

BaroPAM 가이드(Linux)

목차

목차.....	0
1. BaroPAM 설치.....	1
1.1 BaroPAM 설치 전 준비사항.....	1
1.2 BaroPAM 설치 모듈 다운로드.....	2
1.3 BaroPAM 환경 설정 파일 생성.....	4
1.4 BaroPAM 환경 설정.....	8
2. BaroPAM 적용.....	23
2.1 BaroPAM 적용 프로세스.....	23
2.2 BaroPAM 적용 화면.....	23
2.3 Linux 로그인 방법.....	24
2.4 ssh/sftp 접속 툴.....	25
3. BaroPAM 제거.....	31
3.1 BaroPAM 환경 제거.....	31
4. BaroPAM FAQ.....	32
5. MySQL/MariaDB 설치 및 구성.....	38
5.1 MariaDB 설치.....	38
5.2 MariaDB 구성.....	39
6. About BaroPAM.....	43

1. BaroPAM 설치

1.1 BaroPAM 설치 전 준비사항

PAM 모듈을 사용하기 위해서는 기본적으로 PAM 패키지가 반드시 설치되어 있어야 한다. 설치 확인은 다음의 명령어를 실행하여 확인한다. 만약, 설치되어 있지 않으면 Redhat, CentOS는 "dnf install *pam*" 그외는 "sudo apt-get install pam" 명령어로 설치하면 된다.

```
[root]# rpm -qa | grep pam
pam_smb-1.1.7-7.2.1
pam_passwdqc-1.0.2-1.2.2
pam-0.99.6.2-14.e15_11
pam_krb5-2.2.14-22.e15
pam-devel-0.99.6.2-14.e15_11
pam_ccreds-3-5
pam_smb-1.1.7-7.2.1
pam_pkcs11-0.5.3-26.e15
pam-devel-0.99.6.2-14.e15_11
pam_passwdqc-1.0.2-1.2.2
pam-0.99.6.2-14.e15_11
pam_ccreds-3-5
pam_krb5-2.2.14-22.e15
pam_pkcs11-0.5.3-26.e15
```

정보자산에 접속하여 PAM 모듈을 사용하기 위해서는 신뢰성 있고 안전한 ssh, sftp 서비스를 제공하기 위하여 OpenSSH(Open Secure Shell) 패키지가 반드시 설치되어 있어야 한다. 설치 확인은 다음의 명령어를 실행하여 확인한다. 만약, 설치되어 있지 않으면 Redhat, CentOS는 "dnf install *openssh*"과 "dnf install *openssl*" 그외는 "sudo apt-get install openssl" 명령어로 설치하면 된다.

```
[root]# rpm -qa | grep openssh
openssh-clients-4.3p2-82.e15
openssh-server-4.3p2-82.e15
openssh-4.3p2-82.e15

[root]# rpm -qa | grep openssl
openssl-0.9.8e-40.e15_11
openssl101e-1.0.1e-11.e15
openssl097a-0.9.7a-12.e15_10.1
openssl-devel-0.9.8e-40.e15_11
openssl-perl-0.9.8e-40.e15_11
openssl-devel-0.9.8e-40.e15_11
openssl101e-devel-1.0.1e-11.e15
openssl101e-static-1.0.1e-11.e15
openssl-0.9.8e-40.e15_11
openssl101e-devel-1.0.1e-11.e15
openssl101e-static-1.0.1e-11.e15
openssl101e-perl-1.0.1e-11.e15
openssl097a-0.9.7a-12.e15_10.1
openssl101e-1.0.1e-11.e15
```

```
[root]# ssh -V
OpenSSH_4.3p2, OpenSSL 0.9.8e-fips-rhel5 01 Jul 2008
```

Redhat, CentOS인 경우 "Selinux"는 "Security Enhanced Linux"의 약자로 기본의 리눅스보다 더욱 뛰어난 보안정책을 제공하는데, 너무 뛰어난 나머지 활성화 되어 있을 경우 보안문제로 막혀서 BaroPAM이 안되는 부분이 발생(Failed to open tmp secret file "/usr/baropam/.baro_auth~" [Permission denied])한다. 그래서 웬만하면 대부분이 비활성화(SELINUS=enforcing → disabled)한다.

```
[root] /etc > vi /etc/sysconfig/selinux
# 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 - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=targeted

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

바로 적용은 되지 않으며 재부팅을 해야 적용이 된다.

재부팅을 하지 않고 현재 접속된 터미널에 한해 변경된 내용을 적용하고 싶을 경우 다음의 명령어를 실행하면 된다.

```
[root] /etc > /usr/sbin/setenforce 0
```

PAM 인증 중 환경 설정 정보를 MariaDB에 설정하는 경우는 MariaDB Client를 반드시 설치해야 한다.

```
[root] /etc > dnf -y install mariadb-client → Redhat 계열
[root] /etc > sudo apt -y install mariadb-client → 그외
```

BaroPAM 인증 모듈을 다운로드 및 설치하기 위해서 root 계정으로 접속한 후 모듈을 다운로드 및 설치하기 위한 디렉토리(/usr/baropam)를 다음과 같이 생성한다.

```
[root]# mkdir /usr/baropam
```

BaroPAM 모듈을 다운로드 및 설치하기 위한 디렉토리의 권한(읽기, 쓰기, 실행)을 다음과 같이 부여한다.

```
[root]# chmod -R 777 /usr/baropam
```

1.2 BaroPAM 설치 모듈 다운로드

설치하고 하는 Linux 시스템의 운영체제에 대한 이름 또는 시스템 정보, 커널 정보를 확인하기 위하여 root 계정으로 접속한 후 다음과 같은 명령어를 실행한다.

```
[root] /usr/baropam > uname -a
```

```
Linux baropam 3.10.0-862.el7.x86_64 #1 SMP Fri Apr 20 16:44:24 UTC 2018 x86_64 x86_64 x86_64
GNU/Linux
```

BaroPAM 인증 모듈은 root 계정으로 접속한 후 모듈을 다운로드 및 설치하기 위한 디렉토리(/usr/baropam)로 이동하여 모듈을 다운로드 하는 방법은 다음과 같다.

```
[root] /usr/baropam > wget http://nuriapp.com/download/libpam_baro_auth-x.x.tar
```

BaroPAM 인증 모듈의 다운로드가 완료되면 tar 파일의 압축을 해제하는 방법은 다음과 같다.

```
[root] /usr/baropam > tar -xvf libpam_baro_auth-x.x.tar
```

BaroPAM 인증 모듈의 압축을 해제하면 baropam 디렉토리에 다음과 같은 BaroPAM 관련 모듈이 생성된다.

