름으로 풀이
	-n count	전달할 에코 옷어 수(밀리 초)
	-l size	버퍼 크기를 지정
	-f	패킷에 don't Fragment 플래그를 설정
	-i TTL	Time To Live를 출력
	-v TOS	Type of Service를 출력
	-r count	라우트를 레코드해서 홉을 카운트
	-s count	홉 카운트에 대한 타임 스탬프
	-j host-list	호스트 리스트를 가지고 있는 loose 라우트
	-w timeout	최대 응답 대기 시간
arp (ARP 테이블 관리)		IP 주소 대 하드웨어 주소 맵을 출력
	-s [IP]	ARP 맵에 해당 IP 주소의 하드웨어를 추가
netstat (현재 port의 상태를 출력)	-ant [time]	Time 시간마다 갱신해서 출력
nbtstat		IP 주소에 해석된NetBIOS 컴퓨터 이름 목록 출력
ipconfig (네트워크 인터페이스 설정 및 확인)		설치된 네트워크 카드에 대한 현재 구성 출력
	/ALL	자세한 정보를 출력
	/Batch[File]	Ipconfig 요청에서 파일로 정보를 저장
	/Renewl (N)	모든(N) 어댑터에 대한 DHCP 리스트를 갱신
	/Release (N)	모든(N) 어댑터에 대한 DHCP 리스트를 해제
nslookup (호스트에 상응하는 주소를 출력)		기본 서버를 사용하는 대화형 모드
	[도메인]	도메인을 사용하는 대화형 모드
	[host]	기본 서버를 사용하는 host만을 조회
	[domain] [host]	도메인을 사용하는 host만을 조회
Route [-f] [-p] [-4 -6] command [destination]	-f	모든 게이트웨이 항목의 라우팅 테이블을 삭제
	-p	ADD 명령과 함께 옵션을 사용하면 시스템을 부팅

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

[MASK netmask] [gateway] [METRIC metric] [IF interface] (라우팅 테이블 확인 및 설정)		해도 경로가 지정이 유지
	-4	IPv4를 사용
	-6	IPv6를 사용
	destination	호스트를 지정
	MASK	다음 매개변수의 값이 netmask 값을 지정
	netmask	기본값 255.255.255.255이며 설정 시마스크 값 지정
	gateway	게이트웨이 값을 지정
	interface	지정한 경로에 대한 인터페이스 번호 지정
	METRIC	대상의 비용과 같은 매트릭을 지정

사용자, 그룹 설정


Linux Server의 사용자, 그룹설정

- 리눅스는 기본적으로 다중 사용자 시스템이며, 리눅스를 설치하면 기본적으로 root라는 관리자 권한을 가진 슈퍼 유저가 있고, 이 관리자는 모든 시스템과 파일에 접근할 수 있는 권한을 가지고 있다.
- 하지만 모든 사용자가 이와 같은 권한을 소유할 경우 문제가 될 수 있으며, 이에 따라 사용자, 그룹에 따른 권한을 부여함으로써 여러 문제사항을 사전에 방지한다.
- 사용자에 대한 정보는 /etc/passwd에 나와 있으며, 이에 대한 내용은 아래와 같다.

root : x : 0 : 0 : root : /root : /bin/bash

Username : Password : User ID : Group ID : User ID Info : Home directory : Command/Shell

구분	설명
Username	사용자명
Password	사용자의 암호가 기록되는 부분 (x는 암호화를 의미) 사용자의 패스워드는 /etc/shadow에 저장
User ID	사용자 아이디 값으로 0 = root를 의미

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

	(1-99는 이미 정의된 계정, 100-999는 시스템 계정, 1000-는 관리자가 생성한 계정)
Group ID	그룹 아이디로 사용자가 속한 그룹의 번호를 의미
User ID Info	사용자에 대한 추가적인 정보가 기록
Home directory	사용자의 홈 디렉토리
Command/Shell	로그인시 사용되는 셸을 의미

- 사용자 계정의 아이디, 암호의 대한 값은 /etc/shadow에 저장되어 있으며, 이에 대한 내용은 아래와 같다.

root : \$6\$NUnhR3qP8...../ : : 0 : 99999 : 7 : : :

Login name : Encrypted : Last Changed : Minimum : Maximum : Warn : Inactive : Expire : Reserved


구분	설명
Login Name	사용자 계정의 이름
Encrypted	암호화된 패스워드 값 (&[값]&은 해시 값을 의미)
Last Changed	1970년 1월 1일 부터의 패스워드 수정된 날짜의 일수를 계산
Minimum	패스워드 변경 전 최소 사용기간(일)
Maximum	패스워드 변경 전 최대 사용기간(일)
Warn	패스워드 사용 만기일 전에 경고 메시지를 제공하는 일수
Inactive	로그인 접속차단 일 수
Expire	로그인 사용을 금지하는 일 수 (월/일/연도)
Reserved	사용하지 않음

- 그룹의 대한 설정은 /etc/group에 저장되어 있으며, 이에 대한 내용은 아래와 같다.

root : x : 0 :

Group name : Password : GID : members

구분	설명
Group name	그룹의 이름
Password	패스워드 값
GID	GID 값
members	그룹에 속한 유저

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- 그룹또한 /etc/gshadow가 존재하며, 이에 대한 내용은 아래와 같다.


chrony : ! : :

Group name : Password : Admin : members

구분	설명
Group name	그룹의 이름
Password	패스워드 값
Admin	관리자
members	그룹에 속한 유저

Linux Server 사용자, 그룹 명령어


구분	명령어	설명
useradd [계정명] [-c "name"] [-s /bin/bash] [-m -M] [-d /home/name] [-N] [-u UID] [-g UID]	-c "name"	사용자의 이름
	-s /bin/bash	사용자가 사용할 셸
	-m -M	홈 디렉터리 생성 : -m 생성 안함 : -M
	-d /home/name	홈 디렉터리의 위치지정
	-N	개인 그룹을 생성하지 않음
	-u UID	UID 수동으로 사용자 아이디 지정
	-g UID	수동으로 그룹 ID 지정
useradd -D [옵션] usermod [계정명]		useradd : 기본 값 확인 usermod : 사용자 정보 수정
	-l	사용자명 변경
	-c	이름 변경
	-s	로그인 셸 변경
	-d	홈디렉터리 위치 변경
	-u	UID 수동으로 사용자 아이디 지정
	-e [20xx-xx-xx]	설정 기간에 계정이 만료되도록 설정
Passwd [옵션] [계정명]		특정 계정의 패스워드 변경
	-l	계정잠금
	-u	계정 잠금 해제
	-n [count]	Count 만큼의 암호 최소사용기간 설정
	-l [count]	Count 만큼의 계정 만기 유예기간을 설정
	-e	다음 번 로그인 시 암호 강제 변경

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

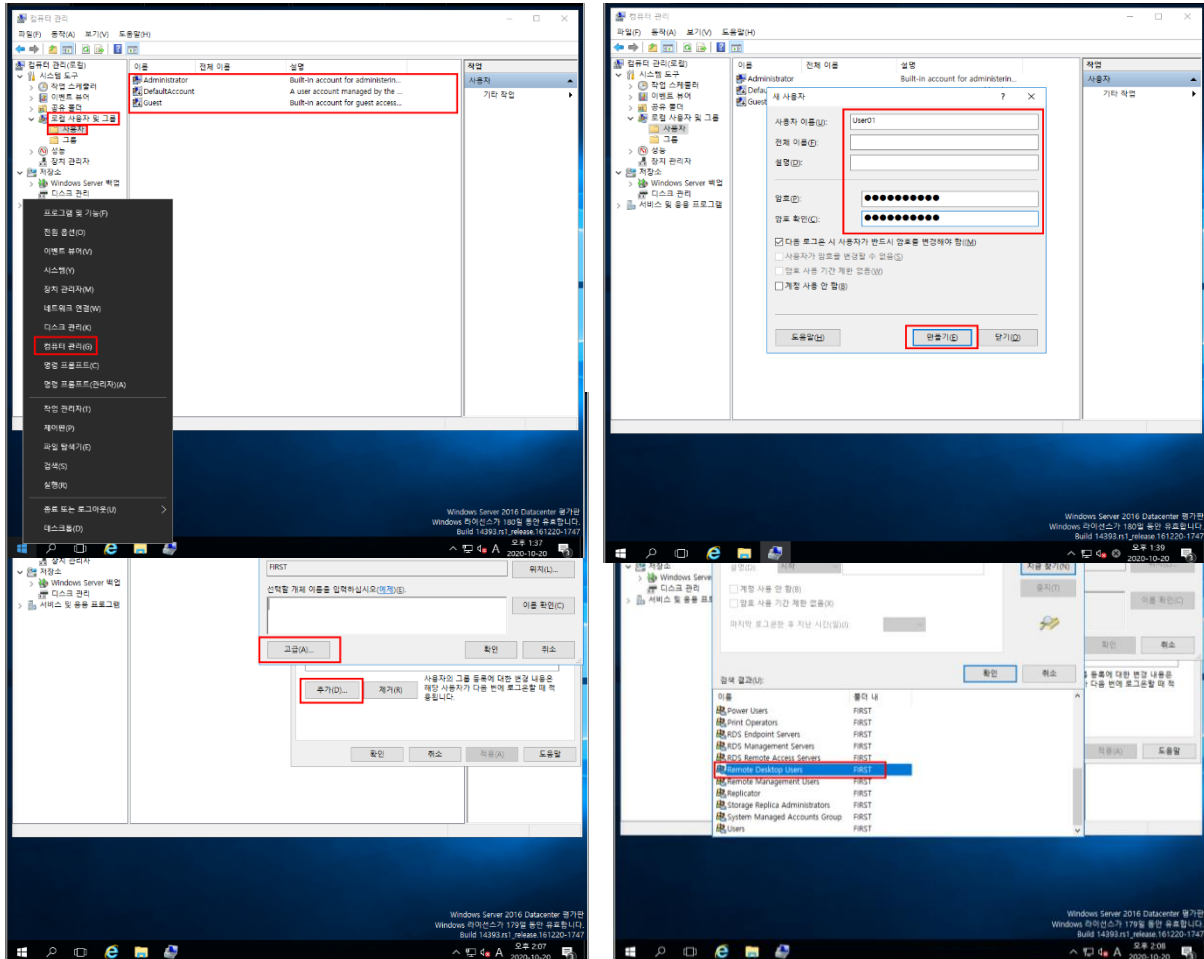
Userdel [옵션] [계정명]	-r	특정 계정을 삭제
groups [계정명]		계정명이 속한 그룹의 리스트를 출력
groupadd [옵션] 그룹명		그룹명의 그룹을 생성
	-g [GID]	[GID] 번호를 가진 그룹을 생성
	-G [그룹1 그룹2]	계정1, 계정2의 그룹을 포함하여 그룹을 생성
groupdel [그룹명]		그룹명의 그룹을 삭제
gpasswd [옵션] [그룹명]		그룹명의 그룹의 암호를 변경
	-r	그룹명 그룹의 암호를 삭제
	-A	현재 사용자를 그룹 관리자로 설정
	-a [계정명]	특정 계정을 그룹에서 포함
	-d [계정명]	특정 계정을 그룹에서 제외
newgrp [그룹명]		작업한 내용들을 그룹명으로 기록
groupmod [옵션] [계정명]	-g [GID]	특정 그룹의 GID를 수정
	-o [GID]	특정 그룹의 GID를 중복허용시에 사용
	-n [new 계정명]	특정 그룹의 이름을 변경시 사용

Windows Server의 사용자, 그룹 설정


- Windows Server 또한 기본적으로 Administrator 계정이 생성되며, Linux의 admin의 역할을 수행한다.
- Windows Server에서 User나 Group를 생성하기 위해서는 제어판 -> 관리 도구 -> 컴퓨터 관리를 실행 후,
- 컴퓨터 관리(로컬) -> 시스템 도구 -> 로컬 사용자 및 그룹에서 사용자 및 그룹을 생성할 수 있다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- Windows Server는 기본적으로 사용자를 생성시 그룹을 지정하는 것이 아닌, 사용자를 생성하고 나서 그룹에 포함시킬 수 있다.



사용자, 그룹 권한 설정

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Linux Server 권한 설정


- Linux에서는 기본적으로 권한(퍼미션, 허가권)이 존재하며 이는 ls -l 명령어로 확인이 가능하다.
- Linux에서의 문서형식은 하단과 같은 형태를 이루고 있으며 다음과 같은 의미를 나타낸다.

