CentOS 7's A to Z

저자 박준현

경북산업직업전문학교

머리말

이 책은 경북산업직업전문학교의 IT강의를 위해 제작되었습니다.

VMWare Workstation을 사용하여 테스트환경을 구축하였기 때문에 이에 맞춰 설명이 진행됩니다.
실제 서버에서의 작업환경과 다소 다를 수 있다는 점을 감안하셔야 합니다.

목차

1. Install / Initial Config

1.1. Install CentOS

1.1.1 Download CentOS

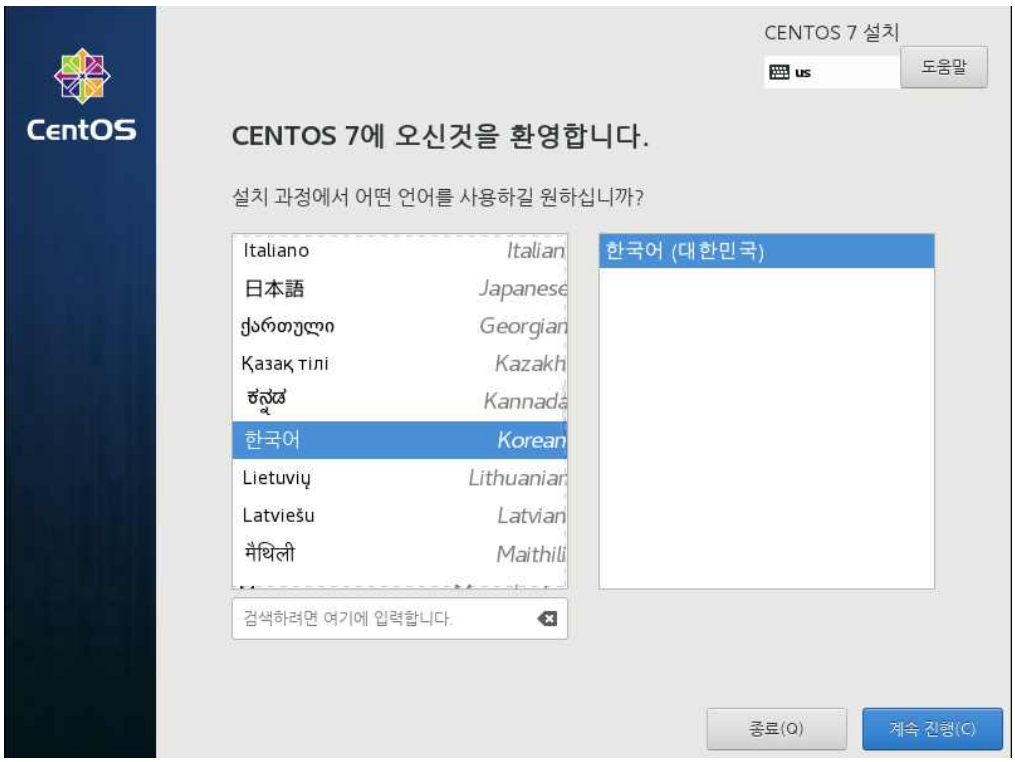
CentOS 7은 2014년 7월7일에 출시 되었으며 2024년 6월 말까지 지원될 예정입니다.
다음 사이트에서 CentOS의 이미지를 다운로드 받을 수 있습니다.

http://isoredirect.centos.org/centos/7.6.1810/isos/x86_64/

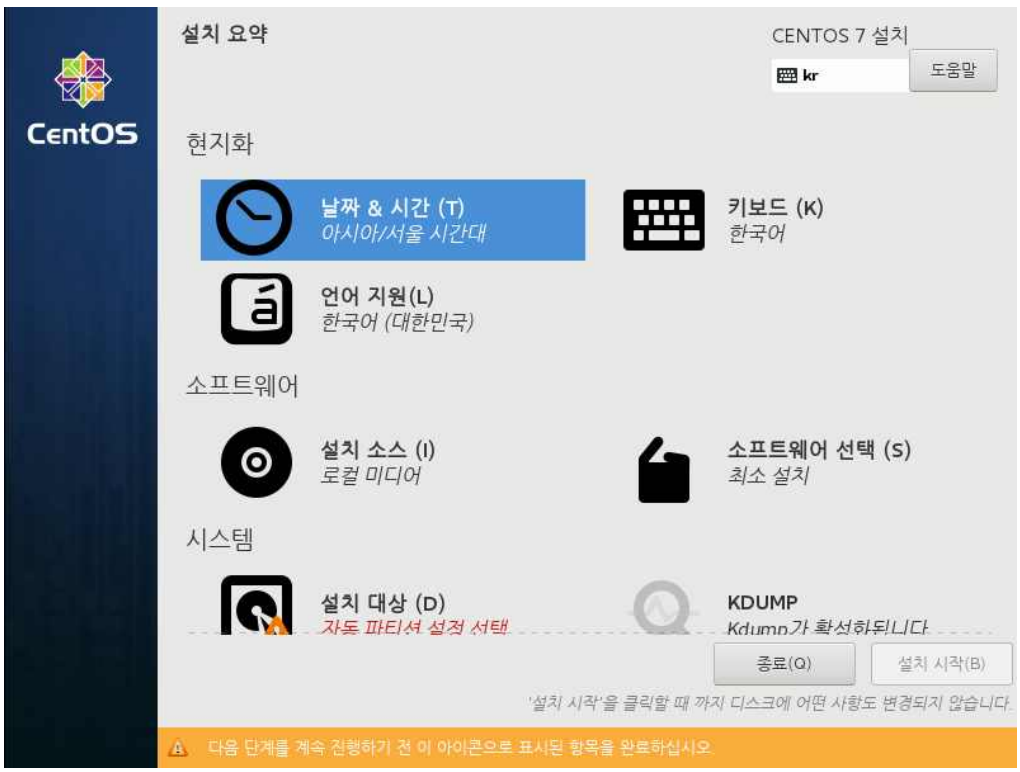
1.1.2 Install CentOS7

[1] CentOS7 설치이미지를 넣고 가상머신을 시작하시면 다음화면을 확인 하실 수 있습니다. Install CentOS 7을 선택합니다.

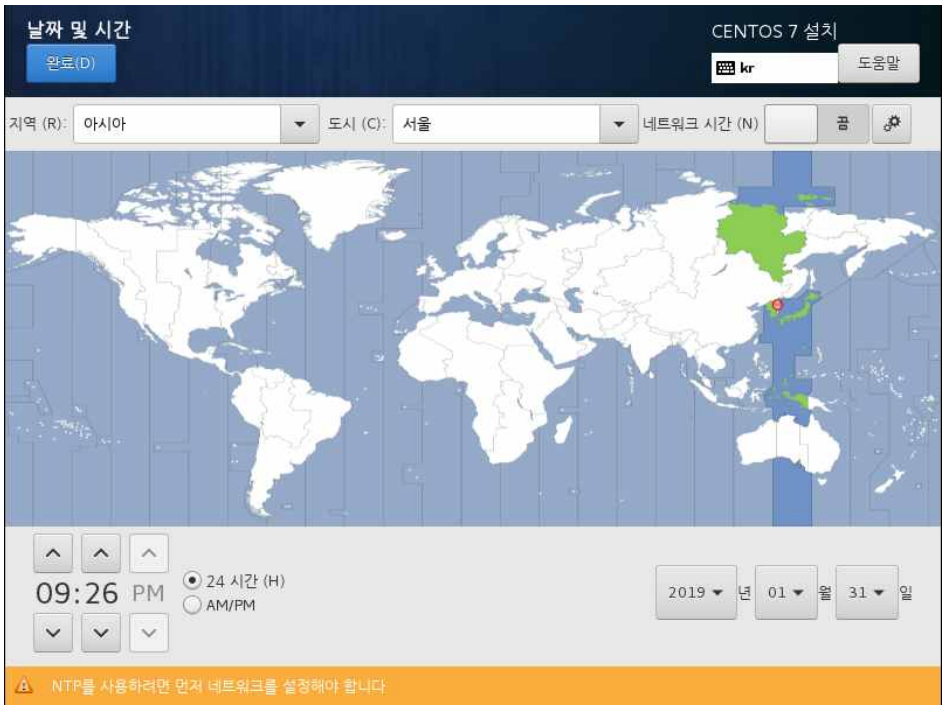
[2] 설치중 사용할 언어를 선택합니다.