```
[root] /usr/baropam > ls -al
합계 180
drwxrwxrwx 7 root root 4096 8월 23 09:59 .
drwxr-xr-x 17 root root 4096 2월 10 2017 ..
-r--r--r-- 1 root root 8 3월 24 2021 .baro_acl
-r--r--r-- 1 root root 305 7월 2 14:41 .baro_auth
-r--r--r-- 1 root root 290 6월 30 12:55 .baro_curl
-r--r--r-- 1 root root 287 2월 28 12:19 .baro_sql
-rwxr-xr-x 1 root root 69149 4월 6 19:12 baro_auth
-rwxr-xr-x 1 root root 65072 6월 29 16:36 baro_curl
-rwxr-xr-x 1 root root 57074 2월 28 12:18 baro_sql
drwxr-xr-x 2 root root 4096 7월 20 2021 jilee
-rwxr-xr-x 1 root root 152649 6월 9 08:19 pam_baro_auth.so
-rwxr-xr-x 1 root root 116158 6월 30 12:54 pam_baro_curl.so
-rwxr-xr-x 1 root root 170863 2월 28 12:18 pam_baro_sql.so
-rw-r--r-- 1 root root 221 6월 27 15:59 setauth.sh
-rw-r--r-- 1 root root 150 6월 29 16:29 setcurl.sh
-rw-r--r-- 1 root root 180 2월 28 12:19 setsql.sh
```

생성한 BaroPAM 인증 모듈이 시스템에 맞는 모듈인지 다음과 같은 명령어를 실행하여 확인한다.

```
[root] /usr/baropam > file pam_baro_auth.so
pam_baro_auth.so: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked,
BuildID[sha1]=d2d7b4ffe8b1a25f6a11685cb7ad4ec9787163b5, not stripped
```

```
[root] /usr/baropam > ldd pam_baro_auth.so
linux-vdso.so.1 => (0x00007ffe7f503000)
libpam.so.0 => /usr/lib64/libpam.so.0 (0x00007f23a3318000)
libssl.so.10 => /usr/lib64/libssl.so.10 (0x00007f23a30a6000)
libcrypto.so.10 => /usr/lib64/libcrypto.so.10 (0x00007f23a2c45000)
libdl.so.2 => /usr/lib64/libdl.so.2 (0x00007f23a2a41000)
libz.so.1 => /usr/lib64/libz.so.1 (0x00007f23a282b000)
libc.so.6 => /usr/lib64/libc.so.6 (0x00007f23a245e000)
libaudit.so.1 => /usr/lib64/libaudit.so.1 (0x00007f23a2235000)
libgssapi_krb5.so.2 => /usr/lib64/libgssapi_krb5.so.2 (0x00007f23a1fe8000)
libkrb5.so.3 => /usr/lib64/libkrb5.so.3 (0x00007f23a1d00000)
libcom_err.so.2 => /usr/lib64/libcom_err.so.2 (0x00007f23a1afc000)
libk5crypto.so.3 => /usr/lib64/libk5crypto.so.3 (0x00007f23a18c9000)
/lib64/ld-linux-x86-64.so.2 (0x00007f23a372f000)
```

```
libcap-ng.so.0 => /usr/lib64/libcap-ng.so.0 (0x00007f23a16c3000)
libkrb5support.so.0 => /usr/lib64/libkrb5support.so.0 (0x00007f23a14b5000)
libkeyutils.so.1 => /usr/lib64/libkeyutils.so.1 (0x00007f23a12b1000)
libresolv.so.2 => /usr/lib64/libresolv.so.2 (0x00007f23a1098000)
libpthread.so.0 => /usr/lib64/libpthread.so.0 (0x00007f23a0e7c000)
libselinux.so.1 => /usr/lib64/libselinux.so.1 (0x00007f23a0c55000)
libpcre.so.1 => /usr/lib64/libpcre.so.1 (0x00007f23a09f3000)
```

1.3 BaroPAM 환경 설정 파일 생성

1) PAM 인증(.baro_auth): 환경 설정 정보를 File에 설정

BaroPAM 환경 설정 파일은 baro_auth 프로그램을 실행하여 반드시 생성하는데, BaroPAM 인증 모듈의 디렉토리인 /usr/baropam 밑에 위치 하도록 한다.

형식)

```
baro_auth -r rate_limit -R rate_time -t cycle_time -k key_method -e encrypt_flag -A acl_type -a
acl_filename -S secure_key -s filename
```

BaroPAM 환경설정 파일의 설정 옵션에 대한 내용은 다음과 같다.

옵션	설명	설정값	비고
-r	일회용 인증키의 제한횟수(1~10)	3	
-R	일회용 인증키의 제한시간(초, 15~600초)	30	
-t	일회용 인증키의 인증주기(초, 3~60초)	30	
-k	일회용 인증키의 인증방식(app1, app256, app384, app512).	app512	
-e	환경설정 파일의 암호화 여부(yes or no)	no	
-A	2차 인증에서 허용(allow) 또는 제외(deny)할지 선택	deny	
-a	2차 인증에서 허용(allow) 또는 제외(deny)할 계정에 대한 ACL 파일명(파일 접근권한은 444)	/usr/baropam/.baro_acl	
-S	반드시 벤더에서 제공하는 Secure key(라이선스 키)	j1q1chbVqdpj7b4PzBpM2DileBvmHFV/	
-s	BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명	/usr/baropam/.baro_auth	

주의) -s 옵션의 filename는 BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명(파일 접근권한은 444)이다.

사용 예)

```
[root] /usr/baropam > ./baro_auth -r 3 -R 30 -t 30 -k app512 -e no -A deny -a
/usr/baropam/.baro_acl -S j1q1chbVqdpj7b4PzBpM2DileBvmHFV/ -s /usr/baropam/.baro_auth
```

만약, 계정마다 BaroPAM 환경 설정파일을 각각 설정하는 경우 해당 계정으로 접속하여 작업을 진행한다. (Not root)

```
[root] /usr/baropam > ./baro_auth -r 3 -R 30 -t 30 -k app512 -e no -A deny -a ~/.baro_acl -S
j1q1chbVqdpj7b4PzBpM2DileBvmHFV/ -s ~/.baro_auth
```

1) Your emergency one-time authentication keys are :

응급 일회용 인증키는 **일회용 인증키** 생성기인 BaroPAM 앱을 사용할 수 없을 때 분실한 경우를 대비하여 SSH 서버에 다시 액세스하는데 사용할 수 있는 접속이 가능한 Super 인증키 이므로 어딘가에 적어 두는 것이 좋다.

- 2) 다음에 나오는 물음에 대해서는 모두 "y"를 입력한다.
 "/usr/baropam/.baro_auth" 파일을 업데이트하시겠습니까 (y/n) **y**
 중간자(man-in-the-middle) 공격을 예방할 것인가 (y/n) **y**

BaroPAM 환경 설정 파일인 .baro_auth에 설정한 내용은 다음과 같다.