-rwxrwxrwx 1 root root 52 jul 3 20:06 test.txt

구분	의미	설명
-rwxrwxrwx	파일 종류 및 권한 (퍼미션) r : Read 권한 w : write 권한 x : execute 권한	(-) 파일 (d) 디렉토리를 의미 rwx : 소유자(owner) 권한 rwx : 그룹(group) 권한 rwx : 타사용자(other) 권한 부여된 각각 권한에 따라 사용가능 여부 가 결정
1	링크 수	링크된 파일 수
root	소유자(owner)	소유자 명
root	그룹(group)	그룹 수
52	파일 사이즈	파일 사이즈
Jul 3 20:06	최종 수정날짜	최종 수정날짜
test.txt	파일의 이름	파일의 이름

Linux Server 권한 명령어


구분	옵션	설명
chmod [옵션] [파일이나 디렉토리]	g + w	u(owner), g(group), o(other), a(all)에게 + 혹은 -, = 를 사용해서 rwx 권한을 부여하거나 삭제할 수 있다.
	uo - r	
	a = rws	
	000	권한은 최대 7로 나타내며 4(read), 2(write), 1(execute) 권한을 의미한다. 000은 그 누구에게도 권한을 주지 않는 것을 의미한다..
	777	모두에게 모든 권한을 주는 것을 의미한다.
chmod [옵션] [파일이나 디렉토리]	742	owner에게 모든 권한(7)을 group에게 읽기 권한(4)을 다른 유저에게는 쓰기 권한(2)만을 부여한다.
	1xxx	SetUID로 소유자만 접근할 수 있는 파일을 일반

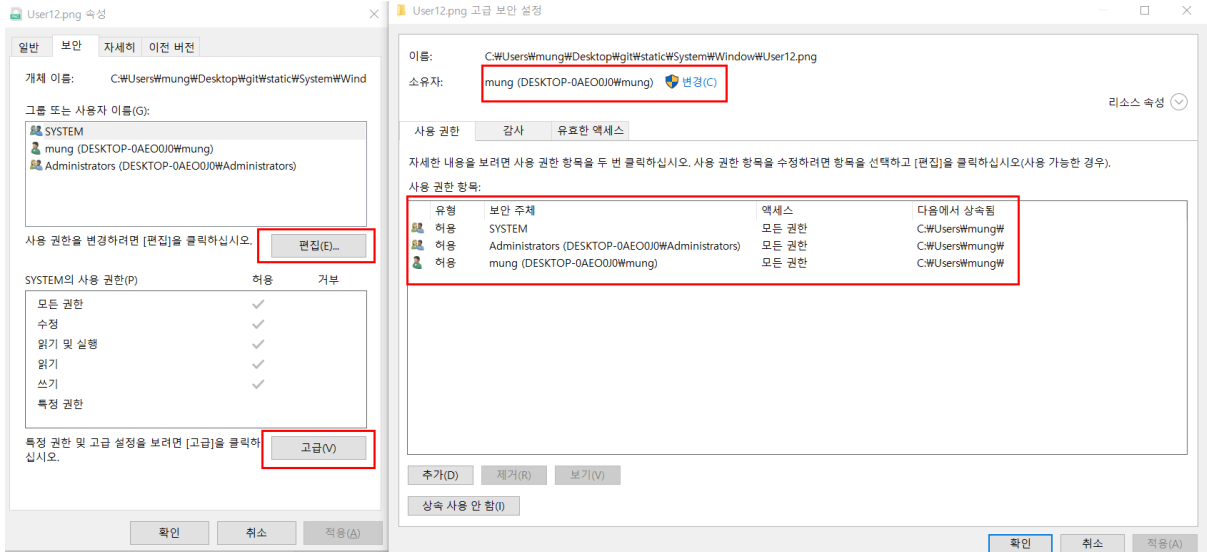
	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

디렉토리] (특수권한)		유저도 접근할 수 있게 한다. 권한표시 : ex) rws rwx rwx = 4777
	2xxx	SetGID로 소유 그룹만 접근 가능한 파일에 일반 유저로 접근이 필요 할 때 사용한다. 권한표시 : ex) rwx rws rwx = 2777
	4xxx	Sticky bit란 특정 디렉토리를 누구나 자유롭게 사용할 수 있도록 할 때 사용한다. (단 삭제와 생성은 owner만 가능) 권한표시 : ex) rwx rwx rwt = 1777
	-R	디렉토리의 경우 내부에 모든 파일, 디렉토리까 지 권한이 변경된다.
chgrp [그룹명] [파일이나 디렉토리]		특정 파일이나 디렉토리의 소유그룹을 변경한다.
	-R	디렉토리의 경우 내부에 모든 파일, 디렉토리까 지 그룹이 변경된다.
chown [계정명] [파일이나 디렉토리]		특정 파일이나 디렉토리의 소유자를 변경한다.
	-R	디렉토리의 경우 내부에 모든 파일, 디렉토리까 지 소유자가 변경된다.
	[계정명.계정명]	계정명.계정명을 사용할 경우 소유자와 소유 그룹을 동시에 변경이 가능하다.
sudo [명령어]		관리자의 권한으로 명령어를 실행시킨다.

Windows Server 권한 설정

- Windows Server 또한 각 사용자, 그룹 및 특정 폴더의 권한의 대한 설정이 가능하다.
- Windows Sever에서의 권한 설정은 그룹의 정책에서 가능하거나, 혹은 특정 파일 디렉터리 우
클릭 -> 속성 -> 보안 -> 그룹 또는 사용자 이름, SYSTEM의 사용 권한을 통해 변경이 가능
하다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	




마운트, 파티션 설정

LVM(Logical Volume Manage)

- LVM이란 물리적인 디스크를 논리적 볼륨그룹으로 구성하여 볼륨그룹 내에 사용자가 원하는 크기 만큼의 논리볼륨을 할당하여 사용하는 방법으로, 여러 개의 디스크를 하나의 논리적인 볼륨그룹(VG)으로 구성할 수 있으며, 각 논리볼륨(LV)는 독립적이므로, 하나의 디스크에도 여러 가지의 파일 시스템을 설치할 수 있다.
- LVM을 사용하면 하나의 디스크에도 여러 가지의 파일 시스템을 설치할 수 있으며, 또한 디스크의 용량증설이 요구될 경우에도 볼륨그룹에 물리볼륨(PV)만 추가하여 용량증설이 가능하고, 데이터량이 감소하여 더 이상 많은 양의 저장 공간이 필요 없는 경우, 볼륨그룹에 속한 물리볼륨 개수를 축소하여 용량 감소가 가능하다.

PV(Physical Volume)

- PV는 하나의 물리적인 Disk를 LVM을 사용할 수 있도록 LVM 데이터 구조를 생성한 것을 의미하며, 디스크에 PV가 생성되면 LVM은 디스크를 하나의 물리볼륨으로 간주하게 되고, 볼륨그룹에 포함시킬 수 있게 된다. 따라서 하나의 디스크를 몇 개의 섹션은 논리볼륨으로, 몇 개의 섹션은 전체디스크로 관리하는 등의 방법은 적용 할 수


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

없다.

- PE(Physical Extent)는 PV가 갖는 일정한 블록의 크기로 디스크에 PV를 생성하게 되면, LVM은 주소를 지정할 수 있는 PE라는 단위로 각 물리 디스크를 나누며, extent는 주소 0번부터 시작하여 1씩 증분하며 순차적으로 디스크에 할당된다.

- **VG(Volume Group)**
 - VG는 하나 또는 그 이상의 PV를 포함하며, LV를 생성할 수 있는 Volume Group의 집합을 의미한다.
 - vg00은 부팅정보와 OS가 있는 root Volume Group으로 변경이 불가능하다.
 - 관리자가 관리상의 편의를 위해 DB의 경우 vgdb##, 일반적인 파일 시스템의 경우에는 vgfil##등으로 VG이름을 지정하여 사용한다.


- **LV(Logical Volume)**
 - LV는 하나 또는 그 이상의 PV로 구성되어 있는 VG 공간을 전체 또는 분할하여 일반 파일 시스템, swap or dump area, raw disk로 사용할 수 있도록 논리적으로 할당한 공간을 의미한다.
 - 운영 중 공간이 부족할 경우 볼륨그룹에 속해 있는 또 다른 물리볼륨을 사용하여 확장이 가능하고, 필요에 따라서 크기를 변경하거나 다른 디스크로 데이터릴 옮길 수 있다.
 - Volume Group vg00에 속한 Logical Volume lvol1, lvol2, lvol3은 각각 /stand, primary swap, / 디렉터리로 변경이 불가능하다.
 - LE(Logical Extent)는 LV가 갖는 블록의 형태로 디스크에 PV를 생성하고, 생성된 PV 이용하여 VG를 구성한 후에 할당이 가능하며, 이에 대한 할당 단위를 LE라 한다.
 - LE는 PV 생성 시 나누어진 PE 영역과 Mapping되며 PE의 크기가 4MB이면 LE의 크기도 4MB가 된다. 즉 LV의 크기는 구성된 LE의 개수 또는 할당할 디스크 용량에 의해 결정된다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

RAID (RedundantArray of Independent Disks)

- RAID는 여러 디스크를 논리적으로 결합 및 분배하여 성능과 내장애성을 높이는 기술이다.
- RAID를 구성하면 디스크의 I/O 성능을 높일 수 있으며, 디스크 하나를 사용할 때 보다 두 개를 병렬로 사용할 때 이론상 두 배 빠른 디스크를 읽고 쓸 수 있게 되며, 이론상으로 8개를 사용하면 여덟 배의 속도를 가지게 되고 이와 같은 디스크 수를 스트라이핑 개수라 한다.
- 기본적으로 RAID 레벨은 0, 1, 2, 3, 4, 5, 6으로 일곱 가지가 존재하며 또한 RAID0와 다른 RAID 레벨을 조합한 RAID10(1+0), RAID(5+0), RAID(6+0)으로 활용된다.

RAID	설명	용도
0	내장애성이 없는 디스크 어레이(스트라이핑)	디스크 I/O 성능을 높여야 할 때 사용되며 내장 애성이 낮고, 주로 로그 집계 등의 임시 저장 영역에 사용
1	이중화(미러링)	내장애성이 높으며, 운영체제가 설치된 파티션 등에 사용
2	비트 단위 전용 오류 보정 부호 드라이브	거의 사용되지 않음
3	비트/바이트 단위 전용 패러티 드라이브	거의 사용되지 않음
4	블록 단위 전용 패러티 드라이브	거의 사용되지 않음
5	블록 단위 전용 패러티 정보기록	저장 용량을 넉넉하게 확보하고자 할 때 사용하며, 주로 파일서버나 로그 저장에 사용
6	블록 단위에서 두 가지 패러티 정보기록	RAID5와 용도는 같지만 RAID5보다 내장애성이 높음
10	RAID1을 스트라이핑	내장애성과 디스크 I/O성능을 모두 만족해야 할 때 사용하며, 주로 데이터베이스 등에 사용
50	RAID5를 스트라이핑	저장 용량확보와 디스크 I/O 성능을 모두 만족해야할 때 사용되며 주로 파일 서버나 로그 저장에 사용
60	RAID6를 스트라이핑	저장 용량확보와 디스크 I/O 성능을 모두 만족해야할 때 사용되며 주로 파일 서버나 로그 저장에 사용

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

500GB 디스크를 12개 사용할 때의 실제 용량

RAID	계산식	실제 용량
0	500GB X 12개	6TB
1	구성 불가능, RAID 10으로 구성	X
5	500GB X (12-1)개	5.5TB
6	500GB X (12-2)개	5TB
10	500GB X (12/2)개	3TB
50	500GB X ((3-1)X4)개	4TB
50	500GB X ((4-1)X3)개	4.5TB
50	500GB X ((6-1)X2)개	5TB

Linux Server 마운트, 파티션, RAID 설정


- 기본적인 Linux Server에서의 LVM의 작업순서

System Partition ID 변경(fdisk CMD) -> PV 생성 -> VG 생성 -> LV 생성 -> F/S 생성 -> 마운트

- 예시

1. fdisk /dev/sdc (Partition ID 83 -> 82)
2. pvcreate /dev/sdc1
3. vgcreate vg1 /dev/sdc1
4. lvcreate -l 100%FREE -n lv1 vg1
5. mkfs.ext3 /dev/vg1/lv1
6. vi /etc/fstab
7. mkdir /MyDir
8. mount /MyDir


- 삭제의 경우 역순으로 진행

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Linux Server 마운트, 파티션, RAID 명령어

- 기본적인 파티션, 용량 명령어


구분	옵션	설명
df [옵션] [경로]		옵션 및 경로를 지정안해두 되며 하드 디스크의 남은 용량을 확인
	-a	0 블록 파일 시스템을 포함하며 출력
	-B=Size	Size를 블록 단위로 정하여 용량에 표시
	-h	사람이 읽기 편하게 출력
	-i	inode의 남은 공간, 사용 공간, 사용 퍼센트를 출력
	-T	형태에 맞춰 각각의 파티션 정보를 출력
du [옵션] [경로]		디렉토리 별 사용량을 확인
	-a	모든 파일들의 기본정보를 출력
	-b	표시단위를 KB대신 Byte로 출력
	-k	표시단위를 KB단위로 출력
	-h	사용량을 1k, 234M, 2G 등의 형태로 출력
	-c	모든 파일의 디스크 사용정보와 합계를 출력
	-x	체크하는 경로안에 다른 시스템이 있으면 생략하고 출력
	-D	심볼릭 링크 파일이 있을 경우 원본 값을 출력
fdisk 입력후 [옵션]	-l	알려진 파티션 형태의 목록을 출력
	-n	새로운 파티션 추가
	-p	파티션 테이블 출력
	-t	파티션의 시스템 id 변경
mkfs [옵션] [장치]		파티션한 하드디스크를 포맷할 때 사용
	-t [타입]	[장치]를 타입에 맞춰 포맷 (기본값 ext2)
mount [-t 옵션] 파일시스템 [-o 옵션] [장치명] [디렉토리명]		[디렉토리명]에 [장치명]을 마운트
	[-t ext2, 3, 4]	ext2, ex3, ext4중 설정에 맞춰 마운트 수행(기본값 auto)
	[-o rw, suid, exec, auto,]	기본값 default이며 잘 사용하지 않음

