

# **S-PASS Installation Guide**

**Ver 1.1**

**2020.12.17**

**한승우**

# Table of Contents

1.	Folder List.....	3
2.	Turn Server.....	3
2.1	TurnServer 설치 파일 경로.....	3
2.2	컴파일을 위한 gcc c++ 설치.....	3
2.3	TurnServer 파일 압축 해제.....	3
2.4	TurnServer (..client-libs, ..utils)rpm 설치.....	4
2.5	TurnServer 참조 rpm 설치.....	5
2.6	TurnServer 참조 rpm(perl-HTTP-Message..) 설치 .....	5
2.7	TurnServer 참조 rpm (perl-libwww-perl..) 설치.....	6
2.8	TurnServer 설치 .....	6
2.9	TurnServer 설정 .....	7
2.10	TurnServer 실행/종료 .....	7
2.11	TurnServer 서비스 확인.....	7
3.	Nginx.....	8
3.1	Nginx 설치 파일 경로.....	8
3.2	Nginx 파일 압축 해제.....	8
3.3	Nginx 컴파일 .....	8
3.4	Nginx 설치.....	9
3.5	Nginx 설정.....	10
3.6	Nginx 실행/종료 .....	11
3.7	Nginx 서비스 확인 .....	12
4.	Nodejs(Signaling) .....	12
4.1	Nodejs 설치 파일 경로 .....	12
4.2	Nodejs 파일 압축해제.....	12
4.3	Nodejs 소스 배포 .....	12
4.4	Nodejs Path 추가 .....	13
4.5	Nodejs 실행/종료.....	13
5.	S-pass 솔루션 .....	13
5.1	S-pass 소스배포.....	13

6.	Turn Proxy JAVA .....	13
6.1	Turn Proxy 설정 파일 경로 .....	13
6.2	Turn Proxy 설정 추가 .....	13
7.	Turn Proxy Nginx .....	14
7.1	Nginx 설정 추가 .....	14

## 1. Folder List

```
/sw/turnserver :      # Turn Server 설치 파일
/sw/nginx :         # Nginx 설치 파일
/sw/nodejs :        # Nodejs 설치 파일
/sw/util/gcc :      # 컴파일 rpm, 참조 rpm.
```

## 2. Turn Server

### 2.1 TurnServer 설치 파일 경로

```
/sw/turnserver : turnserver-3.2.5.9-CentOS7.1-x86_64.tar.gz
/sw/util/gcc : 컴파일 파일 위치 및 경로
```

### 2.2 컴파일을 위한 gcc c++ 설치

```
# rpm -ivh mpfr-3.1.1-4.el7.x86_64.rpm
# rpm -ivh libmpc-1.0.1-3.el7.x86_64.rpm
# rpm -ivh libgcc-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -Uvh libstdc++-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -Uvh libstdc++-devel-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -ivh kernel-headers-3.10.0-862.11.6.el7.x86_64.rpm
# rpm -ivh glibc-headers-2.17-222.el7.x86_64.rpm
# rpm -ivh glibc-devel-2.17-222.el7.x86_64.rpm
# rpm -Uvh libgcc-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -Uvh libgomp-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -ivh cpp-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -ivh gcc-4.8.5-28.el7_5.1.x86_64.rpm
# rpm -ivh gcc-c++-4.8.5-28.el7_5.1.x86_64.rpm
```

### 2.3 TurnServer 파일 압축 해제

```
# tar -xvzf turnserver-3.2.5.9-CentOS7.1-x86_64.tar.gz  
  
turnserver-3.2.5.9/  
  
turnserver-3.2.5.9/turnserver-3.2.5.9-0.el7.centos.x86_64.rpm  
  
turnserver-3.2.5.9/turnserver-client-libs-3.2.5.9-0.el7.centos.x86_64.rpm  
  
turnserver-3.2.5.9/turnserver-utils-3.2.5.9-0.el7.centos.x86_64.rpm  
  
turnserver-3.2.5.9/install.sh  
  
turnserver-3.2.5.9/uninstall.turnserver.sh
```