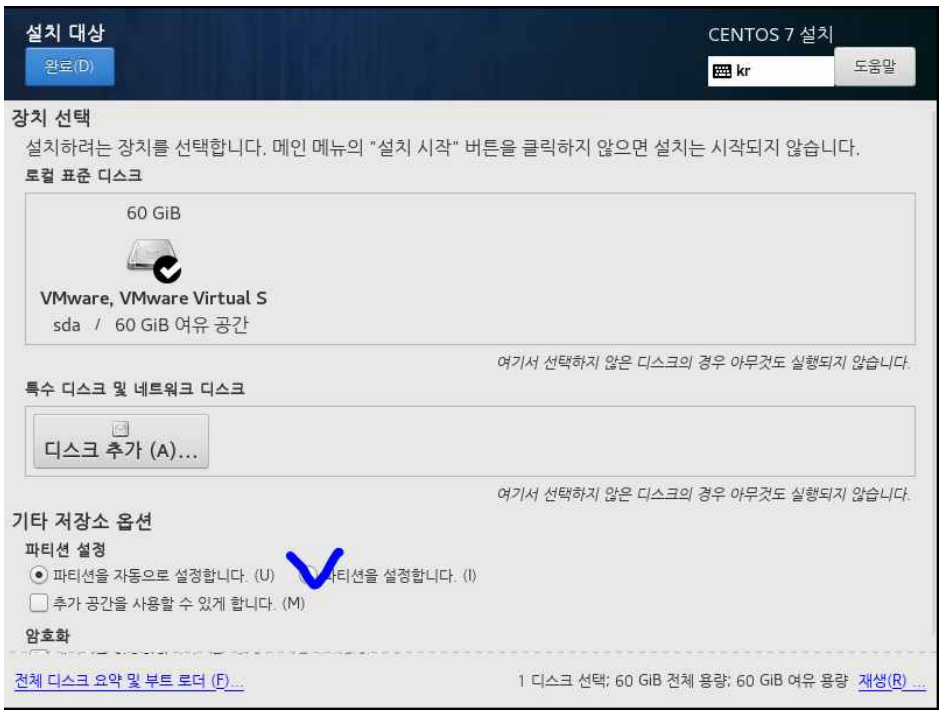
[3] 다음 화면이 기본 화면입니다. 먼저 [DATE&TIME] 아이콘을 클릭하고 시간대를 설정합니다.



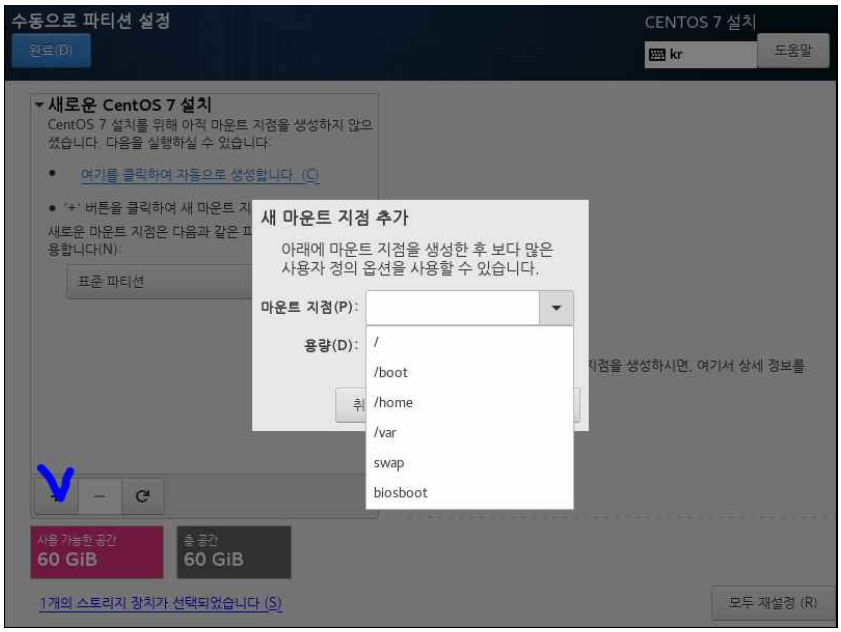
[4] 지도에서 시간대를 설정할 지점을 클릭하고 왼쪽 위에 있는 [완료] 버튼을 누르십시오.



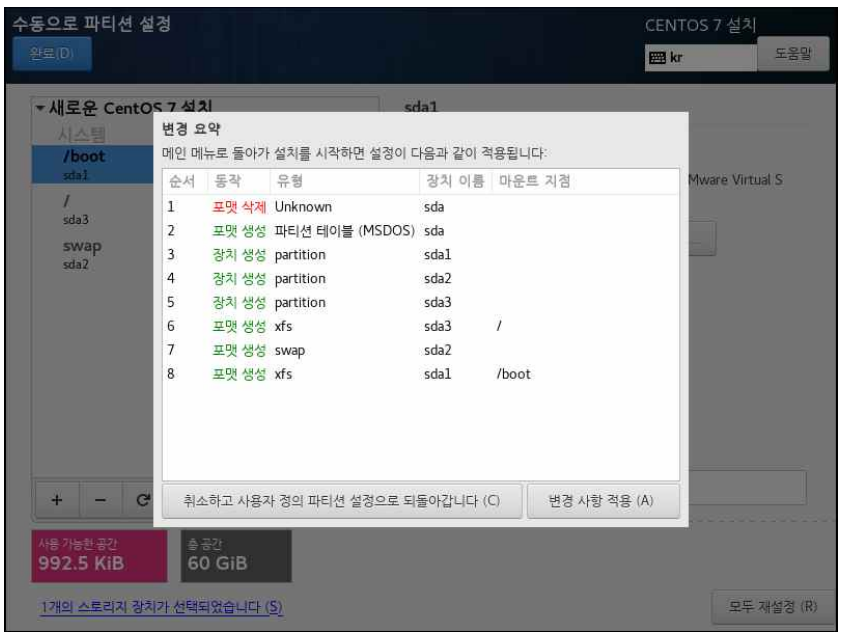
[5] 완료 버튼을 누르고 돌아와 시스템 카테고리의 설치대상(D)를 클릭합니다. 설치할 디스크를 선택하고 파티션을 설정할 수 있습니다.



[6] [+]기호를 클릭해 새 마운트 지점을 추가 할 수 있습니다.
마운트 지점에 나오는 리스트에 따라 용량을 지정해 주시면 됩니다.
용량을 지정하지 않을 경우 남아 있는 모든 용량이 지정이 됩니다.



/boot : 부트로더가 저장될 공간입니다. 일반적으로 1GB 이상이 필요합니다.
/swap : windows의 가상메모리와 같은 역할을 합니다. 메모리가 부족할 경우 하드디스크 공간의 일부를 메모리처럼 사용할 수 있도록 합니다. 일반적으로 실제 메모리의 1.5~2배정도를 잡아주게 됩니다.
/ : 전체공간에 대한 루트디렉터리입니다. 나머지를 모두 잡아주시면 됩니다.

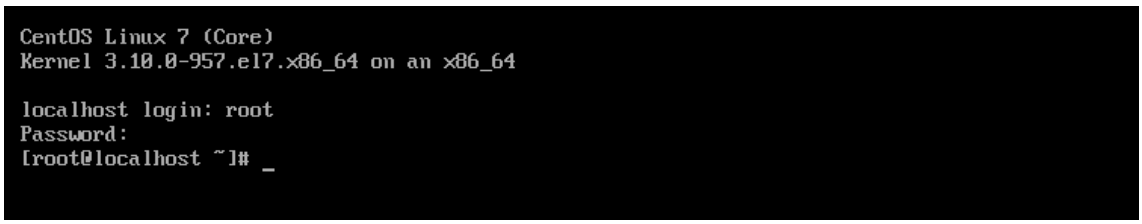


설정이 완료되면 변경 사항 적용(A)를 누르시면 됩니다.

[8] 설정이 완료되고 난 후 설치 시작을 누르시면 설치가 시작됩니다.

설치가 진행되는 동안 ROOT의 암호를 지정해 주시면 됩니다.

[9] 부팅 된 이후 ID/Password를 입력해 로그인 하시면 됩니다.