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


	nouser, async]	
umount [장치명]		[장치명]을 unmount
resize2fs [장치명]		용량 변경사항을 확인
e2fsck -f [장치명]		[장치명]을 모든 상태에 대해서 점검
/etc/fstab 마운트 된 장치들이 기록 (자동으로 기입되지 않음) ex) [/dev/mapper/centos-root] [/] [xfs] [defaults] [0] [0]	[/dev/mapper/centos-root]	첫번째 필드: 디바이스명
	[/]	두번째 필드: 마운트 포인트
	[xfs]	세번째 필드: 파일 시스템의 종류
	[defaults]	네번째 필드: 파일 시스템의 고유한 옵션 default: 읽고 쓰기 가능 noauto: 명시적으로 지정하지 않는 한 마운트 X ro: 읽기 전용을 표시 usrquota: 일반 사용자에게 디스크 사용량을 제한
	[0]	다섯번째 필드: dump 0이면 덤프 X, 1이면 덤프 O
	[0]	여섯번째 필드: fsck 무결성 체크 0이면 체크, 1이면 체크를 수행

- PV, VG, LV 구성

구분	예시	설명
pvcreate [장치명1 장치명2..] (PV 생성)	pvcreate /dev/sda1	/dev/sda1 PV생성
	pvcreate /dev/sda2 /dev/sdb3	/dev/sda2, /dev/sdb3를 개별로 PV생성
pvremove [장치명1 장치명2..] (PV 삭제)	pvremove /dev/sda1	/dev/sda1로 구성된 PV를 제거
	pvcreate /dev/sda1 /dev/sdb2	/dev/sda1 과 /dev/sdb2로 구성된 PV를 제거
pvs		생성된 PV의 정보를 출력

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

pvscan		생성된 PV의 정보를 출력
pvdisplay [장치명]	pvdisplay /dev/sda1	/dev/sda1로 구성된 PV를 출력
lvmdiskscan		생성된 PV의 정보를 출력
vgcreate [vg명] [장치명]	vgcreate vg1 /dev/sda1	/dev/sda1를 vg1 이름으로 vg를 생성
	vgcreate vg1 /dev/sda1 /dev/sdb1	/dev/sda1과 /dev/sdb1를 vg1 이름으로 VG를 생성
vgremove [vg장치명 or vg명]	vgremove vg1	vg1 이름의 VG를 삭제
	vgremove /dev/vg1	
vgrename [vg명 or vg장치명] [바뀔 이름]	vgrename vg1 vg2	vg1의 vg명을 vg2이름으로 변경
	vgrename /dev/vg1 vg2	
vgs		vg에 대한 정보를 출력
vgscan		
vgdisplay [vg명]	vgdisplay vg1	vg1의 정보를 출력
lvcreate [-L or -l] [용량] [vg명] -n [lv명]	lvcreate -L 10G vg1	vg1에서 10G의 LV를 생성 (-n을 안쓸시 자동이름 생성)
	lvcreate -L 10G vg1 -n lv1	vg1에서 10G의 lv1을 생성
	lvcreate -L 1500M -n lv2	vg1에서 1500의 lv1을 생성
	lvcreate -l 60%VG -n lv1 vg1	vg1의 용량의 60%에 해당하는 lv1을 생성
	lvcreate -l 100%FREE -n lv1 vg1	vg1의 남은 모든 여유공간의 해당하는 lv1을 생성
lvremove [장치명]	lvremove /dev/vg1/lv1	lv1의 LV를 삭제
lvrename [장치명 or lv명] [바뀔 이름]	lvrename /dev/vg1/lv1 /dev/vg1/lv2	lv1의 이름을 lv2로 변경
	lvrename lv1 lv2	
lvextend [-L or -l] [용량] [LV]	lvextend -L 12G /dev/vg1/lv1	lv1의 용량을 12G으로 변경
	lvextend -L +1G /dev/vg1/lv1	lv1의 용량을 +1G만큼 높임
	lvextend -l +100%FREE /dev/vg1/lv1	lv1에게 남은 모든 vg1의 용량을 할당
lvreduce [-L or -l] [용량] [LV]	lvreduce -L -3G /dev/vg1/lv1	lv1의 용량을 3G만큼 줄임
	lvreduce -l -3	lv1의 용량을 3 LE만큼 감소

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

	/dev/vg1/lv1	
--	--------------	--


- RAID 구성

Striping 구성
lvcreate [-L or l] [용량] [-i num] [-l size] [-n] [장치명] [vg명]
lvcreate -L 1G -i 2 -l 64 -n Str-LV vg1
[-L or l]: 용량 지정 [-i 옵션]: number of stripes [-l]옵션: stripe size(기본값 64KB) 2개의 PV에 걸쳐서 stripe 크기는 64KB 크기로 vg1에 Str-LV로 구성
Mirroring 구성
lvcreate [-L or l] [용량] [-m num] [-n] [장치명] [vg명]
lvcreate -L 1G -m 2 -n Mir-LV vg1
[-L or l]: 용량 지정 [-m 옵션]: number of copys 2개의 PV에 걸쳐서 mirror를 vg1에 Mir-LV로 구성

Winodws Server 마운트, 파티션, RAID 설정

Backup, Restore

- 백업(BackUP)이란 데이터가 손상되거나 손실될 경우르 ㄹ대비해 저장하는 데이터 사본으로, 원본 데이터는 데이터를 생성한 후에도 지우지 않음
- 백업의 주 목적은 사고가 났을 시에 데이터를 복원(Restore)하는 것이며, 현재는 local 백업 뿐

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

만이 아닌, 클라우드 등을 활용한 백업방법에 중점이 주어지고 있음

- 백업에는 여러 종류가 있으며 특징은 아래의 표와 같음


구분	설명
Full Backup	선택된 모든 폴더 및 DATA를 전체 백업하는 방식
incremental Backup	선택된 폴더를 Full Backup 이후, 변경, 추가된 Data만 백업하는 방식
Differential Backup	선택된 폴더 DATA를 Full Backup 이후 변경, 추가된 Data를 모두 포함하여 백업하는 방식

Linux Server Backup, Restore

- Linux Server에서의 백업은 기본적으로 tar을 사용하거나 dump 혹은 툴을 사용하기도 한다.


Linux Server Backup, Restore 명령어 정리

구분	옵션	설명
tar [옵션] [아카이브 명] [파일1 파일2...]	-f	대상 tar 아카이브 지정
	-c	tar 아카이브 생성(파일 묶기)
	-x	tar 아카이브파일 추출(파일 풀기)
	-v	처리되는 과정 출력
	-z	gzip 압축 적용 옵션
	-j	bzip2 압축 적용 옵션
	-t	tar 아카이브에 포함된 내용 확인
	-C	대상 디렉토리 경로 지정
	-A	지정된 파일을 tar 아카이브에 추가
	-d	tar 아카이브와 파일 시스템 간 차이점 검색
	-r	tar 아카이브에 마지막에 파일들 추가
	-u	tar 아카이브에 마지막에 파일들 추가
	-k	tar 아카이브 추출 시, 기존 파일 유지
	-U	tar 아카이브 추출 전, 기존 파일 삭제
	-w	모든 진행과정에 대해 확인요청
	-e	에러 발생시 중지

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

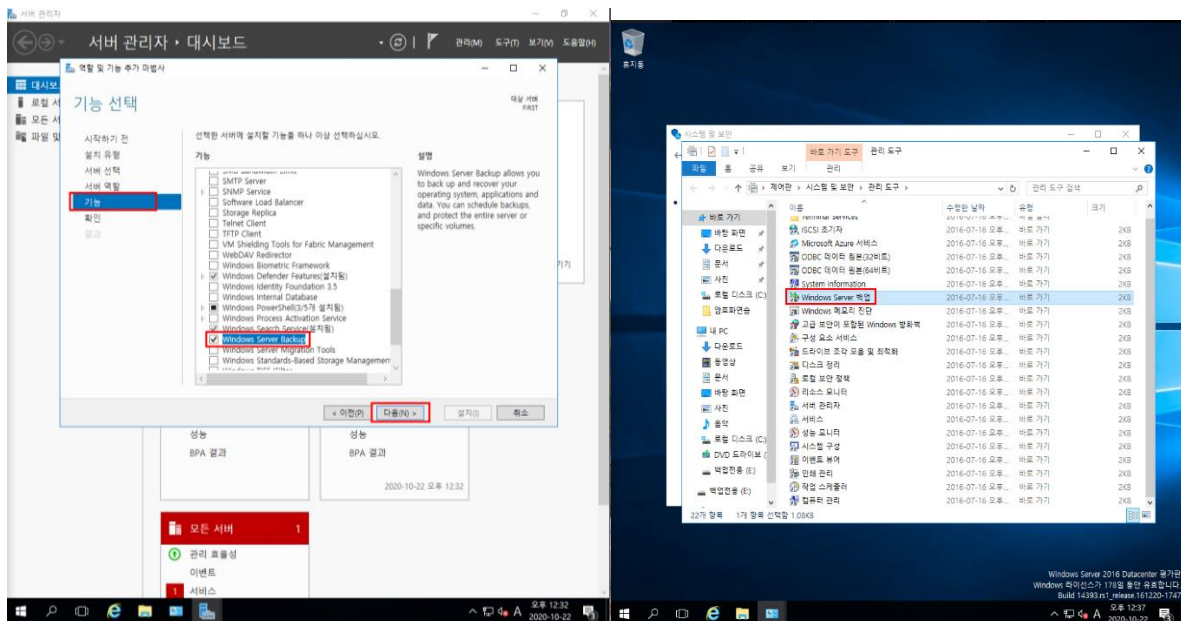
tar 사용예시	-cvf	파일을 지정하여 묶음
	-tvf	아카이브 내용을 확인
	-zcvf	tar로 묶고 gzip으로 압축
	-zxvf	gzip으로 압축된 tar 아카이브를 현재 디렉토리에 풀기
	-jcvf	현재 디렉토리를 tar로 묶고 bzip2로 압축
	-jcvf	bzip2로 압축된 tar로 묶고 bzip2로 압축
	-jxvf	bzip2로 압축된 tar 아카이브를 현재 디렉토리에 풀기
	-cvfw	tar 아카이브를 묶거나 풀 때 파일 별 진행 여부를 확인

전체백업
tar [옵션] [tar_name] [옵션] [path]
tar -cvpzf backup.tar.gz --exclude=backup.tar.gz --exclude=/proc --exclude=/mnt --exclude=/sys --exclude=/tmp --exclude=/dev --exclude=/media --exclude=/run /
[-cvpzf]: 상단의 표를 참고 [tar_name]: 백업할 파일의 이름 [--exclude=...]: 백업시 제외할 디렉토리나 파일 exclude에 제외한 폴더를 제외하고 전체(/) 백업파일을 backup.tar.gz 생성
증분백업
tar [옵션] [Backup file] -g [list name] [path]
tar -cvpzf backup.tar.gz -g list /
[-cvpzf]: 상단의 표를 참고 [Backup File]: 전체백업으로 생성된 파일 [list name]: 증분백업이 기록될 list파일
복구
tar [옵션] [Backup] -C /
tar -xvpzf backup.tar.gz -C /
[-C]: 디렉토리를 지정 backup.tar.gz의 아카이브를 /에 압축해제