## 2.4 TurnServer (..client-libs, ..utils)rpm 설치

- turnserver 관련 rpm 은 2개만 설치합니다.

```
# rpm -ivh libevent-2.0.21-4.el7.x86_64.rpm  
  
# rpm -ivh turnserver-client-libs-3.2.5.9-0.el7.centos.x86_64.rpm  
  
# rpm -ivh turnserver-utils-3.2.5.9-0.el7.centos.x86_64.rpm
```

## 2.5 TurnServer 참조 rpm 설치.

```
[..turnserver]
# rpm -ivh telnet-0.17-64.el7.x86_64.rpm
# rpm -ivh postgresql-libs-9.3.4-1.el7.x86_64.rpm
# rpm -ivh hiredis-0.12.1-1.el7.x86_64.rpm#
# rpm -ivh mailcap-2.1.41-2.el7.noarch.rpm
# rpm -ivh net-tools-2.0-0.22.20131004git.el7.x86_64.rpm
# rpm -Uvh nspr-4.19.0-1.el7_5.x86_64.rpm
# rpm -Uvh nss-util-3.36.0-1.el7_5.x86_64.rpm
# rpm -Uvh nss-softokn-freebl-3.36.0-5.el7_5.x86_64.rpm
[.gcc]# rpm -ivh perl*.rpm
1: perl-HTTP-Tiny-0.033-3.el7.noarch.rpm
2: perl-parent-0.225-244.el7.noarch.rpm
3: perl-Pod-Perldoc-3.20-4.el7.noarch.rpm
4: perl-podlators-2.5.1-3.el7.noarch.rpm
5: perl-Pod-Escapes-1.04-292.el7.noarch.rpm
6: perl-Text-ParseWords-3.29-4.el7.noarch.rpm
7: perl-Encode-2.51-7.el7.x86_64.rpm
8: perl-Pod-Usage-1.63-3.el7.noarch.rpm
9: perl-Carp-1.26-244.el7.noarch.rpm
10: perl-Exporter-5.68-3.el7.noarch.rpm
11: perl-Filter-1.49-3.el7.x86_64.rpm
12: perl-File-Path-2.09-2.el7.noarch.rpm
13: perl-PathTools-3.40-5.el7.x86_64.rpm
14: perl-Scalar-List-Utils-1.27-248.el7.x86_64.rpm
15: perl-Socket-2.010-4.el7.x86_64.rpm
16: perl-Storable-2.45-3.el7.x86_64.rpm
17: perl-Time-HiRes-1.9725-3.el7.x86_64.rpm
18: perl-File-Temp-0.23.01-3.el7.noarch.rpm
19: perl-Time-Local-1.2300-2.el7.noarch.rpm
20: perl-constant-1.27-2.el7.noarch.rpm
21: perl-libs-5.16.3-292.el7.x86_64.rpm
22: perl-macros-5.16.3-292.el7.x86_64.rpm
23: perl-Pod-Simple-3.28-4.el7.noarch.rpm
24: perl-Getopt-Long-2.40-3.el7.noarch.rpm
25: perl-threads-1.87-4.el7.x86_64.rpm
26: perl-threads-shared-1.43-6.el7.x86_64.rpm
27: perl-5.16.3-292.el7.x86_64.rpm
```

## 2.6 TurnServer 참조 rpm(perl-HTTP-Message..) 설치

```
[.perl-HTTP-Message-6.06-6.el7.noarch.rpm]# rpm -ivh perl*.rpm
1: perl-Data-Dumper-2.145-3.el7.x86_64.rpm
2: perl-TimeDate-2.30-2.el7.noarch.rpm
3: perl-Net-Daemon-0.48-5.el7.noarch.rpm
4: perl-Compress-Raw-Zlib-2.061-4.el7.x86_64.rpm
5: perl-Compress-Raw-Bzip2-2.061-3.el7.x86_64.rpm
6: perl-IO-Compress-2.061-2.el7.noarch.rpm
7: perl-PIRPC-0.2020-14.el7.noarch.rpm
8: perl-Business-ISBN-Data-20120719.001-2.el7.noarch.rpm
9: perl-Business-ISBN-2.06-2.el7.noarch.rpm
10: perl-URI-1.60-9.el7.noarch.rpm
11: perl-DBI-1.627-4.el7.x86_64.rpm
12: perl-HTTP-Date-6.02-8.el7.noarch.rpm
13: perl-LWP-MediaTypes-6.02-2.el7.noarch.rpm
14: perl-IO-HTML-1.00-2.el7.noarch.rpm
15: perl-Encode-Locale-1.03-5.el7.noarch.rpm
# rpm -ivh perl-HTTP-Message-6.06-6.el7.noarch.rpm
```