1.2. Initial Configure

1.2.1 유저생성

[1] “test1”이라는 유저를 생성/비밀번호를 설정

```
[root@localhost ~]#
useradd test1
[root@localhost ~]#
passwd test1
Changing password for user cent.
New UNIX password:              # 패스워드 입력
Retype new UNIX password:       # 패스워드 확인
passwd: all authentication tokens updated successfully.
```

[2] 유저 변경

```
[root@localhost ~]# su - test1          # test1로 사용자전환
마지막 로그인: 목  2월 21 01:33:40 KST 2019 일시 tty1
[test1@localhost ~]$ su -               # root로 사용자전환
암호:                                   # root 암호입력
마지막 로그인: 수  2월 20 16:35:46 KST 2019 10.0.0.1에서 시작 일시 pts/0
[root@localhost ~]#
```

[3] 관리자로 설정(test1)

```
[root@localhost ~]# usermod -G wheel test1
[root@localhost ~]# vi /etc/pam.d/su

#%PAM-1.0
auth            sufficient      pam_rootok.so
# Uncomment the following line to implicitly trust users in the "wheel" group.
#auth           sufficient      pam_wheel.so trust use_uid
# Uncomment the following line to require a user to be in the "wheel" group.
# Uncomment the foolowing line
auth            required        pam_wheel.so use_uid
auth            substack        system-auth
auth            include         postlogin
account         sufficient      pam_succeed_if.so uid = 0 use_uid quiet
account         include         system-auth
password        include         system-auth
session         include         system-auth
session         include         postlogin
session         optional        pam_xauth.so
```

1.2.2 FireWalld

[1] 동작확인

```
[root@localhost ~]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor
preset: enabled)
   Active: active (running) since 수 2019-02-20 22:55:35 KST; 6h left
     Docs: man:firewalld(1)
 Main PID: 8314 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─8314 /usr/bin/python -Es /usr/sbin/firewalld --nofork --n...

 2월 20 22:55:34 localhost.localdomain systemd[1]: Starting firewalld...
 2월 20 22:55:35 localhost.localdomain systemd[1]: Started firewalld ...
Hint: Some lines were ellipsized, use -l to show in full.
```

[2] 종료/시작 시 자동실행방지

```
[root@localhost ~]# systemctl stop firewalld
[root@localhost ~]# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
```

1.2.3 SELinux

[1] 상태 확인

```
[root@localhost ~]# getenforce
Enforcing                    # SELinux가 작동하고 있음
```

[2] 종료/시작 시 자동실행방지

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled             # SELinux상태 변경
# SELINUXTYPE= can take one of three values:
#     targeted - Targeted processes are protected,
#       minimum - Modification of targeted policy. Only selected processes are
protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

1.2.4 Networking

[1] 스크립트를 수정해 설정하는 방법

```
[root@localhost ~]# vi /etc/sysconfig/network-scripts/ifcfg-ens33

TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=cf3b7bfe-64b9-4171-9d61-ee68a8665627
DEVICE=ens33
ONBOOT=yes
IPADDR=10.0.0.30
PREFIX=24
GATEWAY=10.0.0.2
DNS1=10.0.0.2
```

[2] Network Manager를 이용해 설정하는 방법

```
[root@localhost ~]# nmcli d
DEVICE  TYPE      STATE         CONNECTION
ens33   ethernet  연결됨        ens33
lo      loopback  관리되지 않음  --
[root@localhost ~]# nmcli c modify ens33 ipv4.addresses 10.0.0.30/24    # IP설정
[root@localhost ~]# nmcli c modify ens33 ipv4.gateway 10.0.0.2          # GW설정
[root@localhost ~]# nmcli c modify ens33 ipv4.dns 10.0.0.2              # DNS설정
[root@localhost ~]# nmcli c down ens33; nmcli c up ens33                # 재시작
연결     'ens33'이(가)     성공적으로     비활성화되었습니다(D-Bus     활성     경로:
/org/freedesktop/NetworkManager/ActiveConnection/2).
연결이       성공적으로       활성화되었습니다       (D-Bus       활성       경로:
/org/freedesktop/NetworkManager/ActiveConnection/3)
```

이 경우에는 재부팅 시 설정이 저장되지 않으므로 주의 하셔야 합니다.

1.2.5 Update System

[1] CentOS를 설치하고 나면, 가능한 가장 먼저 Update를 실행합니다.

```
[root@localhost ~]# yum -y update
Loaded plugins: fastestmirror
Determining fastest mirrors
 * base: mirror.navercorp.com
 * extras: mirror.navercorp.com
 * updates: mirror.navercorp.com
base                                                    | 3.6 kB     00:00
extras                                                  | 3.4 kB     00:00
updates                                                 | 3.4 kB     00:00
(1/4): base/7/x86_64/group_gz                             | 166 kB   00:00
(2/4): extras/7/x86_64/primary_db                         | 179 kB   00:00
...
...
  selinux-policy.noarch 0:3.13.1-229.el7_6.9
  selinux-policy-targeted.noarch 0:3.13.1-229.el7_6.9
  systemd.x86_64 0:219-62.el7_6.3
  systemd-libs.x86_64 0:219-62.el7_6.3
  systemd-sysv.x86_64 0:219-62.el7_6.3
  tzdata.noarch 0:2018i-1.el7

Complete!
[root@localhost ~]#
```

2. NTP Server

2.1. NTP Server

2.1.1 NTPd

[1] NTPd설치

```
[root@localhost ~]# yum -y install ntp
[root@localhost ~]# vi /etc/ntp.conf

# line 17 : 요청을 수신할 수 있는 네트워크 범위를 추가합니다.
restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

# line 21 : 설정되어 있는 ntp코드를 주석처리하고 아시아 주소를 입력하고 저장합니다.
#server 0.centos.pool.ntp.org iburst
#server 1.centos.pool.ntp.org iburst
#server 2.centos.pool.ntp.org iburst
#server 3.centos.pool.ntp.org iburst
server 0.asia.pool.ntp.org
server 1.asia.pool.ntp.org
server 2.asia.pool.ntp.org
server 3.asia.pool.ntp.org
```

[2] 방화벽 설정

```
[root@localhost ~]# firewall-cmd --add-service=ntp --permanent
success
[root@localhost ~]# firewall-cmd --reload
success
```

[3] 서비스 시작

```
[root@localhost ~]# systemctl start ntpd
[root@localhost ~]# systemctl enable ntpd
Created symlink from /etc/systemd/system/multi-user.target.wants/ntpd.service to /usr/lib/systemd/system/ntpd.service.
```

[4] 서비스 확인

```
[root@localhost ~]# ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*ntp1.jst.mfeed. 133.243.236.17   2 u   63   64   17   59.661   -0.549   1.829
 ntp.paina.net   203.178.138.38   2 u   59   64   17   46.717   -1.597   1.541
 i7.ipv9.xyz     202.47.249.20    2 u   58   64   17  122.625   -1.443   3.160
 ntp.xtom.com.hk 179.43.76.147    2 u   61   64   17  164.466   56.710   2.515
```

※ 서버 주소 앞에 붙는 기호의 의미는 다음과 같습니다.

* : 서버와 동기화가 진행중

+ : 서버와 동기화가 가능

표시없음 : 접속이 불가능함

3. FTP Server

3.1. FTP Server

3.1.1 Proftpd

FTP(File Transfer Protocol) Server란 클라이언트 프로그램을 이용해서 서버와 파일을 전송할 수 있도록 만든 프로그램입니다. 별도의 클라이언트가 필요하기 때문에 NFS나 다른 파일/디렉터리 공유 시스템과는 약간 다릅니다.

많은 FTP Server프로그램이 있지만 여기서는 Proftpd를 이용해 FTP Server를 구축해 보고자 합니다.

[1] Proftpd 설치(source comfile)

```
# 소스를 다운 받을 디렉터리를 생성합니다.
[root@localhost ~]# mkdir src
[root@localhost ~]# cd src

# 소스컴파일 설치를 위해선 gcc와 gcc-c++이 반드시 설치되어 있어야 합니다.
[root@localhost ~]# yum -y install gcc gcc-c++

# 소스파일을 다운로드 받습니다.
[root@localhost ~]# wget ftp://ftp.proftpd.org/distrib/source/proftpd-1.3.5e.tar.gz

# 압축을 해제하고 디렉터리를 이동 합니다.
[root@localhost src]# tar zxvf proftpd-1.3.5e.tar.gz
[root@localhost src]# cd proftpd-1.3.5e

# 설치될 파일을 구성에 맞게 설정합니다.
[root@localhost proftpd-1.3.5e]# ./configure ₩
> --prefix=/server/proftpd ₩
> --sysconfdir=/server/conf/proftpd

# 설치할 파일을 생성합니다.
[root@localhost proftpd-1.3.5e]# make

# 파일을 지정된 위치로 복사합니다.
[root@localhost proftpd-1.3.5e]# make install

# 설치 확인
[root@localhost proftpd-1.3.5e]# cd /server/
[root@localhost server]# ls
conf  proftpd
```

[2] Proftpd 설정

```
[root@localhost server]# vi /server/conf/proftpd/proftpd.conf
# line 6 : 서버의 이름을 변경합니다.
ServerName                      "www.ftp.srv “

# line 31 : 기본 Group중에는 nogroup이라는 그룹이 없습니다. 변경해줍니다.
Group                           nobody

# line 59 : 접속시 표시할 글씨를 저장해둔 파일의 경로를 지정합니다.
DisplayLogin                    /server/conf/proftpd/welcome.msg

# 추가
DefaultAddress                  10.0.0.30
```

[3] 방화벽 설정

```
[root@localhost sbin]# firewall-cmd --add-service=ftp --permanent
success
[root@localhost sbin]# firewall-cmd --reload
success
```

[4] systemd에 등록

```
# 운영에 편의를 위해 systemd에 등록 합니다. 다음과 같이 파일을 새로 생성합니다.
[root@localhost sbin]# vi /usr/lib/systemd/system/proftpd.service

# 아래의 내용을 모두 추가합니다.
[Unit]
Description=Proftpd FTP Server
After=network.target

[Service]
Type= forking
ExecStart = /server/proftpd/sbin/proftpd
ExecStop = /server/proftpd/sbin/ftpshut now

[Install]
WantedBy=multi-user.target

# systemd를 재시작합니다.
[root@localhost sbin]# systemctl daemon-reload
```