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

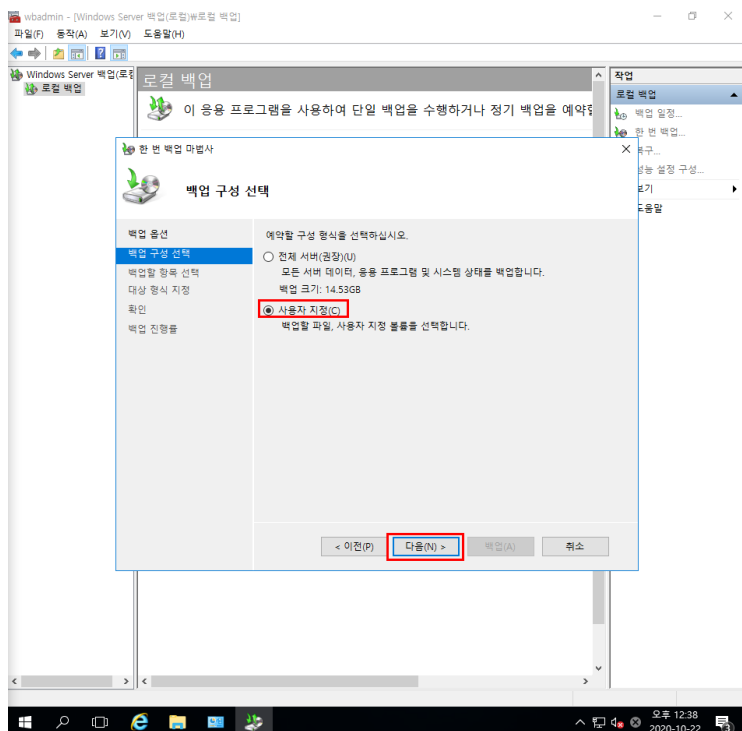
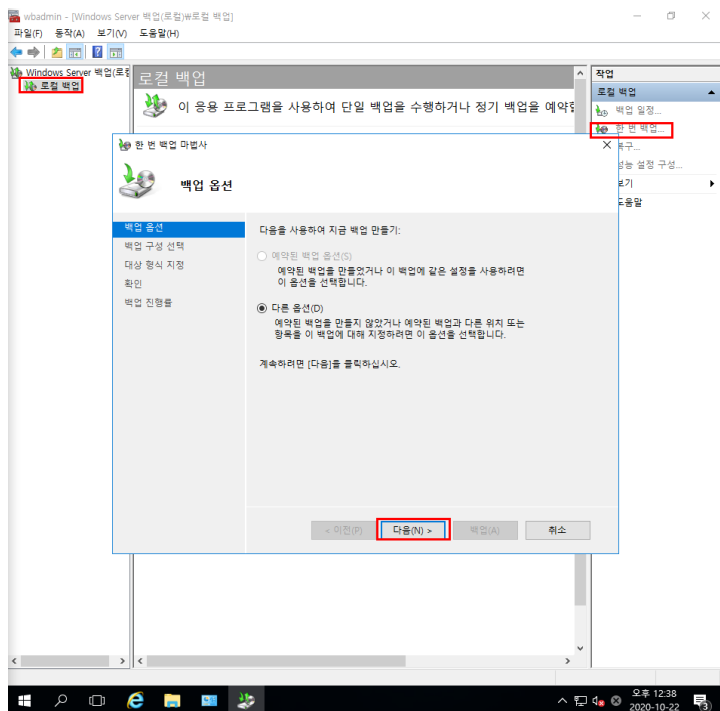
Windows Server Backup, Restore


- Windows Server는 자체적으로 백업 기능을 제공하며, 백업은 대체로 별도의 하드디스크나 다른 컴퓨터의 공유 폴더에 하는 것이 보다 효과적
- Windows Server는 2012버전부터 ReFS 파일시스템의 백업도 지원
- 그림과 같이 서버 관리자에서 Windows Server Backup이라는 자체 기능을 함유
- 설치가 완료되면 제어판 → 시스템 및 보안 → 관리도구 → Windows Server 백업기능 사용가능

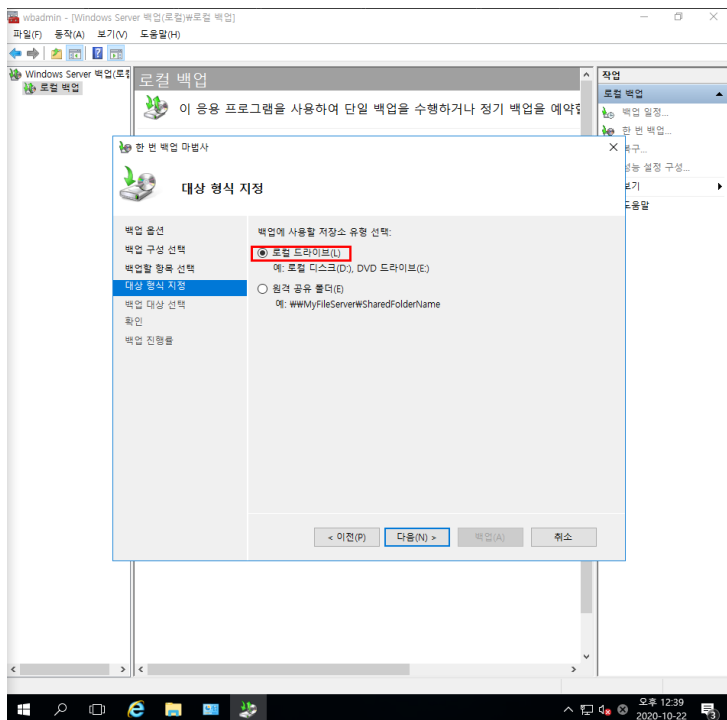


- Windows Server는 Azure Cloud와 연동하여 현재는 클라우드 상에 백업파일을 저장하는 추세
- 그림과 같이 기본적인 설명에 따라 편리하게 백업을 진행할 수 있음

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	



	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	




현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window08/>

FTP Server

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- FTP(File Transfer Protocol) Server는 TCP/IP 프로토콜을 응용하여 서버와 클라이언트 사이의 파일을 전송하기 위한 프로토콜로, TCP/IP 프로토콜 테이블의 응용 계층에 속한다.

Linux FTP Server

- Linux Server에서의 FTP Server는 L-1을 Server, L-2를 Client로 설치한다.


```
(L-1)$ sudo yum install vsftpd -y
# vsftpd 패키지를 설치한다.

(L-1)$ systemctl start vsftpd
# vsftpd 서비스를 시작한다.

(L-1)$ firewall-cmd --zone=public --permanent --add-port=21/tcp
(L-1)$ firewall-cmd --zone=public --permanent --add-service=ftp
(L-1)$ firewall-cmd --reload
# 방화벽에 ftp 서비스를 허용한다.

(L-1)$ vi /etc/vsftpd/vsftpd.conf
local_enable = YES           # 로컬 로그인을 허용
write_enable = YES           # 파일 시스템을 변경하는 FTP 명령을 활성화
anon_upload_enable=YES       # 익명자의 업로드 명령을 활성화
anon_mkdir_write_enable=YES  # 익명자의 폴더생성 명령을 활성화
local_umask = 022            # 로컬 사용자를위한 파일 작성을위한 umask 값
dirmessage_enable = YES     # 사용자가 처음 입력 할 때 메시지 표시 가능 새로운
디렉토리
xferlog_enable = YES        # 로그 파일은 업로드 및 다운로드를 상세하게 관리
connect_from_port_20 = YES  # PORT 스타일 연결을 위해 서버 시스템에서 포트 20 (
ftp-data) 사용
xferlog_std_format = NO     # 표준 로그 파일 형식 유지 (기본값 YES) -
> 기본값은 로그를 볼 수 없다.
listen = NO                  # 독립 실행 형 모드에서 vsftpd 실행 방지
listen_ipv6 = YES            # IPv6 소켓에서 수신 대기
pam_service_name = vsftpd   # PAM 서비스 사용
userlist_enable = YES        # vsftpd 가 사용자 이름 목록 로드 허용.
tcp_wrappers = YES           # tcp wrappers 허용
xferlog_file=/var/log/xferlog # log 를 활성화 한다. (기본은 비활성)
vsftpd_log_file=/var/log/vsftpd.log

# 대부분은 기본설정을 사용하면 되고, anonymous 에 대한 허용여부만 결정하면 된다.
# vsftpd_log_file=/var/log/vsftpd.log 를 추가하고, xferlog_std_format = NO 를
변경하지 않으면 log 가 안보인다.
# 접속 제한 유저 리스트는 수정 및 확인은 /etc/vsftpd/vsftpd_list 에서 가능하다.
```


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
(L-1)$ yum provides -y /usr/sbin/semanage
```

```
# semanage 명령을 설치한다.
```

```
(L-1)$ setsebool -P allow_ftp_full_access on
```

```
# setsebool 이 기본 비활성화이므로, 활성화 시켜준다.
```

```
# semanage 명령을 통해 FTP 가 사용자의 홈 디렉토리를 읽고 쓸 수 있도록 SELinux 정책을 설정한다.
```

```
(L-1)$ chcon -R 777 /var/log/pub
```

```
# 확인을 위해 익명사용자가 사용할 수 있도록 모든 권한을 부여한다.
```

```
(L-1)$ mkdir /var/ftp/pub/hello
```

```
(L-1)$ echo Hello! > hi.txt
```

```
# FTP 기본폴더에서 폴더 및 파일을 생성한다.
```

```
(L-1)$ systemctl restart vsftpd
```

```
# vsftpd 를 재구동한다.
```

```
(L-2)$ sudo yum install vsftpd -y
```

```
# vsftpd 를 설치한다.
```

```
(L-2)$ echo bye! > bye.txt
```

```
# put 테스트를 위해 파일을 생성한다.
```

```
(L-2)$ ftp L1(FTP Server 192.168.10.11)
```

```
ftp> anonymous
```

```
ftp> mail
```

```
# 익명의 사용자로 접속한다.
```

```
(L-2)$ pwd
```

```
(L-2)$ ls
```

```
# 현재 경로 및 디렉토리를 표시한다. (미리 생성해둔 hello 디렉토리와 hi.txt 를 확인할 수 있다.
```


```
(L-2)$ put bye.txt
```

```
# 미리 생성해둔 bye.txt 를 업로드한다.
```

```
(L-2)$ get hi.txt
```

```
(L-2)$ mkdir /pub/Hello/Hi
```

```
# 파일을 다운로드 하고 폴더 생성해본다.
```


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	




```
(L-2)$ ctrl + z
(L-2)$ ls
# 다운로드를 확인할 수 있다.

(L-1)$ cat /var/log/xferlog
# 아래와 같이 ftp 서버에서의 활동들을 확인이 가능하다.
Tue Nov  3 00:29:23 2020 1 ::ffff:192.168.10.12 7 /pub/Hi.txt b _ o a mail ftp 0 * c
Tue Nov  3 00:31:21 2020 1 ::ffff:192.168.10.12 5 /pub/bye.txt b _ i a mail ftp 0 * c
```

← → ↻ ⚠ 주의 요함 | ftp://192.168.10.11/pub/

/pub/의 색인

 [상위 디렉터리]

이름	크기	수정된 날짜
 Hello/		20. 11. 3. 오후 2:26:00
 Hi.txt	7 B	20. 11. 3. 오후 2:28:00
 bye.txt	5 B	20. 11. 3. 오후 2:31:00


위와 같이 윈도우 환경에서도 ftp 서버에 접속하여 확인이 가능하다.

Windows FTP Server

현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window12/>

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

TELNET Server


- TELNET Server는 원격 접속 서비스로서 특정 사용자가 네트워크를 통해 다른 컴퓨터에 연결하여 그 컴퓨터에서 제공하는 서비스를 받을 수 있도록 하는 인터넷 표준 프로토콜
- TELNET Server를 이용하면 네트워크에 있는 컴퓨터를 자신의 컴퓨터처럼 파일 전송, 생성, 디렉토리의 생성, 삭제 등을 자유롭게 할 수 있지만, 보안상의 문제로 SSH로 대체되는 추세
- SSH와 부각되는 주 차이는 서버와 클라이언트 간의 통신 간엔 TELNET은 byte스트림 형식으로 데이터를 주고 받는 반면 SSH는 암호화를 통해 통신을 수행

Linux TELNET Server

```
(L-1)$ yum list telnet*
Available Packages
telnet.x86_64                               1:0.17-65.el7_8
updates
telnet-server.x86_64                       1:0.17-65.el7_8
updates
# TELNET Sever 에 패키지를 확인
# L1 은 서버로 L2 는 클라이언트 용도로 진행

(L-1)$ yum -y install telnet-server
(L-1)$ rpm -qa | grep telnet*
telnet-server-0.17-65.el7_8.x86_64
# TELNET Server 를 다운로드하고 확인

(L-1)$ vi /etc/xinetd.d/telnet
# TELNET 은 xinetd 라는 슈퍼 데몬에 의해 동작
# 설정파일을 생성하여 설정을 진행
# default : on
# description : The telnet server serves telnet sessions; it uses W
# unencrypted username/password pairs for authentication.
service telnet
{
    flags                = REUSE // 소켓 작동 시 사용하고 참조하는 정보
    socket_type          = stream // Telnet 서비스의 소켓 적용 정보 (stream : TCP 소켓
dgram : UDP >소켓)
    wait                = no // 대기 시간 제공 여부 (no : 프로세스 추가하여 제공
yes : 1 개만 연>결 가능)
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```

user          = root // 서비스를 시작하는 권한자
server        = /usr/sbin/in.telnetd // 서비스를 처리할 프로그램 위치
log_on_failure += USERID // 로그인 실패 시 USERID 를 포함해 로그 정보를
쌓는다
disable       = no // 서비스 제공 여부 (no : 제공 한다. yes : 제공하지
않는다)
}

(L-1)$ vi /etc/securetty
console
pts/1          #추가
pts/2          #추가
pts/3          #추가
vc/1
...
# 기본적으로 TELNET 은 root 사용자가 거부되어 추가적인 설정을 수행

(L-1)$ systemctl start telnet.socket
(L-1)$ systemctl status telnet.socket
(L-1)$ netstat -anlp | grep 23
# telnet server 를 시작하고 상태와 포트를 확인

(L-1)$ firewall-cmd --zone=public --permanent --add-port=23/tcp
(L-1)$ firewall-cmd --zone=public --permanent --add-service=telnet
(L-1)$ firewall-cmd --reload
# 방화벽을 설정

(L-2)$ yum -y install telnet
# 클라이언트 사용을 위해 L2 에서 telnet 을 다운로드


(L-2)$ telnet L1(도메인 혹은 IP) 23(PORT)
Trying 192.168.10.11...
Connected to L1.
Escape character is '^]'.