## 2.7 TurnServer 참조 rpm (perl-libwww-perl..) 설치

```
[.perl-libwww-perl-6.05-2.el7.noarch.rpm]# rpm -ivh perl*.rpm
1: perl-IO-Socket-IP-0.21-5.el7.noarch.rpm
2: perl-Net-SSLeay-1.55-6.el7.x86_64.rpm
3: perl-Net-LibIDN-0.12-15.el7.x86_64.rpm
4: perl-Mozilla-CA-20130114-5.el7.noarch.rpm
5: perl-IO-Socket-SSL-1.94-7.el7.noarch.rpm
6: perl-HTML-Tagset-3.20-15.el7.noarch.rpm
7: perl-Digest-1.17-245.el7.noarch.rpm
8: perl-Digest-MD5-2.52-3.el7.x86_64.rpm
9: perl-HTML-Parser-3.71-4.el7.x86_64.rpm
10: perl-Net-HTTP-6.06-2.el7.noarch.rpm
11: perl-HTTP-Daemon-6.01-7.el7.noarch.rpm
12: perl-WWW-RobotRules-6.02-5.el7.noarch.rpm
13: perl-HTTP-Negotiate-6.01-5.el7.noarch.rpm
14: perl-HTTP-Cookies-6.01-5.el7.noarch.rpm
15: perl-HTML-Entities-Numbered-0.04-13.el7.noarch.rpm
16: perl-File-Listing-6.04-7.el7.noarch.rpm
# rpm -ivh perl-libwww-perl-6.05-2.el7.noarch.rpm
```

## 2.8 TurnServer 설치

- TurnServer 설치 기본경로는 /etc/turnserver 입니다.

```
# rpm -ivh turnserver-3.2.5.9-0.el7.centos.x86_64.rpm

[..etc/turnserver]# turnserver.conf

[..etc/turnserver]# turnuserdb.conf
```

## 2.9 TurnServer 설정

```
[../sw/middle/turnserver]# vi turnserver.conf

.....

listening-port=8100          # 서비스 Port.
alt-listening-port=8199     # udp 설정 사용시 방화벽 추가 설정 필요. listening-port 보다 1작은값 사용.
user=spass:spass            # 관리자 아이디 비밀번호.(Turn server 영상 연결시 필요)
no-udp                      # udp 사용하지 않음.
#no-udp                     # udp 사용시 no-udp 주석처리 외 port 대역 추가
min-port=49152
max-port=65535
log-file=/sw/middle/turnserver/logs/turn.log      # 로그파일 경로 설정.
no-stun                      # stun 사용하지 않음.
```

## 2.10 TurnServer 실행/종료

```
[../sw/middle/turnserver]

# ./turnserver_start.sh

    #!/bin/bash

    turnserver -o -c /sw/middle/turnserver/turnserver.conf


# ./turnserver_stop.sh

    #!/bin/bash

    pkill turnserver


./turnserver_start.sh
....
0: log file opened: /var/log/turn_1300_2019-04-30.log
....
0: =====Discovering listener addresses: =====
0: Listener address to use: 127.0.0.1
0: Listener address to use: 192.168.56.101
0: Listener address to use: ::1
0: =====
0: Total: 1 'real' addresses discovered
0: =====
```

## 2.11 TurnServer 서비스 확인.

```
# netstat -nltp |grep 8100

tcp        0      0 192.168.56.101:8100  0.0.0.0:*        LISTEN      1301/turnserver
tcp        0      0 127.0.0.1:8100      0.0.0.0:*        LISTEN      1301/turnserver
tcp6       0      0 :::8100              :::*              LISTEN      1301/turnserver

# ps -ef |grep turnserver
root      1301      1  0 09:45 ?        00:00:00 turnserver -o -c /sw/middle/turnserver/turnserver.conf

# netstat -nltp |grep 8100  udp 설정확인시
```