[5] FTP서버 실행

```
[root@localhost sbin]# systemctl start proftpd
[root@localhost sbin]# systemctl status proftpd
● proftpd.service - Proftpd FTP Server
   Loaded: loaded (/usr/lib/systemd/system/proftpd.service; disabled; vendor
preset: disabled)
   Active: active (running) since 목 2019-02-21 12:10:20 KST; 20s ago
  Process: 19357 ExecStart=/server/proftpd/sbin/proftpd (code=exited,
status=0/SUCCESS)
 Main PID: 19358 (proftpd)
   CGroup: /system.slice/proftpd.service
           └─19358 proftpd: (accepting connections)

 2월 21 12:10:20 localhost.localdomain systemd[1]: Starting Proftpd FTP Serv...
 2월 21 12:10:20 localhost.localdomain systemd[1]: Started Proftpd FTP Server.
 2월 21 12:10:20 localhost.localdomain proftpd[19358]: 10.0.0.30 - ProFTPD 1…UP
 2월 21 12:10:20 localhost.localdomain proftpd[19358]: 10.0.0.30 - /etc/shut...
Hint: Some lines were ellipsized, use -l to show in full.
```

[6] FTP 서버에 접속

FTP서버가 정상적으로 구동이 되는 것을 확인 했다면 서버에 접속해 봅시다. FTP서버는 기본적으로 root의 접속이 불가하도록 되어 있기 때문에 새로이 유저를 생성해 접속하여야 합니다. 여기서는 ftptest라는 유저를 생성해 Windows의 cmd에서 접속을 시도하였습니다.

```
C:₩Users₩User>ftp
ftp> o 10.0.0.30
10.0.0.30에 연결되었습니다.
220 ProFTPD 1.3.5e Server (ProFTPD Default Installation) [10.0.0.30]
500 OPTS UTF8 not understood
사용자(10.0.0.30:(none)): ftptest
331 Password required for ftptest
암호:
230 User ftptest logged in
ftp>
```

[6] FTP서버의 재실행

Proftpd는 따로 실행/종료 스크립트를 제공하지 않고, ftpshut은 관리만 종료하는 프로세스이기 때문에, 종료 후 재실행을 위해선 /etc/shutmsg를 삭제해 주어야 합니다.

```
[root@localhost ~]# systemctl stop proftpd
[root@localhost ~]# rm -vf /etc/shutmsg
removed `/etc/shutmsg'
```

4. DNS Server

4.1. DNS Server

4.1.1 BIND

DNS(Domain Name System)란 호스트의 도메인 이름을 호스트의 네트워크 주소로 바꾸거나 그 반대의 변환을 수행할 수 있도록 해주는 서비스입니다.

주로 Dnsmasq나 BIND가 사용됩니다.

여기서는 BIND를 통해 DNS Server를 구현해 보도록 합니다.

[1] BIND 설치(yum) 및 설정

```
[root@localhost ~]# yum -y install bind bind-util
[root@localhost ~]# vi /etc/named.conf

# line 13, 14 : ipv6를 사용하지 않도록 하고, 모든 쿼리에 대한 응답을 할 수 있도록
  설정합니다.
listen-on port 53 { any; };
listen-on-v6 port 53 { none; };

# line 21 : DNS쿼리를 받을 네트워크를 설정합니다.
allow-query     { localhost; 10.0.0.0/24; }

# line 22 : Secondary DNS를 사용한다면 다음을 추가해 줍니다.
allow-transfer  { localhost; 10.0.0.0/24; };
```

[2] Zone 정보 설정

```
[root@localhost ~]# vi /etc/named.rfc1912.zones

# line 43 : Zone정보를 가진 파일을 등록하기 위해 아래내용을 추가합니다.
zone "rpgmaster.com" IN {
      type master;
      file "rpgmaster.com.zone";          # 정방향 정보를 담고 있는 파일입니다.
      allow-update { none; };
};

zone "0.0.10.in-addr.arpa" IN {
      type master;
      file "rpgmaster.com.rev";           # 역방향 정보를 담고 있는 파일입니다.
      allow-update { none; };
};
```

[3] 정방향(역방향) 조회 영역 파일 만들기

```
[root@rpgmaster ~]# cd /var/named/

# 미리 만들어져 있는 loopback 조회영역을 복사하여 참조해 만듭니다.
[root@rpgmaster named]# cp -v named.loopback rpgmaster.com.zone
`named.loopback' -> `rpgmaster.com.zone'
[root@rpgmaster named]# cp -v named.loopback rpgmaster.com.rev
`named.loopback' -> `rpgmaster.com.rev'

# 정방향 조회 영역 편집
[root@rpgmaster named]# vi rpgmaster.com.zone

$TTL 1D
@       IN SOA  @ rpgmaster.com. (
                                        0       ; serial
                                        1D      ; refresh
                                        1H      ; retry
                                        1W      ; expire
                                        3H )    ; minimum
        NS      rpgmaster.com.
        A       10.0.0.30
www     A       10.0.0.30

# 역방향 조회 영역 편집
[root@rpgmaster named]# vi rpgmaster.com.rev

$TTL 1D
@       IN SOA  rpgmaster.com.          rpgmaster.com.(
                                        0       ; serial
                                        1D      ; refresh
                                        1H      ; retry
                                        1W      ; expire
                                        3H )    ; minimum
        NS      rpgmaster.com.
30      PTR     rpgmaster.com.

# 조회 영역 파일 권한 변경
[root@rpgmaster named]# cd /var/named
[root@rpgmaster named]# chown named.named ./rpgmaster.com.zone
[root@rpgmaster named]# chown named.named ./rpgmaster.com.rev
```

- TTL : 사용자의 DNS레코드에 대한 정보를 서버가 캐시하는 시간입니다. 예를 들어, 사용자가 특정 레코드의 TTL을 한시간으로 설정하는 경우, 서버는 사용자의 승인된 네임서버에서 업데이트된 정보를 검색하기 전에 한시간 동안 로컬로 해당 레코드에 대한 정보를 저장합니다.
- SOA(Start Of Authority) : 권한의 시작. 레코드 설정 정보 SOA레코드의 구성은 다음과 같습니다.

1) 도메인이름 : DNS 관리자 주소
2) 시리얼번호(serial) : 영역 파일을 변경할 때 증가 시킴
3) 갱신간격(refresh) : 2차 네임 서버에 업데이트된 정보를 요청하는 간격
4) 재시도 간격(retry) : 2차 네임 서버가 정보 취득에 실패했을 때 재시도하는 시간 간격
5) 데이터유효시간(expire) : 2차 네임 서버가 영역 정보를 유효로 하는 시간 간격
6) 네거티브캐시시간(minimum) : 이 영역 정보의 검색 실패를 캐시(유지)하는 시간

●) 시간 간격은 모두 초단위입니다. 다시 말해, 10800(3H), 604800(1W), 86400(1D)입니다.

- @ : 도메인 전체(rpgmaster.com)를 대상으로 한다는 것을 명시

[4] BIND 실행

```
[root@rpgmaster named]# systemctl start named
[root@rpgmaster named]# systemctl status named
```

[5] 방화벽 설정

```
[root@rpgmaster named]# firewall-cmd --add-service=dns --permanent
success
[root@rpgmaster named]# firewall-cmd --reload
success
```

[6] DNS Server Test

```
# DNS서버 변경
[root@rpgmaster named]# nmcli c modify ens33 ipv4.dns 10.0.0.30
[root@rpgmaster named]# nmcli c down ens33; nmcli c up ens33
```

```
연결 'ens33'이(가) 성공적으로 비활성화되었습니다(D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/1).
연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/2)

# dig를 이용해 확인
[root@rpgmaster named]# dig rpgmaster.com

; <<>> DiG 9.9.4-RedHat-9.9.4-73.el7_6 <<>> rpgmaster.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60709
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;rpgmaster.com.                 IN      A

;; ANSWER SECTION:
rpgmaster.com.          86400   IN      A       10.0.0.30

;; AUTHORITY SECTION:
rpgmaster.com.          86400   IN      NS      rpgmaster.com.

;; Query time: 0 msec
;; SERVER: 10.0.0.30#53(10.0.0.30)
;; WHEN: 금  2월 22 10:12:49 KST 2019
;; MSG SIZE  rcvd: 72

# nslookup을 이용해 확인
[root@rpgmaster named]# nslookup
> rpgmaster.com
Server:         10.0.0.30
Address:        10.0.0.30#53

Name:   rpgmaster.com
Address: 10.0.0.30
```