Kernel 3.10.0-1127.el7.x86_64 on an x86_64
L1 login: root
Password:
Last login: Tue Nov 3 00:02:34 from 192.168.10.1
# L1 의 23 번 포트로 telnet server 에 접속

(L-2)$ [root@L1 ~]# echo TELNET! > TELNET.txt
# 확인을 위해 파일을 생성

(L-1)$ cat TELNET.txt
TELNET!
# L1 에서 파일이 생성된 것을 확인할 수 있음

```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Windows TELNET Server

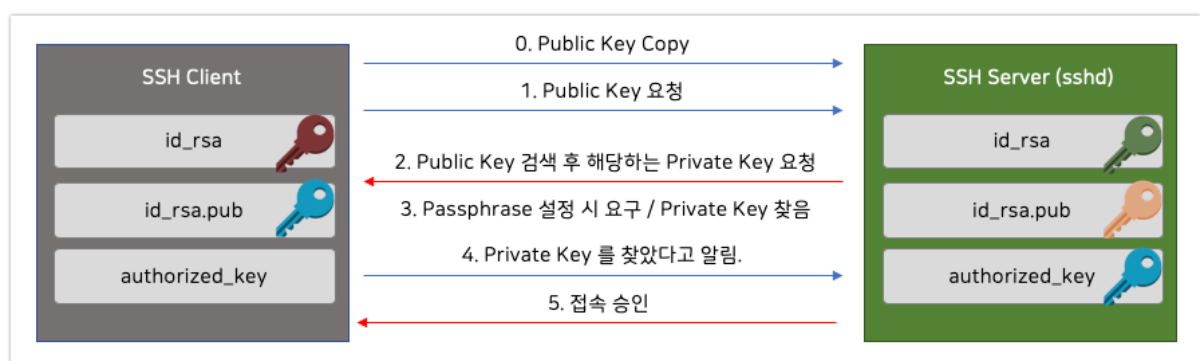
현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!


<https://mung0001.github.io/docs/system/window/window09/>

SSH SERVER

- SSH(Secure Shell Protocol) Server은 네트워크 프로토콜 중 하나로 컴퓨터와 컴퓨터가 인터넷과 같은 통신을 할 때 보안적으로 안전하게 통신을 하기 위해 사용하는 프로토콜로 TELNET과의 유사한 역할을 수행하지만, 보안적인 측면에서 강점을 가지고 있다.
- 대표적으로 GITHUB의 푸시 등의 작업 또한 SSH로 진행된다.



- SSH는 기본적으로 암호를 사용할 수도 있지만, 주로 한 쌍의 키를 사용하며, 이는 Private Key, Public Key로 복호화가 되어 있으며, 서버측에는 Private Key가 저장되어 있어, 이를 사용하기 위해서는 이에 맞는 한 쌍의 키 Public Key가 반드시 필요하다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Linux SSH Server

```
(L-1)$ rpm -qa | grep ssh
openssh-server-7.4p1-21.el7.x86_64
libssh2-1.8.0-3.el7.x86_64
openssh-7.4p1-21.el7.x86_64
openssh-clients-7.4p1-21.el7.x86_64
# 기본적으로 centos 의 경우 ssh 서버와 클라이언트가 설치되어있다.

(L-1)$ systemctl status sshd
(L-1)$ netstat -anlp | grep :22
# 작동중인 서버와 포트를 확인할 수 있다.


(L-1)$ vi /etc/ssh/sshd_config
PermitRootLogin No           # 루트 로그인 비활성화 (기본값 활성화)
AllowUsers User01 User02     # 사용자 접속 제한
AllowUsers User01@172.16.12.25 # 사용자 접속 제한시 IP 연동
Protocol 2                   # SSH Protocol v2 만 허용 (기본값 1, 2)
Port 22                      # SSH Port 변경 (기본 22), 적용시 방화벽적용
필요

PubkeyAuthentication yes     # 퍼블릭 키 기반의 인증 (기본 패스워드)
PasswordAuthentication no    # 퍼블릭 키 기반의 인증 (기본 패스워드)

MaxAuthTries 6               # 접속 가능 횟수 (기본 6 회)
LoginGraceTime 120           # 로그인 대기시간 (기본 120 초)

(L-1)$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:665a304p3+FHauYcQCOg2qgV8sE+dqdVILDp7ZvZ4kw root@L1
The key's randomart image is:
# L1 에서 키 페어 한 쌍을 생성한다.
# 생성된 private Key 는 /root/.ssh/id_rsa 에서 확인할 수 있다.
# 생성된 private Key/root/.ssh/id_rsa.pub 에서 확인할 수 있다.

(L-1)$ ssh-copy-id root@L2
# 생성한 Public Key 를 L2 에게 전달한다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

적용이 완료되면 초기에 패스워드를 입력 후, 그 후로는 Private Key 를 통해 접속하여 패스워드의 입력이 필요없어진다.

```
(L-2)$ ssh L1
```

키를 등록 후에도 L2 에서 L1 은 서로 프라이빗, 퍼블릭 키를 가지고 있지만 L1 이 Private (Client), L2 가 Public(Server)를 가지고 있기 때문에 L2 에서 L1 으로 접속하기 위해서는 키가 아닌 패스워드가 사용된다.

```
(L-1)$ echo SCP! > ~/SCP.txt
```

```
(L-1)$ scp ~/SCP.txt L2:~/
```


```
SCP.txt 100% 5 1.7KB/s 00:00
```

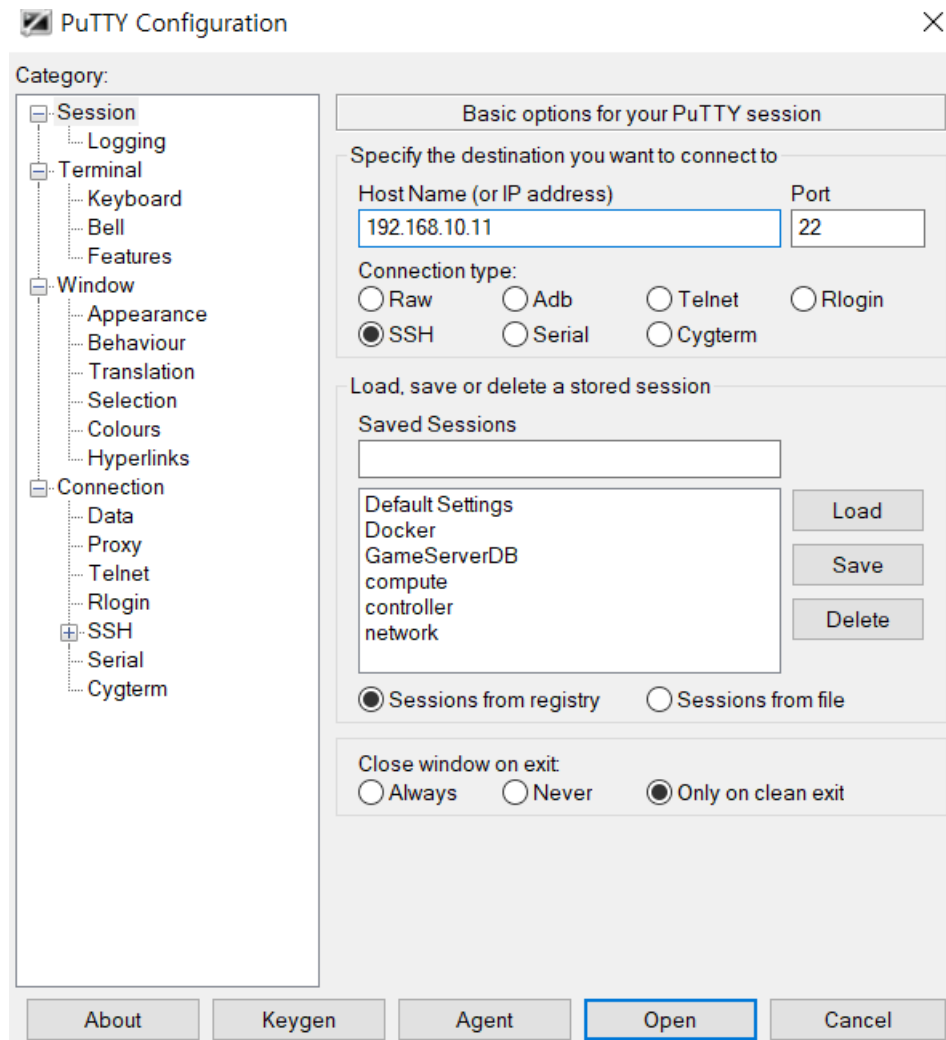
ssh 와 비슷하에 scp(secure copy (remote file copy program))을 통해 호스트 간의 파일을 주고 받을 수 있다.

```
(L-2)$ cat ~/SCP.txt
```

```
SCP!
```

L-2 에서 확인할 수 있다.

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	




- 이 외에도 Windows 환경에서도 Putty 등의 프로그램을 통해 보다 쉽게 SSH Server를 활용할 수 있다.

Windows SSH Server

현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window09/>

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

NTP SERVER

- NTP(Network Time Protocol) Server는 Network 상에 연결된 장비와 장비 간에 시간 정보를 동기화하기 위한 프로토콜을 의미
- NTP는 계층적인 구조를 가지고 있으며, 각각의 계층은 상위 계층으로부터 시간을 동기화하여 다른 서버나 장비들에게 시간을 동기화 할 수 있음

Linux NTP Server


```
(L-1)$ yum -y install ntp
# L-1 에 NTP 서버를 설치

(L-1)$ vi /etc/ntp.conf
...
# server 0.centos.pool.ntp.org iburst           # 주석처리
# server 1.centos.pool.ntp.org iburst           # 주석처리
# server 2.centos.pool.ntp.org iburst           # 주석처리
# server 3.centos.pool.ntp.org iburst           # 주석처리
server L1                                       # L1 을 NTP 서버로 사용하도록 추가
restrict 192.168.10.0 mask 255.255.255.0 nomodify notrap
# NTP IP 허용대역 설정
...
#

(L-1)$ firewall-cmd --zone=public --permanent --add-port=123/udp
(L-1)$ firewall-cmd --zone=public --permanent --add-service=ntp
(L-1)$ firewall-cmd --reload
# 방화벽을 설정

(L-1)$ systemctl restart ntpd
(L-1)$ systemctl status ntpd
(L-1)$ netstat -anlp | grep :123
# NTP Server 의 실행과 포트를 확인할 수 있다.

(L-1)$ ntpq -p
remote          refid          st t when poll reach  delay  offset  jitter
=====
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```

L1          .INIT.          16 u   -   64   0   0.000   0.000   0.000
# 동기화된 시간을 확인할 수 있다.

(L-2)$ yum -y install ntp
# L2 클라이언트에서 L1 NTP 서버에서 시간을 받아오기 위해 NTP 서버를 설치한다.

(L-1)$ vi /etc/ntp.conf
...
# server 0.centos.pool.ntp.org iburst           # 주석처리
# server 1.centos.pool.ntp.org iburst           # 주석처리
# server 2.centos.pool.ntp.org iburst           # 주석처리
# server 3.centos.pool.ntp.org iburst           # 주석처리
server L1                                       # L1 을 NTP 서버로 사용하도록 추가
# 만약 뒤에 prefer 을 붙이면 우선순위로 설정된다.
...
#

(L-2)$ systemctl restart ntpd
# ntp 서비스를 재시작한다.


(L-2)$
remote          refid          st t when poll reach   delay   offset  jitter
=====
L1          .INIT.          16 u  116 1024    0     0.000   0.000   0.000
# L1 의 NTP 서버에서 시간정보를 업데이트 받는 것을 확인할 수 있다.
# 이 외에도 ntpdqte [IP]를 통해 수동을 동기화 시킬 수 있다.

```

Windows NTP Server

NFS SERVER

- NFS(Network File System)은 용어 그대로 네트워크 기반의 파일 시스템을 의미한다. 공유된 원

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

격의 호스트 파일을 다른 리눅스 시스템이 로컬에서 사용할 수 있도록 구현한 방식이며, NFS v2부터 현재 v4에 이르기 까지 널리 사용되어 왔으며, 손쉽게 원격의 파일 시스템을 공유할 수 있다는 점에서 높은 편의성을 가지고 있다.