```
[root] /usr/baropam > cat .baro_auth
" AUTH_KEY
" RATE_LIMIT 3 30
" KEY_METHOD app512
" CYCLE_TIME 30
" SECURE_KEY j1q1chbVqdpj7b4PzBpM2DileBvmHFV/
" ACL_NAME /usr/baropam/.baro_acl
" ACL_TYPE deny
" DISALLOW_REUSE
33458936
19035576
15364353
54649370
84342192
```

BaroPAM 환경설정 파일인 .baro_auth의 설정 항목에 대한 내용은 다음과 같다.

항목	설명	설정값	비고
AUTH_KEY	인증 구분자(고정)		
RATE_LIMIT	일회용 인증키 의 제한횟수(1~10), 제한시간(초, 15~600초)	3 30	
KEY_METHOD	일회용 인증키 의 인증방식(app1, app256, app384, app512)	app512	
CYCLE_TIME	일회용 인증키 의 인증주기(초, 3~60초)	30	
SECURE_KEY	반드시 벤더에서 제공하는 Secure key(라이선스 키)	j1q1chbVqdpj7b4PzBpM2DileBvmHFV/	
ACL_TYPE	2차 인증 에서 허용(allow) 또는 제외(deny) 구분	deny	
ACL_NAME	2차 인증 에서 허용 또는 제외할 계정에 대한 ACL Filename(파일 접근권한은 444)	/usr/baropam/.baro_acl	
DISALLOW_REUSE or ALLOW_REUSE	중간자(man-in-the-middle) 공격을 예방할 경우는 "DISALLOW_REUSE"을 설정한 경우 일회용 인증키 의 인증주기 동안은 다른 사용자가 로그인 할 수 없으며, 만약 허용할 경우는 "ALLOW_REUSE"을 설정한다.	DISALLOW_REUSE	

2) PAM 인증(.baro_sql): 환경 설정 정보를 MariaDB에 설정

BaroPAM 환경 설정 정보가 존재하는 Mariadb와 연동하기 위한 접속 정보는 baro_sql 프로그램을 실행하여 반드시 생성하는데, BaroPAM 인증 모듈의 디렉토리인 /usr/baropam 밑에 위치하도록 한다.

형식)

```
baro_sql -H hostname -u username -p password -d dbname -P portno -e encrypt_flag -s filename
```

BaroPAM 환경설정 파일의 설정 옵션에 대한 내용은 다음과 같다.

옵션	설명	설정값	비고
-H	MariaDB 서버의 호스트 이름 또는 IP 주소	nur it.co.kr	
-u	MariaDB 사용자 이름	nur it	
-p	MariaDB 사용자의 비밀번호	baropam	
-d	연결할 MariaDB 이름	baropamdb	
-P	MariaDB 서버의 포트 번호	3308	
-e	환경설정 파일의 암호화 여부(yes or no)	no	
-s	BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명	/usr/baropam/.baro_sql	

주의) -s 옵션의 filename는 BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명(파일 접근권한은 444)이다.

사용 예)

```
[root] /usr/baropam > ./baro_sql -H nurit.co.kr -e no -u nurit -p baropams -d baropamdb -P 3306 -s /usr/baropam/.baro_sql
```

- 다음에 나오는 물음에 대해서는 모두 "y"를 입력한다.
"/usr/baropam/.baro_sql" 파일을 업데이트하시겠습니까 (y/n) y

BaroPAM 환경 설정 파일인 .baro_sql에 설정한 내용은 다음과 같다.

```
[root] /usr/baropam > cat .baro_sql
" AUTH_KEY
" HOSTNAME nurit.co.kr
" USERNAME nurit
" PASSWORD baropams
" DBNAME baropamdb
" PORTNO 3306
" RATE_LIMIT 3 30
" KEY_METHOD app512
" CYCLE_TIME 30
" SECURE_KEY j!qlclHbVqdpj7b4PzBpM2Di!eBvmHFV/
" ACL_TYPE deny
" MIDDLE_TYPE DISALLOW_REUSE
" MIDDLE_TIME 58014762
" ENV_TYPE share
```

BaroPAM 환경설정 파일인 .baro_sql의 설정 항목에 대한 내용은 다음과 같다.

항목	설명	설정값	비고
AUTH_KEY	인증 구분자(고정)		
HOSTNAME	MariaDB 서버의 호스트 이름 또는 IP 주소	nur it.co.kr	
USERNAME	MariaDB 사용자 이름	nur it	
PASSWORD	MariaDB 사용자의 비밀번호	baropam	
DBNAME	연결할 MariaDB 이름	baropamdb	
PORTNO	MariaDB 서버의 포트 번호	3308	
그외	나머지는 내부용으로 사용함.		

3) curl 인증(.baro_curl)

curl 의 명칭은 "client URL" 을 대표하는 것으로 1997년에 처음 출시되었다. 즉 클라이언트가 스크립트로써 서버에 데이터를 요청하는 것으로 BaroPAM은 curl로 http/https 되어 있는 인증 사이트를 호출하여 인증을 요청한다.

BaroPAM 환경 설정 파일은 baro_curl 프로그램을 실행하여 반드시 생성하는데, BaroPAM 인증 모듈의 디렉토리인 /usr/baropam 밑에 위치 하도록 한다.

형식)

```
baro_curl -r rate_limit -R rate_time -t cycle_time -k key_method -e encrypt_flag -H hostname -u auth_url -s filename
```

BaroPAM 환경설정 파일의 설정 옵션에 대한 내용은 다음과 같다.

옵션	설명	설정값	비고
-r	일회용 인증키의 제한횟수(1~10)	3	
-R	일회용 인증키의 제한시간(초, 15~600초)	30	
-t	일회용 인증키의 인증주기(초, 3~60초)	30	
-k	일회용 인증키의 인증방식(app1, app256, app384, app512).	app512	
-e	환경설정 파일의 암호화 여부(yes or no)	no	
-H	서버의 호스트명(uname -n)	nur it.co.kr	
-u	호출할 URL로 호스트명(hostname), 사용자 계정(username), 인증주기(cycle_time), 일회용 인증키(auth_key) 등의 파라미터가 포함되어 호출	http://1.23.456.789/baropam/web/result_curl.jsp	
-s	BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명	/usr/baropam/.baro_curl	

주의) -s 옵션의 filename은 BaroPAM 환경설정 파일을 생성할 디렉토리를 포함한 파일명(파일 접근권한은 444)이며, 설정한 서버의 호스트명(hostname)이 맞지 않는 경우 BaroPAM이 정상적으로 작동되지 않을 수 있으니, 호스트명(hostname)가 변경되는 경우 반드시 환경 설정의 해당 항목에 반영해야 한다.

사용 예)

```
[root] /usr/baropam > ./baro_curl -r 3 -R 30 -t 30 -k app512 -e no -H nur it.co.kr -u http://1.23.456.789/baropam/web/result_curl.jsp -s /usr/baropam/.baro_curl
```

- 다음에 나오는 물음에 대해서는 모두 "y"를 입력한다.
"/usr/baropam/.baro_curl" 파일을 업데이트하시겠습니까 (y/n) y
중간자(man-in-the-middle) 공격을 예방할 것인가 (y/n) y

BaroPAM 환경 설정 파일인 .baro_curl에 설정한 내용은 다음과 같다.