4.2. Slave DNS Server

```
# DNS서버 변경
[root@rpgmaster named]# nmcli c modify ens33 ipv4.dns 10.0.0.30
[root@rpgmaster named]# nmcli c down ens33; nmcli c up ens33
연결 'ens33'이(가) 성공적으로 비활성화되었습니다(D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/1).
연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/2)

# dig를 이용해 확인
[root@rpgmaster named]# dig rpgmaster.com

; <<>> DiG 9.9.4-RedHat-9.9.4-73.el7_6 <<>> rpgmaster.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60709
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;rpgmaster.com.                 IN      A

;; ANSWER SECTION:
rpgmaster.com.          86400   IN      A       10.0.0.30

;; AUTHORITY SECTION:
rpgmaster.com.          86400   IN      NS      rpgmaster.com.

;; Query time: 0 msec
;; SERVER: 10.0.0.30#53(10.0.0.30)
;; WHEN: 금  2월 22 10:12:49 KST 2019
;; MSG SIZE  rcvd: 72

# nslookup을 이용해 확인
[root@rpgmaster named]# nslookup
> rpgmaster.com
Server:         10.0.0.30
Address:        10.0.0.30#53
```

```
Name:   rpgmaster.com
Address: 10.0.0.30
```

5 WEB Server

Web서비스는 대표적이면서도 보편적인 서비스일 것입니다. 필드에서 가장 활용도가 높은 서비스이며, 때문에 이를 응용한 서비스 컴포넌트도 굉장히 다양합니다. 때문에 웹서비스에 대해 익혀두는 것은 매우 중요하다고 할 수 있습니다. 이번 장에서 우리는 대표적인 Web데몬인 아파치와 nginex를 다뤄볼 것입니다.

5.1. Apache Server

5.1.1 Install httpd

[1] httpd설치

```
[root@localhost ~]# yum -y install httpd
# 우선 테스트에 방해가 되는 웰컴페이지를 삭제합니다.
[root@localhost ~]# rm  -f /etc/httpd/conf.d/welcome.conf

# 이제 httpd.conf를 서버의 설정에 맞게 수정해 줍니다.
[root@localhost ~]# vi /etc/httpd/conf/httpd.conf
     86 ServerAdmin root@rpgmaster.com
     95 ServerName rpgmaster.com:80
    151       AllowOverride All                    # 새로운 접근방식을 우선적용해 인증
    164       DirectoryIndex index.html index.cgi index.php

# 마지막줄에 다음과 같이 넣어 404페이지에 헤더정보가 나타나지 않도록 해 줍니다.
    354 ServerTokens Prod

# keepalive를 on으로 해줍니다.(연결상태 유지)
    355 KeepAlive On
```

[2] 방화벽 설정

```
[root@localhost ~]# firewall-cmd --add-service=http --permanent
success
[root@localhost ~]# firewall-cmd --reload
success
```

[3] 페이지 테스트

```
간단한 HTML문을 작성해 페이지를 테스트 해 봅시다.
[root@localhost ~]# vi /var/www/html/index.html
<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
```

```
Test Page
</div>
</body>
</html>
```

[4] 페이지확인(rpgmaster.com)

5.1.2. Perl Script 사용하기

Perl Script가 이제는 유행이 많이 지났지만 아직 개편하지 못한 소규모 웹서버에서는 왕왕 사용하고 있습니다. 필드에서는 어떤 상황을 만날지 모르니 기초세팅은 일단 배워 둡시다.

[1] Perl설치

```
[root@localhost ~]# yum -y install perl perl-CGI
```

[2] 기본적으로 CGI는 "/var/www/cgi-bin" 디렉토리에 허용됩니다.

```
[root@rpgmaster ~]# grep -n "^ *ScriptAlias" /etc/httpd/conf/httpd.conf
247:     ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
위와 같이 httpd.conf파일에 내용이 삽입된 것을 확인 할 수 있습니다.

만약 다른 디렉터리에서 CGI를 허용해 주려면 다음과 같이 설정합니다.
[root@rpgmaster ~]# vi /etc/httpd/conf.d/cgi-enabled.conf

# 파일 내용을 아래와 같이 새로 작성합니다.
<Directory "/var/www/html/cgi-enabled">
    Options +ExecCGI
    AddHandler cgi-script .cgi .pl
</Directory>

[root@rpgmaster ~]# systemctl restart httpd
```

[3] SELinux

```
SELinux가 활성화 되어 있는 상태에서 기본 디렉터리가 아닌 다른 곳에 CGI를 활성화 시켰다면 다음과 같이 httpd에서 해당 디렉터리를 사용할 수 있게 해주어야 합니다.

[root@rpgmaster ~]# mkdir /var/www/html/cgi-enabled
[root@rpgmaster ~]# chcon -R -t httpd_sys_script_exec_t /var/www/html/cgi-enabled
[root@rpgamster ~]# semanage fcontext -a -t httpd_sys_script_exec_t ₩
> /var/www/html/cgi-enabled
```

◎ CentOS 7을 최소설치하면 selinux 관련 명령어를 포함한 패키지가 몇 가지 설치됩니다. getenforce, setenforce, getsebool, restorecon, setsebool 등의 명령어가 포함된 패키지는 설치가 되지만 semanage가 포함된 패키지는 최소설치 목록에 없죠.
yum install policycoreutils-python명령어를 이용해 policycoreutils-phtyon을 설치하면 semanage, audit2allow 등의 명령어를 사용할 수 있게 됩니다.

[4] 페이지 테스트

```
간단한 CGI구문을 작성해 페이지를 만들어 봅시다.
[root@rpgmaster ~]# vi /var/www/html/cgi-enabled/index.cgi

#!/usr/bin/perl
print "Content-type: text/html₩n₩n";
print "<html>₩n<body>₩n";
print "<div style=₩"width: 100%; font-size: 40px; font-weight: bold; text-align: center;₩">₩n";
print "CGI Test Page";
print "₩n</div>₩n";
print "</body>₩n</html>₩n";

[root@rpgmaster ~]# chmod 705 /var/www/html/cgi-enabled/index.cgi
```

[5] 페이지 확인(rpgmaster.com/cgi-enabled/index.cgi)

5.1.3. PHP Script 사용하기

PHP는 JSP와 함께 가장 많이 사용되고 있는 스크립트 언어일 것입니다. 종종 사용하게 될 것입니다. 과거 보안취약점 때문에 한풀 주춤했었지만, 이를 해결한 후 다시 빠르게 성장해 나갔습니다. 수많은 페이지들이 PHP를 메인으로 사용하고 있으며, 안정화 되고 있습니다.

[1] PHP 설치하기

PHP역시 마찬가지로 yum을 이용해 간단하게 설치가 가능하지만 CentOS리포지터리에는 PHP5버전이 등록되어 있기 때문에 여기서는 PHP7을 설치해 보도록 합시다.
PHP최신 버전을 제공하는 외부 리포지터리 중 유명한 곳은 webtatic과 remi등이 있습니다. 여기서는 webtatic의 리포지터리를 이용해 봅시다.

```
[root@rpgmaster ~]# rpm -Uvh ₩
> https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
[root@rpgmaster ~]# rpm -Uvh ₩
> https://mirror.webtatic.com/yum/el7/webtatic-release.rpm

[root@rpgmaster ~]# yum -y install php70w
```

다음 명령어로 php7관련 패키지 중 설치할 수 있는 리스트들을 확인할 수 있습니다.
필요한 경우 함께 설치해 주도록 합시다.

```
[root@rpgmaster ~]# yum search php70w
```

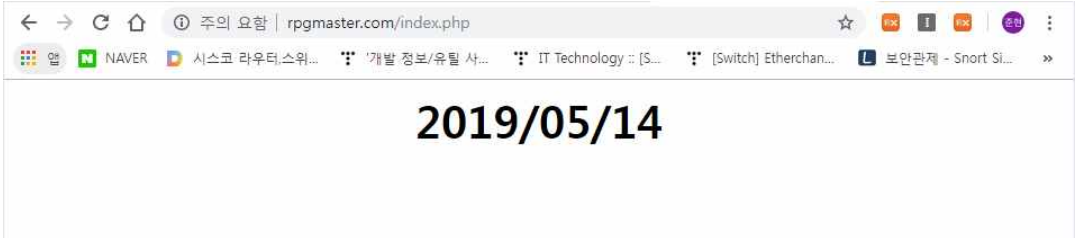
[2] 페이지 테스트

이번에도 간단한 PHP구문을 만들어 확인해 보도록 합시다.

```
[root@rpgmaster ~]# vi /var/www/html/index.php

<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
<?php
    print Date("Y/m/d");
?>
</div>
</body>
</html>
```

[3] 페이지 확인(rpgmaster.com/index.php)



5.1.4. Ruby Script 사용하기

요즘 Open Source Project중에는 Ruby가 많이 등장합니다. 루비를 CGI로 사용할 수 있도록 세팅해 봅시다.