- NFS를 사용하게 되면 Linux Server가 네트워크를 통해 다른 리눅스/ 운영체제 클라이언트와 디렉토리를 공유할수 있으며, NFS 서버는 해당 디렉토리를 외부로 보내고, NFS 클라이언트는 해당 디렉토리를 마운트하는 형태로 NFS는 현재 주로 v3, v4가 사용된다.

- NFS v3 (1995 년)


- NFS version3 는 Async 방식의 Writed 모드를 지원
- 64bit File Size 의 Offset 을 지원하므로 2GB 이상의 파일 액세스 가능
- TCP/UDP Port 2049 를 지원하지만 실제 IP Network 에서 UDP 기반 Stateless 방식으로 연결됨
- UDP 방식 전송으로 인해 신뢰성에 이슈가 있을 수 있음.

- NFS v4 (2003 년)

- NFS version4 는 rpcbind 서비스가 필요없고 ACL 을 지원하며 Statuful 방식 운영이 가능함.
- RHEL/CentOS 6 이상에서는 모두 지원하며, 기본 mounting 될 때 NFS v4 가 기본으로 구성됨.
- TCP 를 사용하기 때문에 신뢰성을 보장 받을 수 있음.
- TCP Port 2049 를 지원
- 2010 년 NFS v4.1 발표, 현재 NFS v4.2 가 최신

- NFS RPC 프로세스 및 주요서비스

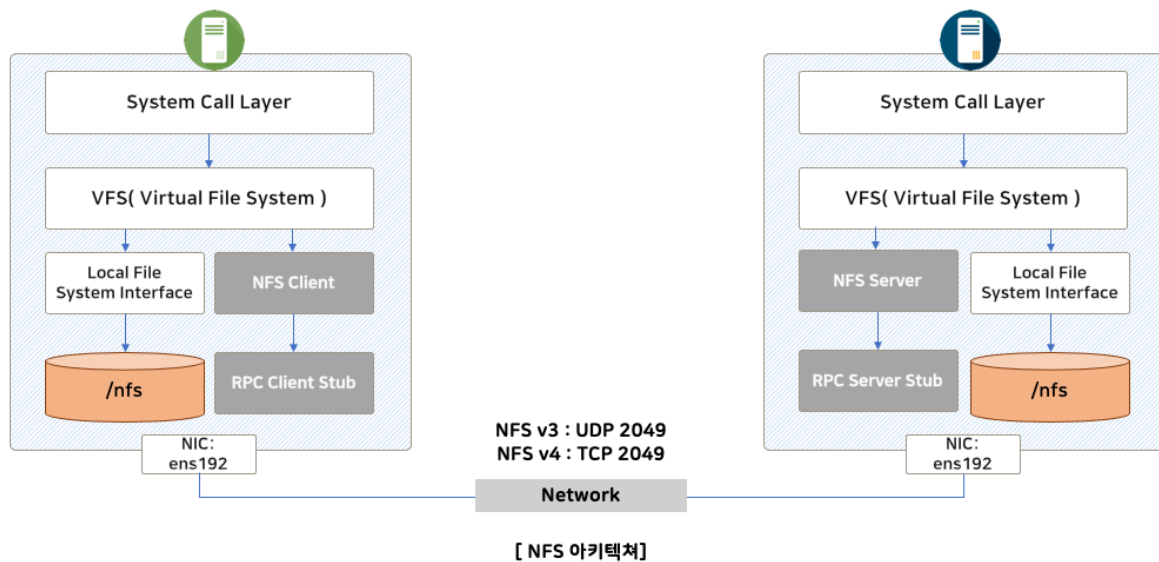
- rpc.statd : NFS 클라이언트와 서버간의 모니터링 프로토콜
- rpc.mountd : NFS v3 클라이언트의 마운트 요청을 서버 측에서 구현하는 NFS mount 데몬
- rpc.idmapd : NFSv4 이름과 로컬 UID 및 GID 매핑

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

- rpc.rquotad: 원격 사용자에게 대한 사용자 할당량 정보 제공
- rpcbind : RPC 프로그램 번호를 범용 주소를 변환
- nfs-lock / rpc-statd : NFS 파일 Lock


- NFS 주요 구성 파일

- /etc/exports : 기본 구성 파일이며, 원격 호스트로 내보낼 파일 시스템을 제어하고 옵션을 지정
- /etc/fstab : 시스템이 부팅될 때 NFS 디렉토리를 포함하여, 어떤 파일 시스템이 마운트 되지는 제어하는 데 사용
- /etc/sysconfig/nfs : 필요한 RPC 서비스가 실행되는 포트를 제어하는 데 사용
- /etc/hosts.allow, /etc/host.deny : TCP wrapper 기반으로 NFS 서버 접근 제어로 사용.



Linux NFS Server

```
(L-1)$ rpm -qa | grep nfs-util
(L-1)$ yum -y install nfs-utils rpcbind
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
# nfs 유틸의 설치유무를 확인하고 설치한다.

(L-1)$ for NFS in rpcbind nfs-server rpc-statd;
> do  systemctl start $NFS
> done
(L-1)$ systemctl status rpcbind ...
# nfs 관련 패키지를 시작하고 각 서비스의 상태를 확인한다.

(L-1)$ mkdir /nfs
(L-1)$ chmod 777 /nfs
# NFS 용 디렉토리를 생성하고, 퍼블릭 상태로 권한을 부여한다.

(L-1)$ vi /etc/exports
/nfs L1 (rw,sync,no_root_squash)
/nfs L2 (rw,sync,no_root_squash)
# /etc/exports 에는 공유할 디렉터리, 공유 허가된 클라이언트 주소, 권한을 명시한다.
# rw : 공유 디렉터리에 폴더에 읽기 쓰기 권한
# sync : 해당 파일시스템에 모든 변경내용이 즉시 변경되어짐
# no_root_squash : 루트 스쿼시를 비활성화

(L-1)$ exportfs -r
# -r : /etc/exports 와 /var/lib/nfs/etab 을 동기화
# -a : /etc/exports 항목은 보내지만, 동기화는 진행하지 않음


(L-1)$ firewall-cmd --permanent --zone public --add-service=mountd
(L-1)$ firewall-cmd --permanent --zone public --add-service=rpc-bind
(L-1)$ firewall-cmd --permanent --zone public --add-service=nfs
(L-1)$ firewall-cmd --reload
(L-1)$ systemctl restart nfs-server
# 방화벽을 설정하고 nfs 서버를 재시작한다.

(L-2)$ yum -y install nfs-utils libnfsidmap
# L1(NFS Sever)에서 설정이 완료되면 L2(Client)를 설정하기 위해 nfs client 패키지를
설치한다.

(L-2)$ systemctl start rpcbind
# rpcbind 서비스를 시작한다.

(L-2)$ showmount -e L1
Export list for L1:
/nfs (everyone)
# showmount 명령을 통해 NFS 서버와 접속이 정상적으로 이루어지는 지를 확인한다.

(L-2)$ mkdir /mnt/nfs
(L-2)$ mount L1:/nfs /mnt/nfs/
(L-2)$ mount | grep nfs
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
L1:/nfs on /mnt/nfs type nfs4
(rw,relatime,vers=4.1,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,r
etrans=2,sec=sys,clientaddr=192.168.10.12,local_lock=none,addr=192.168.10.11)
# mount 할 디렉터리를 생성하고, 생성한 폴더에 마운트한다.


(L-2)$ df -h
...
L1:/nfs          47G  1.4G  46G   3% /mnt/nfs
...
# 정상적으로 마운트된 것을 확인할 수 있다.

(L-1)$ echo Hello! > /nfs/Hello.txt
(L-2)$ echo Bye! > /mnt/nfs/Bye.txt
# 각 파일을 생성하면 동일하게 공유되는 것을 확인할 수 있다.
# nfs 서버도 mount 하는 방식으로 지속적으로 유지시키기 위해서는 /etc/fstab 에 등록이
필요하다.
# 이는 Windows, Mac, Linux, Unix 가 공용으로 사용할 수 있어 활용성이 높다.
```

Windows NFS Server

DHCP Server

- DHCP(Dynamic Host Configuration Protocol) Server는 BOOTP를 확장한 프로토콜로 DHCP Discover, Offer, Request, Ack 메시지를 이용하여 서버와 클라이언트 간에 IP 주소 및 이와 관련된 다양한 정보를 클라이언트가 수신하기 위한 목적으로 사용된다.
- 이 때 주로 사용되는 프로토콜이 BOOTP(Bootstrap Protocol)이며, 해당 프로토콜을 통해 IP 주소를 포함하는 다양한 네트워크 정보를 서버측으로부터 수신받도록 하는 방식이다.
- DHCP는 UDP 68, 69를 사용하게 BOOTP의 기본이 RARP를 이용하는 방식이므로, 브로드캐스트

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

트 전송 방식을 이용하게 된다.

- DHCP 서버가 Client에 전송할 때는 Option 정보를 포함하여, 단순히 IP 정보만 수신하는 게 아닌, 다양한 정보를 수신 및 송신 할 수 있으며 하단의 정보들을 송, 수신할 수 있다.
 - Client MAC 주소
 - IP 주소
 - Option 1 : Subnet Mask
 - Option 3 : Router(Gateway)
 - Option 6 : DNS
 - Option 51 : IP Lease Time(IP 임대시간)
 - Option 54 : DHCP Server ID (DHCP Ack를 보낸 DHCP 서버 주소)
 - 이 외에도 다양한 정보들을 송수신 가능하며 [DHCP Option](#)에서 확인 가능하다.


Linux DHCP Server

```
(L-1)$ rpm -qa | grep dhcp
dhcp-libs-4.2.5-79.el7.centos.x86_64
dhcp-common-4.2.5-79.el7.centos.x86_64
# 위와 같이 기본적으로 CentOS7에서는 dhcp를 사용하기 위한 패키지가 다운로드 되어
있지만, 서버로 사용하기 위해서는 추가적인 다운로드가 필요하다.

(L-1)$ yum -y install dhcp
# L-1에서 dhcp 서버를 다운로드 한다.

(L-1)$ vi /etc/sysconfig/dhcpd
DHCPDARGS = ens32(Interface)
# dhcp 메시지를 주고 받을 인터페이스를 지정한다.

(L-1)$ vi /etc/dhcp/dhcpd.conf
option domain-name "My-CentOS.com";
option domain-name-servers ns1.My-CentOS.com, ns2.My-CentOS.com;
default-lease-time 3600;
max-lease-time 7200;
authoritative;
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
# domain name 은 "My-CentOS.com", domain-name-server 는 "My-
CentOS.com"으로 지정한다.
# default-lease-time 은 3600 초, max-lease-time 은 7200 초이다.

subnet 192.168.10.0 netmask 255.255.255.0 {
    option routers                192.168.10.2;
    option subnet-mask            255.255.255.0;
    option domain-search          "My-CentOS.com";
    option domain-name-servers    8.8.8.8;
    range 192.168.10.150 192.168.10.199;
}

# Subnet 192.168.10.0/24, Gateway 는 192.168.10.2, domain 은 "My-CentOS.com"
# nameserver 주소는 "8.8.8.8", IP 할당 범위는 "192.168.10.150~
192.168.10.199"이다.

host test1 {
hardware ethernet 00:50:56:aa:aa:01;
fixed-address 10.72.78.140;
}

# 이 파일은 기본적으로 비어 있으며, 이에 대한 샘플 파일을 vi /usr/share/doc/dhcp-
4.2.5/dhcpd.conf.example 에서 확인 할 수 있다.

(L-1)$ systemctl start dhcpd
(L-1)$ systemctl status dhcpd
# dhcp 데몬을 활성화하고, 이를 확인한다.


(L-1)$ firewall-cmd --permanent --zone=public --add-service=dhcp
(L-1)$ firewall-cmd --reload
# dhcp 서비스를 위해 방화벽을 설정한다.

(L-2)$ vi /etc/sysconfig/network-scripts/ifcfg-ens32
# DHCP 확인을 위해 L-2 클라이언트에서 네트워크에 대한 설정을 한다.

(L-2)$ systemctl restart network
(L-2)$ ip a
# 네트워크를 재 설정 후 ip a 를 입력하면 대역에 맞게 IP 를 할당받을 것을 확인 할 수
있다.

(L-1)$ cat /var/lib/dhcpd/dhcpd.leases
server-duid "W000W001W000W001'3W320W034W000W014)W334W"~";

lease 192.168.10.150 {
    starts 2 2020/11/03 08:27:52;
    ends 2 2020/11/03 09:27:52;
    cltt 2 2020/11/03 08:27:52;
```