```
[root] /usr/baropam > cat .baro_curl
" AUTH_KEY
" RATE_LIMIT 3 30
" AUTH_URL http://1.23.456.789/baropam/web/result_curl.jsp
" KEY_METHOD app512
" CYCLE_TIME 30
" HOSTNAME baropam
" DISALLOW_REUSE
```

BaroPAM 환경설정 파일인 .baro_curl의 설정 항목에 대한 내용은 다음과 같다.

항목	설명	설정값	비고
AUTH_KEY	인증 구분자(고정)		
RATE_LIMIT	일회용 인증키 의 제한횟수(1~10), 제한시간(초, 15~600초)	3 30	
AUTH_URL	호출할 URL로 호스트명(hostname), 사용자 계정(username), 인증주기(cycle_time), 일회용 인증키(auth_key) 등의 파라미터가 포함되어 호출	http://1.23.456.789/baropam/web/result_curl.jsp	
KEY_METHOD	일회용 인증키 의 인증방식(app1, app256, app384, app512)	app512	
CYCLE_TIME	일회용 인증키 의 인증주기(초, 3~60초)	30	
HOSTNAME	서버의 호스트명(uname -n)	nurit.co.kr	
DISALLOW_REUSE or ALLOW_REUSE	중간자(man-in-the-middle) 공격을 예방할 경우는 "DISALLOW_REUSE"을 설정한 경우 일회용 인증키 의 인증주기 동안은 다른 사용자가 로그인 할 수 없으며, 만약 허용할 경우는 "ALLOW_REUSE"을 설정한다.	DISALLOW_REUSE	

1.4 BaroPAM 환경 설정

1) PAM 인증: 환경 설정 정보를 File에 설정

① 추가 인증(로그인-ID, 비밀번호 이외의 추가 인증으로 일회용 인증키 적용)

BaroPAM 모듈을 설정하기 위해서 sshd, su, sudo 파일 등에 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

참고로 secret 파라미터는 BaroPAM 환경설정 파일명, encrypt 파라미터는 BaroPAM 환경설정 파일의 암호화 플래그(yes or no)를 설정한다.

만약, 계정마다 BaroPAM 환경 설정파일을 각각 설정하는 경우 BaroPAM 모듈을 설정하기 위해서 sshd 파일에 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=${HOME}/.baro_auth encrypt=no
```

계정마다 BaroPAM 환경 설정파일을 각각 설정하지 않고 특정 디렉토리에 계정별로 BaroPAM 환경 설정파일을 다르게 설정하고자 하는 경우 BaroPAM 모듈을 설정하기 위해서 sshd 파일에 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/auth/.$[USER]_auth
encrypt=no
```

* "nullok"는 호출된 PAM 모듈이 null 값의 암호를 입력하는 것을 허용한다는 의미다. 단, Redhat 9.x 이상에서는 /etc/pam.d/sshd 설정에서 "nullok" 옵션을 지원하지 않는다.

```
[root] /usr/baropam > vi /etc/pam.d/su
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

/etc/pam.d/su 파일에 BaroPAM 모듈을 최 상단에 추가하면 "su" 명령어로 일반계정이 "root"로 권한 상승을 시도하는 경우에도 **2차 인증(추가 인증)** 입력을 적용할 수 있어서 보안이 한층 더 향상된다.

```
$ su - root
Verification code:
```

Desktop Linux 인 경우 GUI 로그인 화면에서 BaroPAM을 사용하고자 하는 경우 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

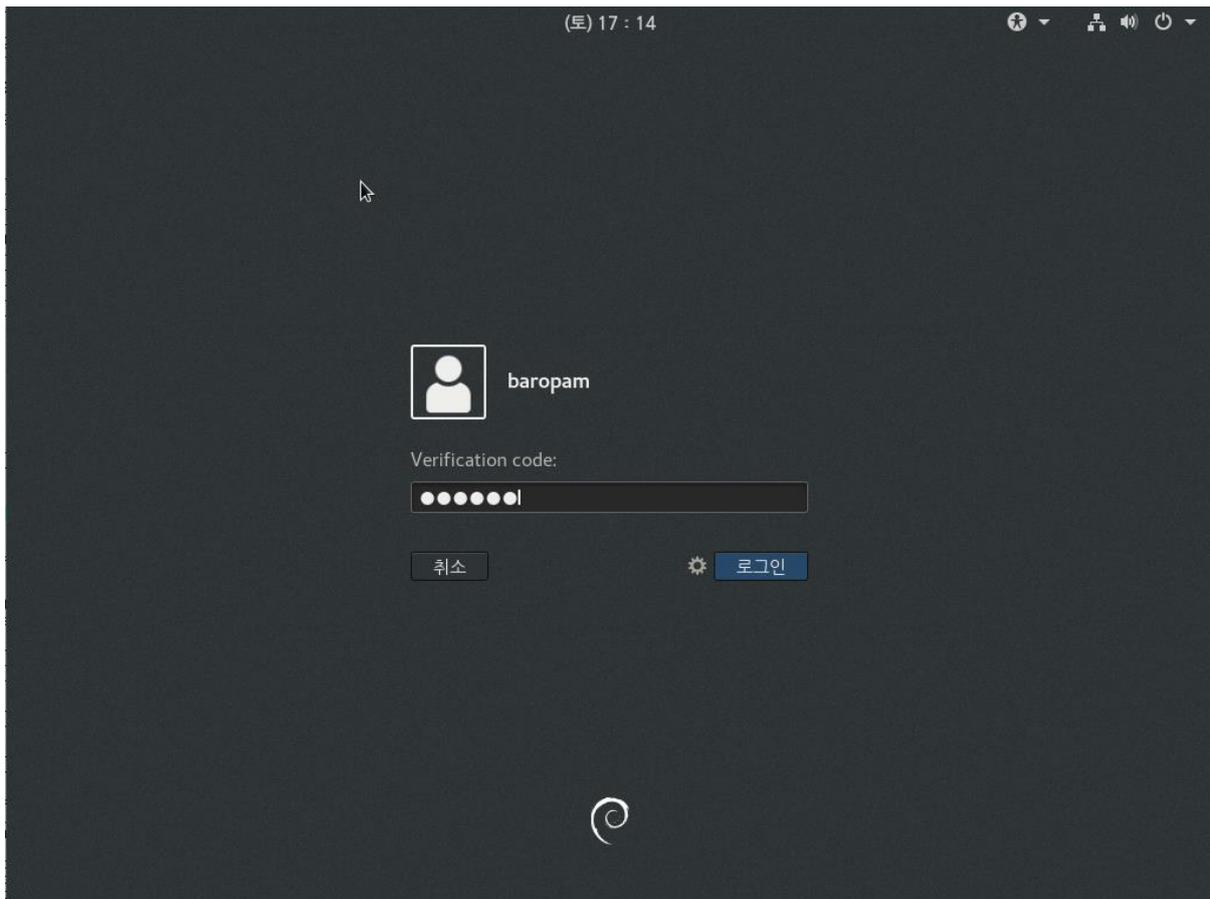
예) Debian, Ubuntu, SUSE, fedora Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/gdm-password or gdm-autologin
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