[1] Ruby 설치 및 설정

```
루비역시 yum을 이용해 간단하게 설치가 가능합니다.

[root@rpgmaster ~]# yum -y install ruby

앞서 Perl을 할 때 설명했듯이 기본적으로 CGI는 "/var/www/cgi-bin"디렉터리 아래에 허용됩니다. 그 아래의 모든 파일은 CGI로 처리가 됐었죠?

다른 디렉터리에서 CGI를 사용하기 위해 설정파일을 수정했던 것을 기억해 봅시다.

[root@rpgmaster ~]# vi /etc/httpd/conf.d/cgi-enabled.conf

<Directory "/var/www/html/cgi-enabled">
    Options +ExecCGI
    AddHandler cgi-script .rb
</Directory>

[root@rpgmaster ~]# systemctl restart httpd

SELinux에 대한 설정은 Perl을 사용할 때 했었으니 생략합시다.
```

[2] 페이지 테스트

```
[root@rpgmaster ~]# vi /var/www/html/cgi-enabled/index.rb

#!/usr/bin/ruby

print "Content-type: text/html₩n₩n"
```

```
print "<html>₩n<body>₩n"
print "<div style=₩"width: 100%; font-size: 40px; font-weight: bold; text-align: center;₩">₩n"
print "Ruby Script Test Page"
print "₩n</div>₩n"
print "</body>₩n</html>₩n"

[root@rpgmaster ~]# chmod 705 /var/www/html/cgi-enabled/index.rb
```

[3] 페이지 확인(rpgmaster.com/cgi-enabled/index.rb)

◎ 예제] 파이썬을 설치해 파이썬을 CGI로 구성해 페이지를 확인 할 수 있도록 해 봅시다.(Python, .py)

5.1.5. Userdir 사용하기

Userdir기능을 사용하면 사용자별 웹사이트를 만들 수 있습니다. Userdir기능을 응용하면 다양한 웹사이트 연출이 가능하며, 우리가 흔히 알고 있는 호스팅 서비스가 이 Userdir기능과 VitualHost기능을 응용한 것입니다.

[1] httpd설정하기(userdir.conf)

```
[root@rpgmaster ~]# vi /etc/httpd/conf.d/userdir.conf

    17      # UserDir disabled        //주석 처리해 기능을 활성화 시킵니다.
    24      UserDir public_html       //이번엔 주석을 제거해 활성화 시킵니다.

    31 <Directory "/home/*/public_html">
    32      AllowOverride All
    33      Options None
    34      Require method GET POST OPTIONS
    35 </Directory>

[root@rpgmaster ~]# systemctl restart httpd
```

[2] SELinux설정하기

```
Userdir에 대한 액세스 권한을 열어주어야 겠죠?
[root@rpgmaster ~]# setsebool -P httpd_enable_homedirs on
[root@rpgmaster ~]# restorecon -R /home
```

[3] 테스트 페이지 만들기

Userdir을 테스트하려면 우선 유저를 생성해야 겠죠? 여기서는 cent라는 유저를 만들어서 테스트해보도록 하겠습니다.

```
[root@rpgmaster ~]# useradd cent
[root@rpgmaster ~]# su cent
[cent@rpgmaster root]$ cd ~
[cent@rpgmaster ~]$ mkdir public_html
[cent@rpgmaster ~]$ chmod 711 /home/cent
[cent@rpgmaster ~]$ chmod 755 /home/cent/public_html
[cent@rpgmaster ~]$ vi ./public_html/index.html

<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
UserDir Test Page
</div>
</body>
</html>
```

[4] 페이지 확인(rpgmaster.com/~cent/)

5.1.6. Virtual Hostings

이제 별칭을 사용한 가상호스팅을 구성해 보도록 하겠습니다. 아래 예제는 도메인이름이 rpgmaster.com이고 가상도메인이름이 virtual.host(/home/cent/public_html)인 환경으로 설정했습니다.(이해하기 쉽도록 이름을 완전히 다르게 구성했습니다.) 일반적으로는 userid.rpgmaster.com 같은 형태일 겁니다. 당연히 Userdir은 설정이 되어 있어야 겠죠?

[1] Virtual Hosting 설정

```
[root@rpgmaster ~]# vi /etc/httpd/conf.d/vhost.conf
아래와 같이 내용을 작성해 줍니다.
# original domain
<VirtualHost *:80>
    DocumentRoot /var/www/html
    ServerName rpgmaster.com
</VirtualHost>
# virtual domain
<VirtualHost *:80>
    DocumentRoot /home/cent/public_html
    ServerName virtual.host
    ServerAdmin root@virtual.host
    ErrorLog logs/virtual.host-error_log
    CustomLog logs/virtual.host-access_log combined
</VirtualHost>

[root@rpgmaster ~]# systemctl restart httpd
```

[2] 테스트페이지 작성

```
간단하게 테스트 페이지를 작성해 봅시다. 당연히 cent유저로 작업해야 겠죠?
[root@rpgmaster ~]# su - cent
마지막 로그인: 화  5월 14 22:34:04 KST 2019 일시 pts/1
[cent@rpgmaster ~]$ vi ~/public_html/virtual.php

<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
Virtual Host Test Page
</div>
</body>
</html>
```

[3] 테스트페이지 확인(virtual.host/virtual.php)

5.1.7. SSL/TLS 설정

이번에는 암호화를 통해 보안이 강화된 https 연결을 사용하도록 SSL/TLS를 구성해 봅시다. https를 구성하려면 우선 SSL인증서가 필요합니다.

SSL 또는 TLS인증서는 웹사이트를 운영하면서 거의 필수적인 요소입니다. 클라이언트와 서버간의 통신이 발생할 때 전송되는 모든 패킷 데이터를 암호화하여 감청이나 식별을 어렵게하여 보안에 있어 강력한 역할을 합니다.

웹사이트 인증서를 발급해주는 대표적인 발급기관(CA)으로는 Verisign, Comodo나 GlobalSign등이 있으며 대개 호스팅 업체에서 등록을 대행해 주기도 합니다.

무료 SSL인증서를 제공하는 Let's Encrypt에서 SSL인증서를 가져오도록 합시다. 또한 인증서의 만료일은 90일이므로 다음 90일 이내에 업데이트해야 합니다.

또한 터미널에 접속하여 root권한을 사용할 수 있어야 합니다. 일부 호스팅 업체에선 root권한 획득을 제한하기도 하기 때문에 웹호스팅 서비스를 사용중이라면 해당업체에 문의하여 사용/설치 가능 여부를 확인해보셔야 합니다.

Let's Encrypt는 퍼블릭 도메인이 할당된 서버에서만 발급이 가능하기 때문에 내부 테스트용으로 구성된 서버이거나 공개서버가 아닌등의 이유로 자신의 서버에 IP만 할당되어 있는 경우에는 인증서 발급과 설치에 어려움이 있으므로 반드시 도메인을 할당해주어야 합니다.

모든 준비가 되었다면 Certbot을 설치해 보도록 합시다.

[1] Certbot Client설치

```
Certbot을 이용해 인증서를 가져올 것입니다. Certbot은 기본리포지터리에 등재되어 있지 않기 때문에 epel리포지터리를 사용해야 합니다.
[root@rpgmaster ~]# yum --enablerepo=epel -y install certbot

또한 웹서비스에 맞는 플러그인도 설치해 줘야 합니다.
[root@rpgmaster ~]# yum -y install python2-certbot-apache
```

[2] 인증서 생성

인증서를 발급 받을 때 주의할 점은 하나의 호스트 또는 도메인에서 1일에 3회 이상의 발급을 시도할 수 없기 때문에 발급 절차 시 실수를 하지 않도록 해야 한다는 것입니다.

인증서를 발급받는 방법은 webroot와 Standalone, DNS의 세가지 방식이 있습니다.

먼저 webroot 방식은 실제 웹 디렉토리 내에 인증서의 유효성을 확인할 수

있는 파일을 업로드하여 인증서를 발급하는 방법입니다. 웹서비스의 중단없이 인증서를 발급 받을 수 있는 장점이 있지만 한 번의 명령에 하나의 도메인 인증서만 발급받을 수 있습니다.
다음으로 Standalone 방식은 일시적으로 호스트 내의 웹 서비스를 빌려 인증서 유효성을 확인하는 방법입니다. 여러 도메인을 발급받을 수 있으나 이 방법을 사용하면 인증서가 발급되는 동안 운영 중인 웹 서비스가 잠시 중단되어야 합니다.
마지막으로 DNS 방식은 도메인을 쿼리하여 나타나는 TXT 레코드에서 인증서 유효성을 확인하는 방법입니다. 이 경우 해당 도메인의 DNS를 관리/수정할 수 있는 조건이 되어야 합니다.
결론적으로 인증 기관(CA)이 발급할 서버에 방문하여 인증서와 동일한 도메인인지 확인하는 과정이 진행되며, 이러한 과정에 ACME 프로토콜을 사용하여 CA와 서버 간의 유효성 검증을 시도(Challenge) 합니다. 따라서 투명성을 위해 인증 도중 서버의 IP와 트랜잭션 내역이 인증기관에 기록되게 됩니다.
다양한 발급방법이 있고, 또한 각각의 장단점이 있으니 상황에 따라 방법을 선택하여 진행하실 수 있도록 해야 합니다.
설치에 앞서 일반적인 명령어 사용 옵션에 대해 살펴봅시다.
--apache / -nginx : 기본 명령에 웹서비스이름을 옵션값으로 붙여주어 해당 서비스에 맞는 발급과정을 진행하도록 합니다.
--standalone / --webroot : 인증서 발급 방식을 결정합니다. DNS방식을 사용할 경우 --preferred-challenges dns 옵션을 사용합니다.
-d [도메인] : 인증서를 받고자 하는 FQDN을 입력합니다. 여러 도메인을 사용하는 경우 콤마(,)를 이용해 구분하여 입력하면 됩니다.
certonly : 원래는 인증서 발급시 웹서비스의 설정파일을 직접 편집해 주어야 합니다. certonly값을 붙이면 웹서비스 설정파일에 인증서 관련 내용을 임의로 수정하지 않도록 합니다.
이제 설치를 진행해 보도록 합시다.
webroot방식으로 진행할 경우 유효한 메일주소가 필요하기 때문에 여기서는 Standalone방식으로 진행해 보겠습니다.

```
앞에서 설명한 것과 같이 standalone방식은 서비스가 중단되어야 합니다.

[root@rpgmaster ~]# systemctl stop httpd
```