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
binding state active;
next binding state free;
rewind binding state free;
hardware ethernet 00:0c:29:db:ec:68;
client-hostname "L2";
}
# L1 Server 에서 DHCP 를 통해 할당받은 IP 를 확인할 수 있다.
```


Windows DHCP Server

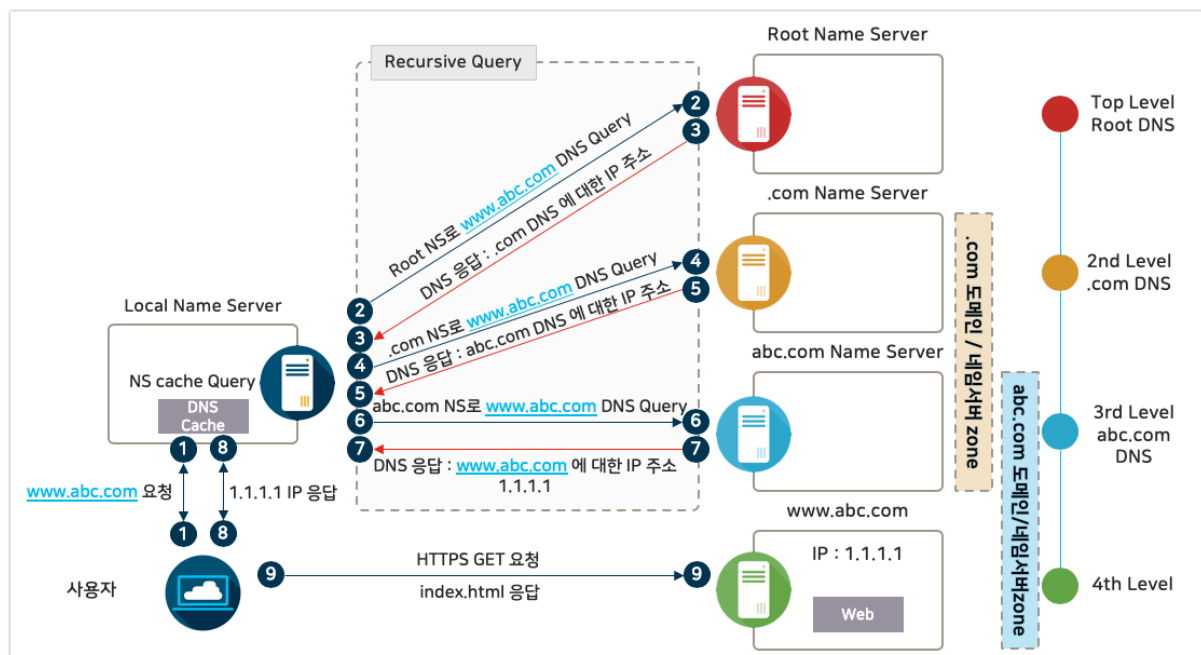
현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window14/>

DNS Server

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	




- DNS(Domain Name Service) Server는 호스트의 도메인 이름을 IP 주소로 변환하거나 또는 그 역으로 IP 주소를 호스트의 도메인 이름으로 변환 할 수 있도록 하는 서비스를 의미한다.
- DNS는 서비스를 제공하는 서버와 서비스를 제공받는 클라이언트가 페어로 이루어진다.
- 여기서는 상위 DNS와 연동되기 때문에 서버 및 클라이언트가 되기도 하며, DNS Server는 Name Service를 위한 Resolver, DataBase, Cache 기능을 가지고 있으며, 사용자는 Resolver 기능만을 가지고 있다.
- DNS는 기본적으로 53번 Port를 사용하며, UDP를 우선시하고 TCP도 사용이 가능하다.
- DNS의 프로세스는 호스트 입장에서는 찾아가고자 하는 목적지의 호스트 도메인 이름을 곧바로 DNS에 질의하는 것이 아닌, 호스트 서버의 Cache, Host, DNS 서버를 조회한다.

Linux DNS Server

```
(L-1)$ hostnamectl status
Static hostname: L1

(L-1)$ cat /etc/hosts
192.168.10.11    L1
192.168.10.12    L2
192.168.10.13    L3
# L-1 은 DNS Server로서의 설정을 위해 hostname과 host 목록을 확인 및 설정한다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
(L-1)$ yum -y install bind bind-utils
# DNS Sever 를 구현하기 위해 bind 를 설치한다.

(L-1)$ systemctl start named.service
(L-1)$ systemctl status named.service
(L-1)$ netstat -anlp | grep :53
# DNS Server 의 설치 및 포트의 동작여부를 확인할 수 있다.

(L-1)$ firewall-cmd --permanent --zone=public --add-port=53/tcp
(L-1)$ firewall-cmd --permanent --zone=public --add-port=53/udp
(L-1)$ firewall-cmd --permanent --zone=public --add-service=dns
(L-1)$ firewall-cmd --reload
# dns 서버를 사용하기 위해 방화벽을 설정합니다.

(L-1)$ vi /etc/named.conf
listen-on port 53 { 127.0.0.1; }; → listen-on port 53 { 127.0.0.1; 192.168.10.0; };
# 53 번 포트로 namerserver 가 listen 할 수 있도록 설정을 추가한다.


allow-query {127.0.0.1;}          →      allow-query {192.168.10.0}
# 허용할 IP 대역을 설정한다.

allow-transfer { localhost; 192.168.10.20; }
# slave DNS 를 설정한다. (slave DNS 를 사용시에만)
# /etc/named.conf 에서 Bind 에 대한 전체적인 환경설정을 구성할 수 있다.

(L-1)$ vi /etc/named.rfc1912.zones
...
zone "L1.com" IN{
    type master;
    file "L1.zone";
    allow-update { none; };
    notify yes;
    also-notify {192.168.10.20;};
};

# forward zone 에 대한 레코드 설정이다. 해당 파일은 /var/named/L1.zone 으로 생성한
다.
# notify yes; , also-
notify {192.168.10.20;}; 은 slave DNS 을 구성하기 위한 정의이다.

zone "L2.com" IN{
    type master;
    file "L2.zone";
    allow-update { none; };
    notify yes;
    also-notify {192.168.10.20;};
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	


```
};

# reverse zone 에 대한 레코드 설정이다. 해당 파일은 /var/named/L2.com 으로 생성한다
.
# notify yes; , also-
notify {192.168.10.20;}; 은 slave DNS 을 구성하기 위한 정의이다.
# 기본적인 Zone 파일에 대한 설정이 완료되면, 이 파일들을 라우팅 해줄 파일들을
생성해야 한다.

(L-1)$ vi /var/named/L1.zone
$TTL 3H
@          IN      SOA      L1.com. root.L1.com. (
                                0          ; serial
                                1D         ; refresh
                                1H         ; retry
                                1W         ; expire
                                3H )       ; minimum
;Define nameservers
                IN      NS      ns1.L1.com.
                IN      NS      ns2.L1.com.
;Define 86Net Servers

ns1          IN      A        192.168.10.11
ns2          IN      A        192.168.10.11
ntp1         IN      CNAME    ns1.L1.com.

# serial : 시리얼 번호를 기준으로 Primary/Secondary 간에 최신 업데이트 기준을 참고
한다. 작은쪽에서 큰쪽에게 데이터를 업데이트 받는다. 날짜 기준으로 관리하는 것이 편
하다.
# refresh : Secondary 서버가 Primary 서버로 부터 업데이트 받는 기준을 의미. (1D -
하루)
# Retry : 새로 고침 간격을 의미. (1H - 1시간)
# expire : Primary 에 연결되지 않았을 경우의 지정시간. 이 기간이 넘으면 해당 도메인
에 응답하지 않는다.
# TTL - Time to Live 의 약어로 , 자신의 DNS 정보를 다른 DNS 가 가져간 이후 캐시에
저장하는 기간을 의미합니다.
# 예.(3H - 3시간)
# IN - Internet
# SOA - Start of Authority
# NS - NameServer 예시에서는 Master, Slave NameServer 가 해당
# MX - Mail Exchanger , 메일서버
# A - A Record , 호스트이름과 매핑되는 IP 주소 체계
# CNAME - 호스트 Alias 와 유사한 기능을 한다. 예제에서는 NTP1 서버가 ns1 서버로 CN
AME 되었다.
# CNAME 은 호스팅 서비스를 받거나 , AWS, Azure, GCP 등 클라우드 서비스 할 경우에도
네임서비스 편의성과 동적 IP 들에 # 대한 대응 방안으로 많이 사용되기도 한다.
# 실제로는 A 레코드 관리에 대한 부담을 덜기위해서도 많이 사용된다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
(L-1)$ vi /var/named/L2.zone
$TTL 3H
@      IN      SOA      L2.com. root.L2.com. (
                                0      ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum

;Define nameservers
                IN      NS      ns1.L2.com.
                IN      NS      ns2.L2.com.
;Define 86Net Servers

ns1                IN      A      192.168.10.12
ns2                IN      A      192.168.10.12
ntp1              IN      CNAME   ns1.L2.com.
# L2.zone 파일 또한 설정을 진행


(L-1)$ vi /etc/named.conf
logging {
    channel default_debug {
        file "data/named.run";
        print-time yes;
        print-severity yes;
        print-category yes;
        severity debug;
    };
    category queries { "default_debug"; };
    category resolver { "default_debug"; };
};
# 위와 같이 로그를 수정하여, TimeStamp를 구성하고 debug 레벨로 로그를 확인한다.

(L-1)$ chgrp named -R /var/named/
# var/named 파일들에 대해 named가 소유권을 가질 수 있도록 변경한다.

(L-1)$ named-checkconf /etc/named.conf
# named.conf 파일에 오류를 체크한다.

(L-1)$ named-checkconf /etc/named.rfc1912.zones
# named.rfc1912.zones 파일에 오류를 체크한다.

(L-1)$ named-checkzone L1.com /var/named/L1.zone
zone L1.com/IN: loaded serial 0
OK
# L1.zone 파일에 오류를 체크한다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
(L-1)$ named-checkzone L2.com /var/named/L2.zone
zone L2.com/IN: loaded serial 0
OK
# L2.zone 파일에 오류를 체크한다.

(L-1)$ service named restart
# 최종적으로 설정이 완료되면 DNS Server 를 재시작한다.
# 체크시 에러가 발생하지 않아도, 서비스 재구동에서 에러가 발생하는 경우가 존재한다.
# 에러 발생시 대부분 문법에러인 경우이므로 파일 모두를 재점검해야 한다.

(L-2)$ vi /etc/sysconfi.network-scripts/ifcfg-ens32
#
```

Windows DNS Server


현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window15/>

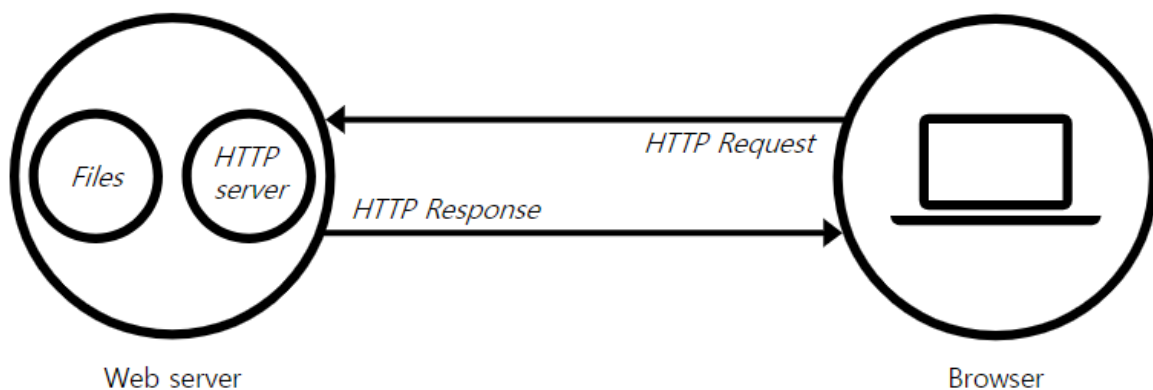
Web Server

- Web Server는 하드웨어, 소프트웨어 혹은 두 개 모두를 의미할 수 있다.
- 하드웨어의 측면에서 Web Server는 website의컴포넌트 파일들을 저장하는 컴퓨터를 의미하며, HTML, images, CSS stylesheets, JavaScript files를 두루 다루며, 이를 엔드 유저에게 전달하는

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

역할을 수행한다.


- 소프트웨어의 측면에서의 Web Server는 기본적으로 웹 사용자가 어떻게 호스트 파일들에 접근하는 지를 관리하며, HTTP 서버는 URL(Web addresses)와 HTTP(브라우저)가 웹 페이지를 보여주기 위해 사용하는 프로토콜을 의미한다.
- 가장 기본적인 단계에서, 브라우저가 웹 서버에서 불러진 파일을 필요로 할 때, 브라우저는 HTTP를 통해 파일을 요청한다. 요청이 올바른 웹 서버(하드웨어)에 도달하였을 때, HTTP 서버 (software)는 하단의 그림과 같이 요청된 문서를 HTTP를 이용하여 전송한다.