gdm-password, gdm-autologin 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 gdm-password의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart gdm-password
```

그러면 다음과 같이 로그인 화면에 BaroPAM의 **일회용 인증키**인 "Verification code:"를 입력하는 화면이 나타난다.



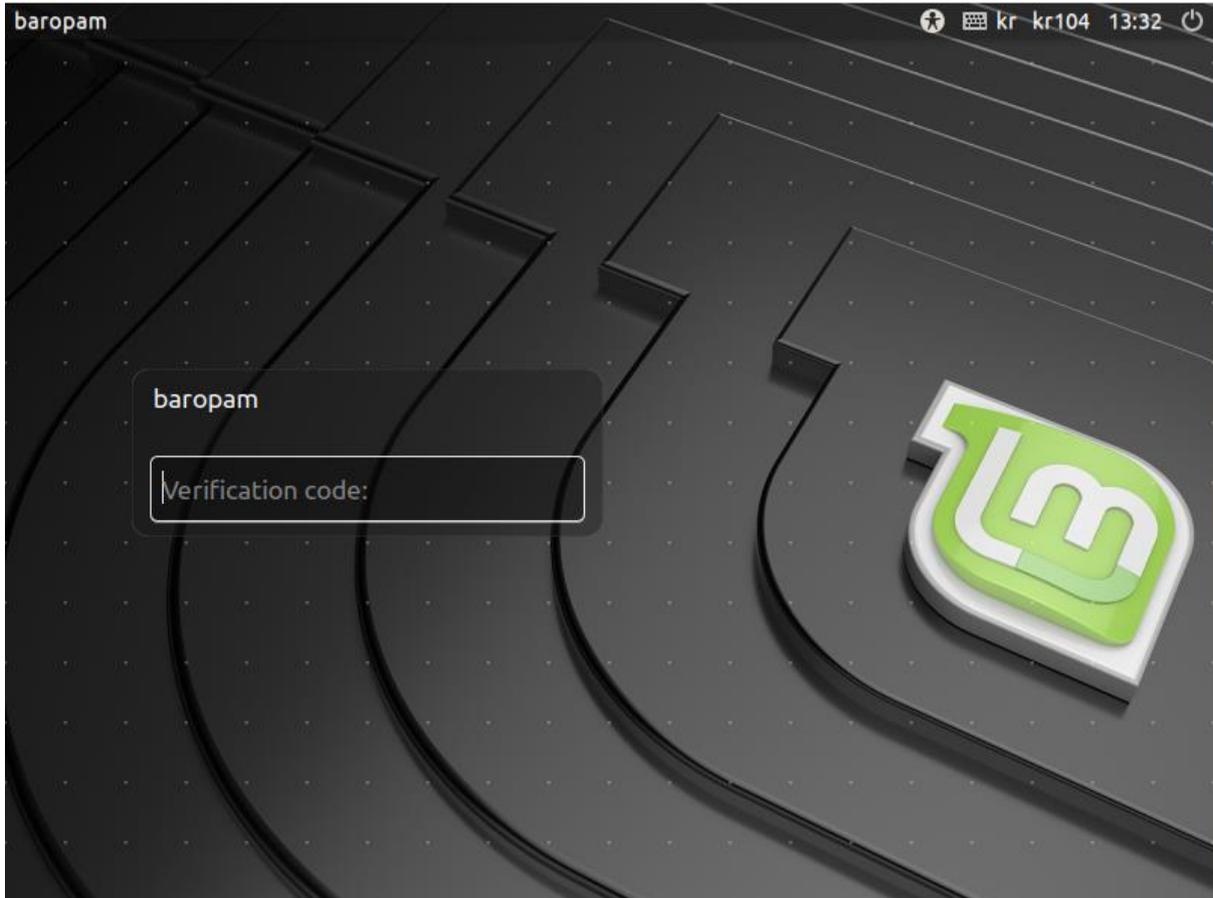
예) 하모니카OS, 구름OS, Mint Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/lightdm or logtmdm-autologin
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

`lightdm`, `lightdm-autologin` 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 `lightdm`의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart lightdm
```

그러면 다음과 같이 로그인 화면에 BaroPAM의 일회용 인증키인 "Verification code:"를 입력하는 화면이 나타난다.



참고) 개방형OS 같은 Desktop Linux인 경우는 "`passwd -p username`" 명령어로 비밀번호를 제거하면 "Verification code:"의 입력하는 화면에 **일회용 인증키**만 입력하면 비밀번호는 묻지 않는다.

예) xrdp를 이용한 원격 데스크톱 접속을 하는 경우

```
[root] /usr/baropam > vi /etc/pam.d/xrdp-sesman
#%PAM-1.0
auth    required    /usr/baropam/pam_baro_auth.so forward_pass secret=/usr/baropam/.baro_auth
encrypt=no
```

`forward_pass`를 이용하여 암호 입력창(Password)에 **일회용 인증키**를 입력하면 된다. 예를 들어, **일회용 인증키**가 "123456" 이라면 "123456"만 입력하면 된다.

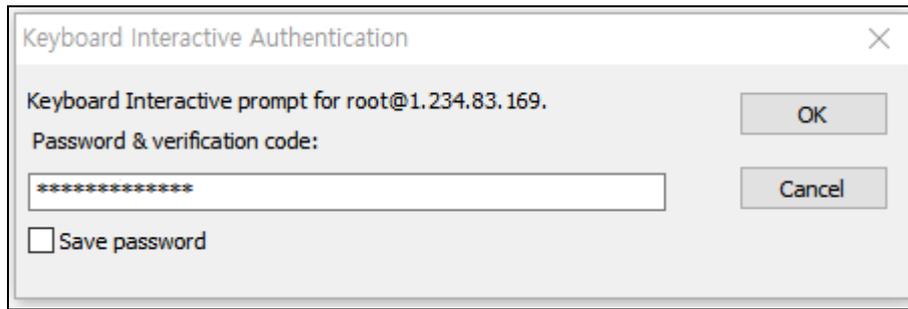
② 비밀번호 대체(비밀번호를 대신 일회용 인증키로 대체)