## 3. Nginx

### 3.1 Nginx 설치 파일 경로

```
/sw/nginx : nginx-1.14.0.tar.gz
/sw/nginx : openssl-1.1.1.tar.gz
/sw/nginx : pcre-8.38.tar.gz
/sw/nginx : zlib-1.2.11.tar.gz
```

### 3.2 Nginx 파일 압축 해제

```
tar -xvzf nginx-1.14.0.tar.gz
tar -xvzf pcre-8.38.tar.gz
tar -xvzf openssl-1.1.1.tar.gz
tar -xvzf zlib-1.2.11.tar.gz
```

### 3.3 Nginx 컴파일



```
# cd nginx-1.14.0
# ./configure
--with-zlib=../zlib-1.2.11
--with-pcre=../pcre-8.38
--with-openssl=../openssl-1.1.1
--with-http_ssl_module
--with-debug
--with-http_stub_status_module
--with-http_addition_module
--with-http_degradation_module
--with-http_flv_module
--with-http_gzip_static_module
--with-http_mp4_module
--with-http_random_index_module
--with-http_realip_module
--with-http_sub_module
--with-http_realip_module
--with-stream=dynamic          #strem 방식 사용시(내부구간서버)
--prefix=/sw/middle/nginx
Configuration summary
+ using PCRE library: ../pcre-8.38
```

### 3.4 Nginx 설치

```
# make && make install

....

test -f '/sw/middle/nginx/conf/nginx.conf' &#x2194;
    || cp conf/nginx.conf '/sw/middle/nginx/conf/nginx.conf'
cp conf/nginx.conf '/sw/middle/nginx/conf/nginx.conf.default'
test -d '/sw/middle/nginx/logs' &#x2194;
    || mkdir -p '/sw/middle/nginx/logs'
test -d '/sw/middle/nginx/logs' &#x2194;
    || mkdir -p '/sw/middle/nginx/logs'
test -d '/sw/middle/nginx/html' &#x2194;
    || cp -R html '/sw/middle/nginx'
test -d '/sw/middle/nginx/logs' &#x2194;
    || mkdir -p '/sw/middle/nginx/logs'
make[1]: Leaving directory `/sw/nginx/nginx-1.14.0'
```

\* command 방식이 아닌 별도 .sh 생성으로 작업가능.

```
#!/usr/bin/env bash
```

```
./configure
--with-http_ssl_module
--with-http_stub_status_module
--with-http_addition_module
--with-http_degradation_module
--with-http_flv_module
--with-http_gzip_static_module
--with-http_mp4_module
--with-http_random_index_module
--with-http_realip_module
--with-http_sub_module
--with-http_realip_module
--with-stream=dynamic
--with-pcre=../pcre-8.41
--with-zlib=../zlib-1.2.11
--with-debug
--prefix=/shbdev/nofacedomain/nginx_dev

make install
```

### 3.5 Nginx 설정

```
[/sw/middle/nginx/conf]# vi nginx.conf
```

```
# WSS Server
```

```
server {  
    listen      8200 ssl;          # 서비스 Port.  
  
    server_name localhost;  
  
    access_log  /sw/middle/nginx/logs/acc_log_8200.log;      # 로그파일 경로 설정.  
    error_log   /sw/middle/nginx/logs/err_log_8200.log;  
  
    ssl_certificate      /sw/middle/nginx/cert/nginx_cert.crt      # 인증서 경로  
    ssl_certificate_key  /sw/middle/nginx/cert/nginx_key.key;  
  
    location / {  
        proxy_pass      https://192.168.56.102:18200;          # Proxy 연결 IP / Port  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection "upgrade";  
        proxy_read_timeout 3600;  
        proxy_set_header Host $host;  
        proxy_set_header X-Real-IP $remote_addr;  
        proxy_set_header X-Forwarded-For $remote_addr;  
        proxy_set_header X-Forwarded-Host $remote_addr;  
    }  
}
```

```
# HTTPS Server
```

```
server {  
    listen      8300 ssl;          # 서비스 Port.  
  
    server_name localhost;  
  
    access_log  /sw/middle/nginx/logs/acc_log_8300.log;      # 로그파일 경로 설정.  
    error_log   /sw/middle/nginx/logs/err_log_8300.log;  
  
    ssl_certificate      /sw/middle/nginx/cert/cert/nginx_cert.crt;      # 인증서 경로  
    ssl_certificate_key  /sw/middle/nginx/cert/cert/nginx_key.key;  
  
    location / {  
        proxy_pass      https://192.168.56.102:18300;          # Proxy 연결 IP / Port  
        proxy_max_temp_file_size 0;  
        client_max_body_size 500M;  
        client_body_buffer_size 128k;  
    }  
}
```