Standalone방식의 경우 Certbot의 웹서버 기능을 사용하여 인증서를 얻습니

다. 하지만 어쨌든 Let's Encrypt의 확인이 필요하기 때문에 80번 포트에서 인터넷에서 작업 서버로 액세스 할 수 있어야 합니다.

```
[root@rpgmaster ~]# certbot certonly —standalone –d rpgmaster.com

다음과 같은 내용이 나타나면 도메인 관리자의 이메일 주소를 입력해 줍니다. 해당 이메일로 갱신 알림이나 주요한 소식들이 발송될 수 있습니다.

Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator webroot, Installer None
Enter email address (used for urgent renewal and security notices)
# for only initial using, register your email address and agree to terms of use
# specify valid email address
(Enter 'c' to cancel): root@mail.rpgmaster.com
Starting new HTTPS connection (1): acme-v01.api.letsencrypt.org

이번에는 이용원칙 동의 및 인증기관에 등록되는 사항에 관한 내용입니다.
어차피 동의해야 하므로 A를 입력하고 엔터를 입력해 줍니다.

-------------------------------------------------------------------------------
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You must
agree in order to register with the ACME server at
https://acme-v01.api.letsencrypt.org/directory
-------------------------------------------------------------------------------
# agree to the terms of use
(A)gree/(C)ancel: A

다음 나오는 내용은 제3자 업체에게 정보를 공유하겠냐고 하는 내용입니다. 원치 않는 경우 N을 입력하여 거부할 수 있습니다.

-------------------------------------------------------------------------------
Would you be willing to share your email address with the Electronic Frontier
Foundation, a founding partner of the Let's Encrypt project and the non-profit
organization that develops Certbot? We'd like to send you email about EFF and
our work to encrypt the web, protect its users and defend digital rights.
-------------------------------------------------------------------------------
# answer Yes or No
(Y)es/(N)o: Y
```

```
이제 특별한 문제가 나타나지 않는다면 발급과정이 진행됩니다.

Starting new HTTPS connection (1): supporters.eff.org
Obtaining a new certificate
Performing the following challenges:
http-01 challenge for www.srv.world
Using the webroot path /var/www/html for all unmatched domains.
Waiting for verification...
Cleaning up challenges

IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/www.srv.world/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/www.srv.world/privkey.pem
   Your cert will expire on 2018-05-22. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - Your account credentials have been saved in your Certbot
   configuration directory at /etc/letsencrypt. You should make a
   secure backup of this folder now. This configuration directory will
   also contain certificates and private keys obtained by Certbot so
   making regular backups of this folder is ideal.
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:    https://letsencrypt.org/donate
   Donating to EFF:                     https://eff.org/donate-le
```

위와 같이 Congratulations!가 나타났다면 발급이 정상적으로 완료된 것입니다. 인증서는 [/etc/letsencrypt/live/(FQDN)/] 디렉토리에 생성됩니다.
생성된 인증서 파일은 다음과 같습니다.
cert.pem ⇒ SSL 서버 인증서 (공개 키 포함)
chain.pem ⇒ 인증서 체인
fullchain.pem ⇒ cert.pem 및 chain.pem 결합 파일
privkey.pem ⇒ 개인 키 파일
여기서 잠깐 인증서 체인에 대해 조금 알아보고 넘어가도록 합시다. 인증기관(CA, Certificate Authority)은 서로 간에 어떤 계층관계가 있습니다. 루트 CA는 자기-서명 인증서를 가지고 있습니다. 하지만 그 하부 CA들은 자기 직

속상관이 발급한 인증서를 가지고 있습니다. 그래서 특정CA는 이런 인증서들의 묶음을 가지게 되는데 이를 인증서체인이라고 합니다. 인증서 체인은 자기 직속 상위 CA가 발급한 인증서를 가지고 있고, 그 상위 CA는 또 그 위의 상위 CA가 발급한 인증서를 가지고 있는 식입니다.

다시 Certbot의 사용으로 넘어와서, 마지막으로 기존인증서를 업데이트 하는 것은 certbot renew를 입력하시면 됩니다.

```
[root@rpgmaster ~]# certbot renew
```

여기까지 인증서를 사용하는 방법이었습니다.

이제 SSL/TLS에 httpd를 구성해야 합니다.

```
mod_ssl을 설치합니다.
[root@rpgmaster ~]# yum -y install mod_ssl

아파치 설정에서 ssl을 사용하도록 설정합니다.
[root@rpgmaster ~]# vi /etc/httpd/conf.d/ssl.conf

# line 59: 주석을 제거해 줍니다.
DocumentRoot "/var/www/html"
# line 60: 주석을 제거하고 도메인 주소를 입력해줍니다.
ServerName rpgmaster.com:443
# line 75: 아래와 같이 변경해 줍니다.
SSLProtocol -All +TLSv1 +TLSv1.1 +TLSv1.2
# line 100: 앞에서 받았던 인증서의 경로 정보를 입력해 줍니다.
SSLCertificateFile /etc/letsencrypt/live/rpgmaster.com/cert.pem
# line 107: 마찬가지로 앞에서 받았던 인증서의 경로 정보를 입력해 줍니다.
SSLCertificateKeyFile /etc/letsencrypt/live/rpgmaster.com/privkey.pem
# line 116: 역시나 마찬가지..
SSLCertificateChainFile /etc/letsencrypt/live/www.srv.world/chain.pem
이제 아파치를 재시작 합니다.
[root@rpgmaster ~]# systemctl restart httpd
```

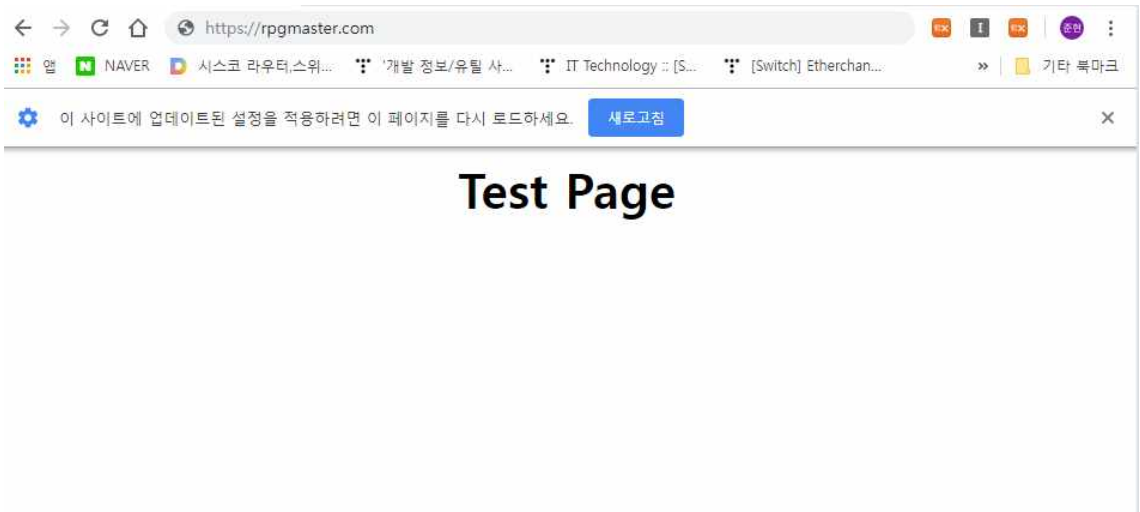
만약 항상 HTTPS로 리다이렉션 하도록 HTTP연결을 설정하려면 다음과 같이 각 가상 호스트를 구성하시면 됩니다.

```
[root@rpgmaster ~]# vi /etc/httpd/conf.d/vhost.conf
<VirtualHost *:80>
    DocumentRoot /var/www/html
    ServerName rpgmaster.com
    RewriteEngine On
    RewriteCond %{HTTPS} off
    RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [R=301,L]
</VirtualHost>
```

이제 마지막으로 방화벽에서 HTTPS 서비스를 추가해 주시면 됩니다.

```
[root@rpgmaster ~]# firewall-cmd --add-service=https --permanent
success
[root@www ~]# firewall-cmd --reload
success
```

이제 PC에서 https를 사용하여 액세스 할 수 있는지 확인해 보시면 됩니다.