filezilla처럼 "Interactive process"가 불가능한 프로그램들을 위해서는 PAM에서 `forward_pass` 옵션을 사용하여 암호 입력 시에 **일회용 인증키**를 입력하도록 하는 수 밖에 없다. 이 경우, openssh client나 Windows의 RDP(Remote Desktop Protocol), Radius, filezilla 등 모두 이렇게 입력을 하는 수 밖에 없다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth    required    /usr/baropam/pam_baro_auth.so forward_pass secret=/usr/baropam/.baro_auth
encrypt=no
```

`forward_pass`를 이용하여 암호 입력창(Password & verification code:)에 **일회용 인증키**를 입력하면 된다.

예를 들어, **일회용 인증키**가 "123456" 이라면 "123456"만 입력하면 된다.



참고) 비밀번호를 **일회용 인증키**로 대체하는 경우 해당 계정의 비밀번호는 "passwd *username*" 명령어로 미리 로그인-ID와 동일하게 설정해야 한다.

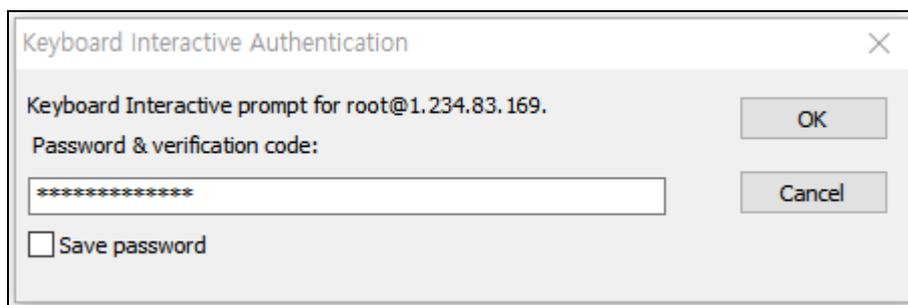
개방형OS 같은 Desktop Linux인 경우는 "passwd -p *username*" 명령어로 비밀번호를 제거하면 "Password & Verification code:"의 입력하는 화면에 **일회용 인증키**만 입력하면 비밀번호는 묻지 않는다.

③ 새로운 비밀번호(비밀번호와 일회용 인증키 결합하여 비밀번호를 일회용 인증키 생성주기별로 새로운 일회용 비밀번호를 생성하여 적용)

filezilla처럼 "Interactive process"가 불가능한 프로그램들을 위해서는 PAM에서 **forward_pass** 옵션을 사용하여 암호 입력 시에 암호와 **일회용 인증키**를 같이 입력하도록 하는 수 밖에 없다. 이 경우, openssh client나 Windows의 RDP(Remote Desktop Protocol), Radius, filezilla 등 모두 이렇게 입력을 하는 수 밖에 없다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_auth.so forward_pass secret=/usr/baropam/.baro_auth
encrypt=no
```

forward_pass를 이용하여 암호 입력창(Password & verification code:)에 암호와 같이 **일회용 인증키**를 입력할 경우, 암호를 먼저 입력하고 공백 없이 이어서 **일회용 인증키**를 입력하면 된다. 예를 들어, 암호가 "baropam" 이고 **일회용 인증키**가 "123456" 이라면 "baropam123456"으로 입력하면 된다.



forward_pass를 이용하면 인증을 필요로 하는 대부분의 서비스에 **2-factor 인증**을 가능하게 할 수 있다.

2) PAM 인증: 환경 설정 정보를 MariaDB에 설정

① 추가 인증(로그인-ID, 비밀번호 이외의 추가 인증으로 일회용 인증키 적용)

BaroPAM 모듈을 설정하기 위해서 sshd, su, sudo 파일 등에 설정하는 방법은 다음과 같이 최 상단에 입력

해 준다.

```
[root] /usr/baropam > vi /etc/pam.d/ssh
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_sql.so nullok secret=/usr/baropam/.baro_sql
encrypt=no auth=sshd
```

참고로 **secret** 파라미터는 BaroPAM 환경설정 파일명, **encrypt** 파라미터는 BaroPAM 환경설정 파일의 암호화 플래그(yes or no), **auth** 파라미터는 BaroPAM를 사용하여 인증하는 곳인 **sshd**, **su**, **sudo**, **login**, **radiusd**, **gdm-password**, **lightdm**, **xrdp-sesman** 등을 설정한다.

* "nullok"는 호출된 PAM 모듈이 null 값의 암호를 입력하는 것을 허용한다는 의미다. 단, Redhat 9.x 이상에서는 /etc/pam.d/ssh 설정에서 "nullok" 옵션을 지원하지 않는다.

```
[root] /usr/baropam > vi /etc/pam.d/su
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_sql.so nullok secret=/usr/baropam/.baro_sql
encrypt=no auth=su
```

/etc/pam.d/su 파일에 BaroPAM 모듈을 최 상단에 추가하면 "su" 명령어로 일반계정이 "root"로 권한 상승을 시도하는 경우에도 2차 인증(추가 인증) 입력을 적용할 수 있어서 보안이 한층 더 향상된다.

```
$ su - root
Verification code:
```

Desktop Linux 인 경우 GUI 로그인 화면에서 BaroPAM을 사용하고자 하는 경우 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

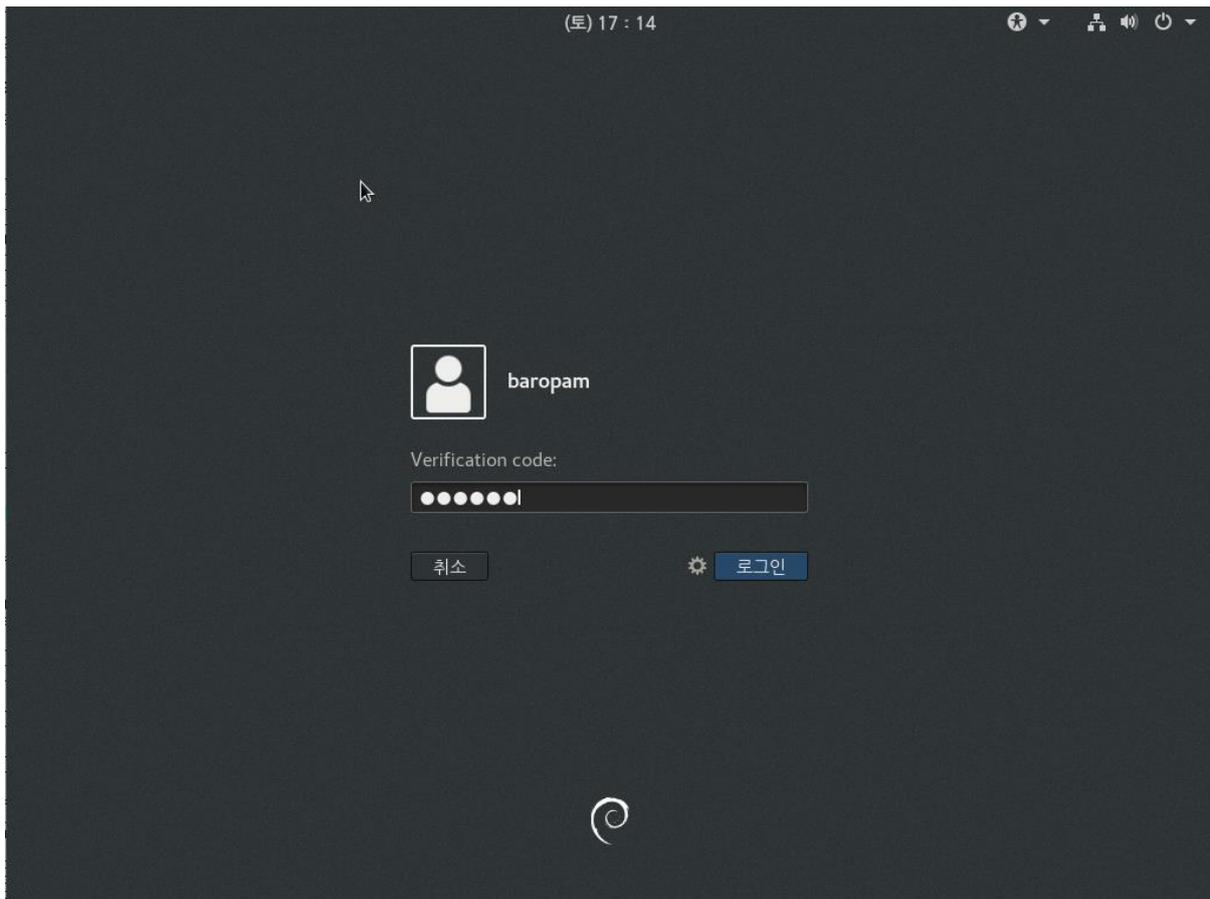
예) Debian, Ubuntu, SUSE, fedora Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/gdm-password or gdm-autologin
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_sql.so nullok secret=/usr/baropam/.baro_sql
encrypt=no auth=gdm-password
```

gdm-password, **gdm-autologin** 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 **gdm-password**의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart gdm-password
```