### 3.6 Nginx 실행/종료

## 기본설정

```
# ./start_nginx.sh

#!/bin/bash

./sbin/nginx -c /sw/middle/nginx/conf/nginx.conf &

# ./stop_nginx.sh

#!/bin/bash

pkill nginx

* 배포시 설정(시스템 설정 및 HOST 설정에 따라서 설정파일 조회하도록 처리.)

#./start_nginx.sh

PATH_HEAD=/run

If[ -d /dev/nofacedomain]; then

    PATH_HEAD=/dev

Fi

HOST='hostname'

PATH=$PATH_HEAD/nofacedomain

cd $PATH/nginx/sbin

LD_LIBRARY_PATH=$PATH/lib ./nginx -p $PATH/nginx -c $PATH/nginx/conf/$HOST.nginx.conf
```

## 3.7 Nginx 서비스 확인

```
# netstat -nltp |grep 8200

tcp        0      0 0.0.0.0:8200          0.0.0.0:*            LISTEN     19307/nginx: master

# netstat -nltp |grep 8300

tcp        0      0 0.0.0.0:8300          0.0.0.0:*            LISTEN     19307/nginx: master
```

# 4. Nodejs(Signaling)

## 4.1 Nodejs 설치 파일 경로

```
/sw/nodejs : node-v8.12.0-linux-x64.tar.xz
```

## 4.2 Nodejs 파일 압축해제

```
# tar -xvf node-v8.12.0-linux-x64.tar
```

## 4.3 Nodejs 소스 배포

```
/sw/middle/signaling/

./server.js          # nodejs 메인 파일
```

## 4.4 Nodejs Path 추가

```
# vi /etc/profile  
  
export PATH=$PATH:/sw/middle/node-v8.12.0-linux-x64/bin
```

## 4.5 Nodejs 실행/종료

```
# ./signaling_start.sh  
  
#!/bin/bash  
  
export NODE_PATH=/sw/signaling/node_modules  
  
forever start server.js  
  
# ./signaling_stop.sh  
  
#!/bin/bash  
  
pkill nodejs
```

# 5. S-pass 솔루션

## 5.1 S-pass 소스배포

```
/sw/middle/spass/  
./index.jsp  
./WEB-INF
```

# 6. Turn Proxy JAVA

## 6.1 Turn Proxy 설정 파일 경로

```
web.xml : s-pass 설치경로/WebContents/WEB-INF/
```

## 6.2 Turn Proxy 설정 추가

```

<servlet>

  <servlet-name>TurnProxy</servlet-name>

  <servlet-class>com.shinhan.noface.webrtc.proxy.TurnProxyStartServlet</servlet-class>

  <init-param>
    <param-name>listenPort</param-name>
    <param-value>18100</param-value>          # 서비스 Port.
  </init-param>
  <init-param>
    <param-name>turnserverIP</param-name>
    <param-value>192.168.56.101</param-value>  # Proxy 연결 IP
  </init-param>
  <init-param>
    <param-name>turnserverPORT</param-name>
    <param-value>8100</param-value>          # Proxy 연결 Port
  </init-param>
  <load-on-startup>2</load-on-startup>
</servlet>

```

## 7. Turn Proxy Nginx

### 7.1 Nginx 설정 추가

\*사전 설치방법은 3.Nginx 참조.

[/sw/middle/nginx/conf]# vi nginx.conf

load\_module modules/nginx\_stream\_module.so; # udp 방식 사용시 필요.

worker\_process 1;

events...

stream{

upstream appservers{

server ip:port; # Proxy 연결 IP, Port

}

server{

listen 19100; # 서비스 Port

proxy\_pass appservers;

}

server{

listen 19100 udp; # 서비스 Port

proxy\_pass appservers;