5.1.8 기본 인증 사용하기

특정 웹페이지에 대한 엑세스를 제한하기 위한 기본인증설정을 사용할 수 있습니다. 백문이 불여 일견! /var/html/auth-basic디렉터리에 기본인증 설정을 지정해 보도록 합시다.

```
[root@rpgmaster ~]# vi /etc/httpd/conf.d/auth_basic.conf
파일을 생성하고 아래와 같이 설정합니다.
<Directory /var/www/html/auth-basic>
    AuthType Basic
    AuthName "Basic Authentication"
    AuthUserFile /etc/httpd/conf/.htpasswd
    require valid-user
</Directory>
```

```
새로운 유저를 생성하고 –c 옵션을 사용해 새파일을 작성합니다. 초기 등록에 대해서만 –c옵션을 추가하시면 됩니다.
[root@rpgmaster ~]# htpasswd -c /etc/httpd/conf/.htpasswd cent
New password: # set password
Re-type new password: # confirm
Adding password for user cent
비밀번호 설정이 끝났다면, 아파치를 재시작한 후
[root@rpgmaster ~]# systemctl restart httpd
디렉터리를 생성하고
[root@rpgmaster ~]# mkdir /var/www/html/auth-basic
새로 테스트할 페이지를 만들어 봅시다.
[root@rpgmaster ~]# vi /var/www/html/auth-basic/index.html
<html>
<body>
<div style="width: 100%; font-size: 40px; font-weight: bold; text-align: center;">
Test Page for Basic Auth
</div>
</body>
</html>
```

이제 페이지를 테스트해 봅시다. (rpgamster.com/auth-basic)

위와 같이 사용자 이름과 암호를 입력하도록 하는 페이지가 나타납니다.
아까 설정해두었던 cent 사용자와 암호를 입력해 주시면 아래와 같이 설정해 두었던 페이지가 나타나는 것을 확인하실 수 있습니다.

5. Storage Server

5.1. NFS

네트워크에서 디렉터리를 공유할 수 있도록 NFS서버를 구성해 보도록 합시다.

5.1.1. Configure NFS Server

테스트 환경은 다음과 같이 구성합니다.

우선 NFS Server부터 설정해 봅시다.

```
[root@srv ~]# yum -y install nfs-utils

[root@srv ~]# vi /etc/idmapd.conf
# 도메인이름을 변경하고 주석을 제거해 줍니다.
       5 Domain = rpgmaster.com

[root@srv ~]# vi /etc/exports
# NFS에서 사용될 네트워크를 설정합니다.
/home 10.0.0.0/24(rw,no_root_squash)

[root@srv ~]# systemctl start rpcbind nfs-server
[root@srv ~]# systemctl enable rpcbind nfs-server
```

이제 방화벽에 NFS서비스를 추가해 줍니다.

```
# NFSv4
[root@srv ~]# firewall-cmd --add-service=nfs --permanent
success

# NFSv3
[root@srv ~]# firewall-cmd --add-service={nfs3,mountd,rpc-bind} --permanent
success

[root@srv ~]# firewall-cmd --reload
success
```

5.1.2. Configure NFS Client

이제 NFS Client에서의 설정입니다. 네트워크 드라이브를 마운트하고 확인해 봅시다.

```
[root@localhost ~]# yum -y install nfs-utils

[root@localhost ~]# vi /etc/idmapd.conf
# 도메인이름을 변경하고 주석을 제거해 줍니다. 서버와 같아야 겠죠?
       5 Domain = rpgmaster.com

[root@localhost ~]# systemctl start rpcbind
[root@localhost ~]# systemctl enable rpcbind

# 네트워크 드라이브를 마운트 합니다.
[root@localhost ~]# mount -t nfs srv.rpgmaster.com:/home /home
[root@localhost ~]# df -hT
Filesystem              Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos-root xfs        17G  1.1G   16G   7% /
devtmpfs                devtmpfs  898M     0  898M   0% /dev
tmpfs                   tmpfs     910M     0  910M   0% /dev/shm
tmpfs                   tmpfs     910M  9.6M  901M   2% /run
tmpfs                   tmpfs     910M     0  910M   0% /sys/fs/cgroup
/dev/sda1               xfs      1014M  146M  869M  15% /boot
tmpfs                   tmpfs     182M     0  182M   0% /run/user/0
srv.rpgmaster.com:/home nfs4       26G  5.9G   21G  23% /home
```

이제 이 드라이브가 부팅 시에 자동으로 마운트 될 수 있도록 fstab을 설정해 줍시다.

```
[root@localhost ~]# vi /etc/fstab

# 마지막줄에 아래와 같이 추가해 줍니다.
srv.rpgmaster.com:/home                    nfs     defaults        0 0
```

5.1.3. NFSv4 ACL Tool

NFSv4에서 접근제어가 가능하도록 해 봅시다. 이를 위해선 파일시스템에 ext4나 xfs의 ACL설정이 되어 있어야 합니다.

우선은 ACL(Access Control List) 설정을 할 수 있도록 해 봅시다.

구성은 위에서 사용한 걸 그대로 사용할 수 있도록 해 봅시다.

```
[root@localhost ~]# yum -y install acl

# 테스트를 위해 임의의 사용자를 추가해줍니다.
[root@localhost ~]# useradd cent
[root@localhost ~]# passwd cent

# ACL을 테스트할 파일도 만들어 줍시다.
[root@localhost ~]# touch /home/test.txt
[root@srv ~]# chmod 700 /home/test.txt
[root@localhost ~]# ll /home/test.txt
-rwx------. 1 root root 0  5월 21 14:32 /home/test.txt

[root@localhost ~]# setfacl -m u:cent:r /home/test.txt
[root@localhost ~]# ll /home/test.txt
-rwx------+ 1 root root 0  5월 21 14:32 /home/test.txt
퍼미션 부분이 변경된 것이 보이시나요? ACL을 설정하면 속성부분에 +가 추가됩니다.

설정을 확인해 봅시다.
[root@localhost ~]# getfacl /home/test.txt
getfacl: Removing leading '/' from absolute path names
# file: home/test.txt
# owner: root
# group: root
user::rw-
user:cent:r--
group::r--
mask::r--
other::r--

이제 ACL이 제대로 적용되는지 테스트해 봅시다. 일반유저로 접속해 봅시다.
우선 읽기 권한을 주었던 cent계정으로 접속해 봅니다.
[root@srv ~]# su cent
[cent@srv root]$ cat /home/test.txt
ACL test file

당연히 잘 읽어집니다. 이번엔 다른 계정으로 접속해 봅시다.

[root@srv ~]# su RHEL
[RHEL@srv root]$ cat /home/test.txt
cat: /home/test.txt: 허가 거부
```

이번엔 디렉터리에 ACL설정을 해 보도록 합시다.
먼저 디렉터리를 만들어 주고,

```
[root@srv ~]# mkdir /home/testdir
```

테스트용 파일도 하나 만들어 줍시다. 권한도 설정해주고요.

```
[root@srv ~]# touch /home/testdir/testfile
[root@srv ~]# chmod 700 /home/testdir/testfile
```

ACL을 설정합니다. 하위디렉터리까지 모두 설정이 되도록 -R옵션을 줍시다.

```
[root@srv ~]# setfacl -R -m u:cent:r /home/testdir
[root@srv ~]# ll /home/testdir/
합계 0
-rwxr-----+ 1 root root 0  5월 21 15:28 testfile
```

위와 같이 디렉터리 안에 있는 파일도 같이 설정이 적용된 것을 볼 수 있습니다.

한번 확인해 봅시다.

```
[root@srv ~]# getfacl -R /home/testdir
getfacl: Removing leading '/' from absolute path names
# file: home/testdir
# owner: root
# group: root
user::rwx
user:cent:r--
group::r-x
mask::r-x
other::r-x

# file: home/testdir/testfile
# owner: root
# group: root
user::rwx
user:cent:r--
group::---
mask::r--
other::---
```

```
이번엔 그룹으로 적용해 봅시다. security라는 그룹을 만들어서 테스트해 보겠습니다.
```

퍼미션을 통해서 파일에 대한 접근권한을 설정할 수도 있지만, 이처럼 ACL을 통해 좀더 세밀한 접근권한 설정이 가능합니다. 또한 ACL을 사용하면 사용/접근 로그가 남기 때문에 좀더 보안에 신경을 쓸 수 있겠죠?