그러면 다음과 같이 로그인 화면에 BaroPAM의 일회용 인증키인 "Verification code:"를 입력하는 화면이 나타난다.



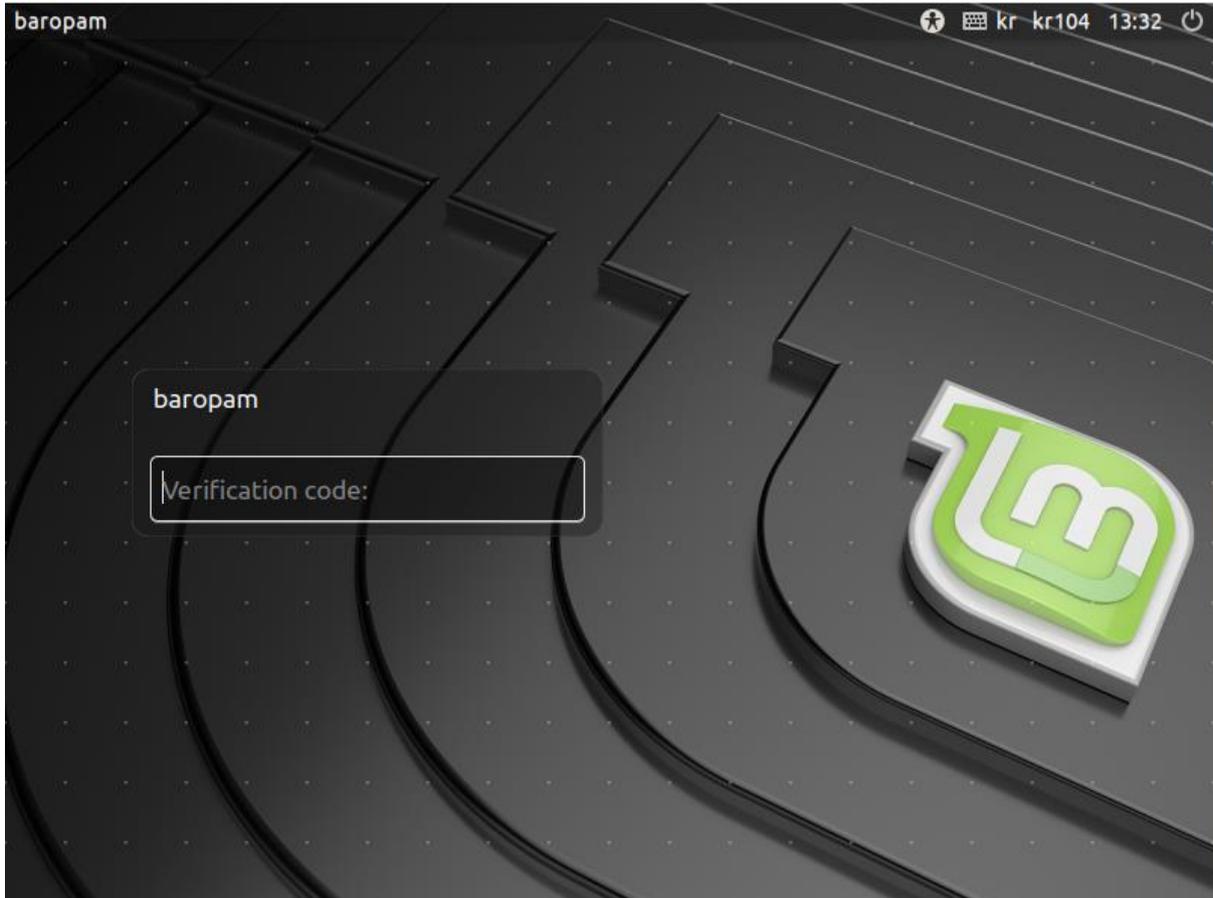
예) 하모니카OS, 구름OS, Mint Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/lightdm or logtldm-autologin
#%PAM-1.0
auth          required          /usr/baropam/pam_baro_sql.so nullok secret=/usr/baropam/.baro_sql
encrypt=no auth=lightdm
```

`lightdm`, `lightdm-autologin` 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 `lightdm`의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart lightdm
```

그러면 다음과 같이 로그인 화면에 BaroPAM의 일회용 인증키인 "Verification code:"를 입력하는 화면이 나타난다.



참고) 개방형OS 같은 Desktop Linux인 경우는 "`passwd -p username`" 명령어로 비밀번호를 제거하면 "Verification code:"의 입력하는 화면에 **일회용 인증키**만 입력하면 비밀번호는 묻지 않는다.

예) xrdp를 이용한 원격 데스크톱 접속을 하는 경우

```
[root] /usr/baropam > vi /etc/pam.d/xrdp-sesman
#%PAM-1.0
auth    required    /usr/baropam/pam_baro_sql.so forward_pass secret=/usr/baropam/.baro_sql
encrypt=no auth=xrdp-sesman
```

`forward_pass`를 이용하여 암호 입력창(Password)에 **일회용 인증키**를 입력하면 된다. 예를 들어, **일회용 인증키**가 "123456" 이라면 "123456"만 입력하면 된다.