- Web Site를 공개하기 위해서는 기본적으로 정적 혹은 동적 웹 서버가 필요로 하며, 정적 웹 서버, 스택은 HTTP 서버(하드웨어 or 소프트웨어)로 구성되어 있다.
- 여기서 정적서버란 웹 페이지를 수정시에 차후 커밋이나 푸쉬를 통해 배포되는 것을 의미하며, 동적서버란 즉시 수정 및 배포되는 서버를 의미한다.

Linux Web Server

```
(L-1)$ rpm -qa | httpd
# 설치되어 있는 Apache Web Server 를 확인한다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

Apache2 Web Server 는 redhat 계열에서는 httpd, Debian 계열에서는 apache2 라는 패키지명을 사용한다.

```
(L-1)$ firewall-cmd --permanent --add-service=http
(L-1)$ firewall-cmd --permanent --add-service=https
(L-1)$ firewall-cmd --permanent --add-port=80/tcp
(L-1)$ firewall-cmd --permanent --add-port=443/tcp
(L-1)$ firewall-cmd --reload
```

Web Server 를 사용하기 위해 방화벽을 설정한다.

```
(L-1)$ systemctl start httpd
(L-1)$ systemctl status httpd
(L-1)$ systemctl netstat -anlp | grep :80
```

Web Server 을 시작하고 포트와 상태를 확인한다.
설정이 완료되면 http://[L1의 IP]의 접속하여 서버를 확인할 수 있다.

```
(L-1)$ vi /etc/httpd/conf/httpd.conf
```

대부분의 Apache 에 대한 설정은 /etc/httpd 에서 가능하며, 여기서는 접속가능 디렉토리를 지정해보겠다.

```
<Directory />
    AllowOverride none
    Require all denied
```

```
</Directory>
```

conf 파일을 살펴보면 위와 같이 “/” 최상단의 디렉토리는 막혀 있는 것을 확인할 수 있다.
즉 기본적으로 생성되는 모든 디렉터리가 막혀있으며, 이는 설정을 통해 오픈이 가능하다.

```
DocumentRoot "/var/www/html"
```

Root 디렉터리가 /var/www/html 로 된 것을 확인할 수 있다.


```
<Directory "/var/www/html">
    ...
    Require all granted
```

하단에는 이와 같이 “/var/www/html”>을 보면 오픈되어 있는 것을 확인할 수 있다.

```
(L-1)$ mkdir /var/www/html/MyWeb
(L-1)$ echo Hello > /var/www/MyWeb/html/index.html
(L-1)$ systemctl restart httpd
```

MyWeb 디렉터리를 생성하고 그 안에 접속시 Hello 가 출력될 수 있도록 index.html 을 생성한다.
서버를 재 시작 후, http://192.168.10.11(IP)/MyWeb/에 접속해보면 권한이 없어 접근이 불가능한 것을 확인할 수 있다.

```
(L-1)$ vi /etc/httpd/conf/httpd.conf
...
<Directory /var/www/html/MyWeb/>
```


	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```

        AllowOverride none
        Require all granted
</Directory>
...
# 이제 다시 httpd.conf 로 들어가 위와 같이 설정을 추가한다.

(L-1)$ systemctl restart httpd
# 서버를 재 시작후 접속하면 창이 뜨는 것을 확인 할 수 있다.

(L-1)$ rpm -qa openssl
# 이번에는 인증된 웹 페이지 Https 를 사용하기 위해 openssl 를 통해 인증서를
생성해보도록 하겠다.
# 기본적으로 CentOS7 의 경우는 openssl 이 설치되어 있다.

(L-1)$ openssl genrsa -des3 -out /etc/httpd/conf/server.key 2048
# 암호를 입력 후에, 개인키를 생성한다.


(L-1)$ openssl req -new -key /etc/httpd/conf/server.key -out /etc/httpd/conf/server.csr
Country Name (2 letter code) [XX]:KR
State or Province Name (full name) []:Seoul
Locality Name (eg, city) [Default City]:Nob
Organization Name (eg, company) [Default Company Ltd]:Nob
Organizational Unit Name (eg, section) []:Nob
Common Name (eg, your name or your server's hostname) []:Nob.com
Email Address []:myemail.mail.com
# 공개 키를 생성하기 위해 각 값들을 입력한다.

(L-1)$ cp /etc/httpd/conf/server.key /etc/httpd/conf/server.key.orgin
(L-1)$ openssl rsa -in /etc/httpd/conf/server.key.orgin -out /etc/httpd/conf/server.key
# 인증서를 보다 편리하게 사용하기 위해 암호를 제거한다.

(L-1)$ openssl x509 -req -days 3650 -in s/etc/httpd/conf/erver.csr -signkey
/etc/httpd/conf/server.key -out /etc/httpd/conf/server.crt
out server.crt
Signature ok
subject=/C=KR/ST=Seoul/L=Nob/O=Nob/OU=Nob/CN=Nob.com/emailAddress=myemail.m
ail.com
Getting Private key
# 개인키와 공개키를 통해 인증서를 생성한다.

(L-1)$ vi /etc/httpd/conf/httpd.conf
NameVirtualHost *:443
<VirtualHost *:443>
SSLEngine on
SSLCertificateFile /etc/httpd/conf/server.crt
SSLCertificateKeyFile /etc/httpd/conf/server.key
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown

```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
CustomLog logs/ssl_request_log "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x
W"%rW" %b"
DocumentRoot /var/www/MyWeb/
</VirtualHost>
# 설정을 맞게 작성 후, 하단에 추가한다.

(L-1)$ yum -y install mod_ssl
# SSL 모듈을 설치한다.

(L-1)$ systemctl restart httpd
# Web Server 를 재시작 후 https://192.168.10.11(IP)/WebServer/에 접속하여 확인한다.
```

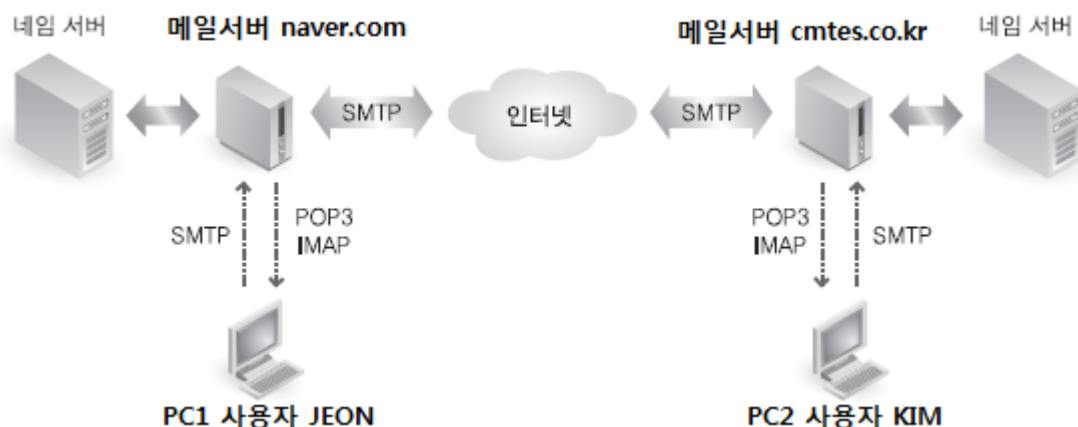
Windows Web Server

현재 정리중입니다!


Windows WEB (IIS) Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window15/>

Mail Server



- Mail Server란 이메일을 송수신할 수 있는 하나의 서버를 뜻하며, 각 클라이언트는 Mail Server

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

를 통해 클라이언트간의 Mail 통신이 가능하다.

- Mail Server는 하단의 프로토콜을 사용한다.
 - SMTP (Simple Mail Transfer Protocol)
 - 클라이언트가 메일을 보내거나, 메일 서버 간에 메일을 주고받을 때 사용하는 프로토콜
 - POP3 (Post Office Protocol)
 - 메일 서버에 도착되어 있는 메일을 클라이언트로 가져올 때 사용되는 프로토콜
 - IMAP (Internet Mail Access Protocol)
 - POP3와 같으나, 동기화 방식으로 부하가 큰 특징을 가지는 프로토콜

Linux Mail Server


```
(L-1)$ yum -y update
# 패키지를 업데이트 한다.

(L-1)$ yum -y install postfix
# postfix 를 사용하기 위해 패키지를 인스톨한다.
# postfix 란 메일서버를 구축하는 오픈소스이다.

(L-1)$ mkdir -p /etc/postfix/ssl
(L-1)$ cd /etc/postfix/ssl
# ssl 인증서를 생성하기 위해 폴더를 생성한다.
# 굳이 받을 필요는 없음

(L-1)$ openssl req -x509 -nodes -newkey rsa:2048 -keyout server.key -out server.crt -nodes -day 365
# ssl 인증서를 생성한다.


(L-1)$
myhostname = mail.L1.com
mydomain = L1.com
myorigin = $mydomain
home_mailbox = mail/
mynetworks = 127.0.0.0/8
inet_interfaces = all
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
inet_protocols = all
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_local_domain =
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_sasl_auth_enable = yes
smtpd_recipient_restrictions =
permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination
smtp_tls_security_level = may
smtpd_tls_security_level = may
smtp_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 1
smtpd_tls_key_file = /etc/postfix/ssl/server.key
smtpd_tls_cert_file = /etc/postfix/ssl/server.crt
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom# 설정파일을 설정 값에 맞게 수정한다.
# 여기서는 myhostname 과 mydomain 만 L1 으로 수정하겠다.
```

```
(L-1)$ vi /etc/postfix/master.cf
```

```
smtp      inet  n       -       n       -       -       smtpd
#smtp     inet  n       -       n       -       1       postscreen
#smtpd    pass  -       -       n       -       -       smtpd
#dnsblog  unix  -       -       n       -       0       dnsblog
#tlsproxy unix  -       -       n       -       0       tlsproxy
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
  -o smtpd_recipient_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
smtps     inet  n       -       n       -       -       smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
  -o smtpd_recipient_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
# 상단과 같이 주석을 해제한다.
# 여기서 SMTP 프로토콜의 설정이 가능하다.

(L-1)$ yum -y install dovecot
# dovecot 패키지를 설치하여 IMAP/POP3 프로토콜을 이용해 이메일을 수신할 수 있도록 한다.

(L-1)$ vi /etc/dovecot/conf.d/10-master.conf
...
# Postfix smtp-auth
unix_listener /var/spool/postfix/private/auth {
    mode = 0660
    user = postfix
    group = postfix
# Postfix smtp-auth를 찾아 상단과 같이 하단에 값을 추가한다.
# 만약 다른 User, Group을 사용할 경우 값을 변경한다.


(L-1)$ vi /etc/dovecot/conf.d/10-auth-conf
auth_mechanisms = plain → auth_mechanisms = plain login
# auth_username_format = %Lu → auth_username_format = %Lu
# 이 설정은 test@example.com 이 아닌 test로 입력하면 로그인 가능하도록 하는 설정이다.
# 편의를 위해 하는 것이며, 강제되는 것은 아니다.

(L-1)$ vi /etc/dovecot/conf.d/10-mail.conf
mail_location = maildir:~/mail
# 10-mail.conf의 최상단에 위에서 설정한 home_mailbox와 동일한 값을 추가한다.
# 만약 값이 틀릴 경우 mail수신이 안될 수 있다.

(L-1)$ vi /etc/dovecot/conf.d/20-pop3.conf
# pop3_uidl_format = $08xu%08Xv → pop3_uidl_format = $08xu%08Xv
# 주석을 제거한다.

(L-1)$ for i in postfix dovecot; do systemctl enable --now $i; done
# 각 서비스를 시작한다.

(L-1)$ firewall-cmd --permanent --add-service=smtp
(L-1)$ firewall-cmd --permanent --add-port=587/tcp
(L-1)$ firewall-cmd --permanent --add-port=465/tcp
(L-1)$ firewall-cmd --permanent --add-port=110/tcp
(L-1)$ firewall-cmd --permanent --add-service=pop3s
(L-1)$ firewall-cmd --permanent --add-port=143/tcp
(L-1)$ firewall-cmd --permanent --add-service=imaps
(L-1)$ firewall-cmd --permanent --add-service=http
(L-1)$ firewall-cmd --reload
# 방화벽에 서비스와 포트를 등록한다.
```

	Report	Version	Last Modified	Linux, Windows Server 비교
	Final Report	3.3	2020.11.12	

```
(L-1)$ adduser test
(L-1)$ passwd test
# 확인을 위해 User 와 Pw 을 생성

(L-1)$ mail -s "mail test" test@L1.com
Hi
# ctrl + D 입력
EOT

(L-1)$ su - test
(L-1)test$ cd ~/mail/new/
(L-1)test$ cat ...
# 보낸 메시지를 확인 할 수 있다.
# 이와 같이 메일 서버를 응용하면 WEB 에서 메일을 주고 받을 수 있도록 할 수 있으며,
DHCP, DNS 와 함께 혼용하면 사내 메일서버를 구축할 수도 있다.
```

Windows Mail Server

현재 정리중입니다!

Windows Server에 대한 설명은 하단의 제 블로그에 정리되어있습니다!

<https://mung0001.github.io/docs/system/window/window15-2/>