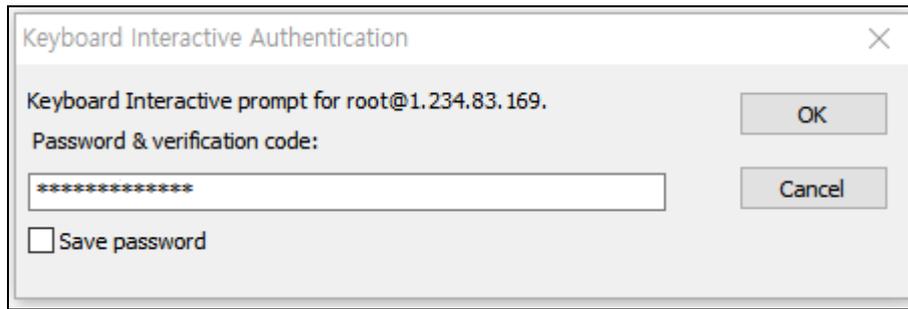
② 비밀번호 대체(비밀번호를 대신 일회용 인증키로 대체)

filezilla처럼 "Interactive process"가 불가능한 프로그램들을 위해서는 PAM에서 `forward_pass` 옵션을 사용하여 암호 입력 시에 **일회용 인증키**를 입력하도록 하는 수 밖에 없다. 이 경우, openssh client나 Windows의 RDP(Remote Desktop Protocol), Radius, filezilla 등 모두 이렇게 입력을 하는 수 밖에 없다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth    required    /usr/baropam/pam_baro_sql.so forward_pass secret=/usr/baropam/.baro_sql
encrypt=no auth=sshd
```

`forward_pass`를 이용하여 암호 입력창(Password & verification code:)에 **일회용 인증키**를 입력하면 된다.

예를 들어, **일회용 인증키**가 "123456" 이라면 "123456"만 입력하면 된다.



참고) 비밀번호를 **일회용 인증키**로 대체하는 경우 해당 계정의 비밀번호는 "`passwd username`" 명령어로 미리 로그인-ID와 동일하게 설정해야 한다.

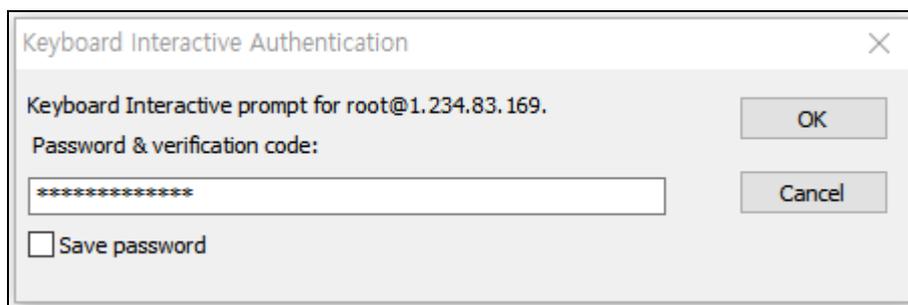
개방형OS 같은 Desktop Linux인 경우는 "`passwd -p username`" 명령어로 비밀번호를 제거하면 "Password & Verification code:"의 입력하는 화면에 **일회용 인증키**만 입력하면 비밀번호는 묻지 않는다.

③ 새로운 비밀번호(비밀번호와 일회용 인증키 결합하여 비밀번호를 일회용 인증키 생성주기별로 새로운 일회용 비밀번호를 생성하여 적용)

filezilla처럼 "Interactive process"가 불가능한 프로그램들을 위해서는 PAM에서 `forward_pass` 옵션을 사용하여 암호 입력 시에 암호와 **일회용 인증키**를 같이 입력하도록 하는 수 밖에 없다. 이 경우, openssh client나 Windows의 RDP(Remote Desktop Protocol), Radius, filezilla 등 모두 이렇게 입력을 하는 수 밖에 없다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth    required    /usr/baropam/pam_baro_sql.so forward_pass secret=/usr/baropam/.baro_sql
encrypt=no auth=sshd
```

`forward_pass`를 이용하여 암호 입력창(Password & verification code:)에 암호와 같이 **일회용 인증키**를 입력할 경우, 암호를 먼저 입력하고 공백 없이 이어서 **일회용 인증키**를 입력하면 된다. 예를 들어, 암호가 "baropam" 이고 **일회용 인증키**가 "123456" 이라면 "baropam123456"으로 입력하면 된다.



`forward_pass`를 이용하면 인증을 필요로 하는 대부분의 서비스에 **2-factor 인증**을 가능하게 할 수 있다.

3) dURL 인증

BaroPAM 모듈을 설정하기 위해서 sshd, su, sudo 파일 등에 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so nullok secret=/usr/baropam/.baro_curl
encrypt=no
```

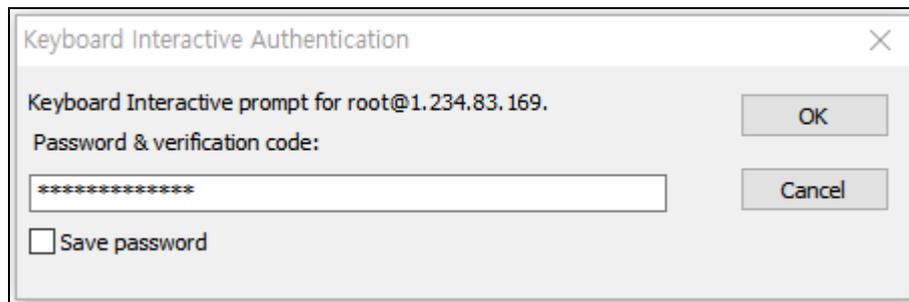
참고로 **secret** 파라미터는 **BaroPAM** 환경설정 파일명, **encrypt** 파라미터는 **BaroPAM** 환경설정 파일의 암호화 플래그(**yes** or **no**)를 설정한다.

* "**nullok**"는 호출된 PAM 모듈이 null 값의 암호를 입력하는 것을 허용한다는 의미다. 단, Redhat 9.x 이상에서는 **/etc/pam.d/sshd** 설정에서 "**nullok**" 옵션을 지원하지 않는다.

filezilla처럼 "**Interactive process**"가 불가능한 프로그램들을 위해서는 PAM에서 **forward_pass** 옵션을 사용하여 암호 입력 시에 암호와 **일회용 인증키**를 같이 입력하도록 하는 수 밖에 없다. 이 경우, openssh client나 Windows의 RDP(Remote Desktop Protocol), Radius, filezilla 등 모두 이렇게 입력을 하는 수 밖에 없다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so forward_pass secret=/usr/baropam/.baro_curl
encrypt=no
```

forward_pass를 이용하여 암호 입력창(Passwd & verification code:)에 암호와 같이 **일회용 인증키**를 입력할 경우, 암호를 먼저 입력하고 공백 없이 이어서 **일회용 인증키**를 입력하면 된다. 예를 들어, 암호가 "baropam" 이고 **일회용 인증키**가 "123456" 이라면 "baropam123456"으로 입력하면 된다.



forward_pass를 이용하면 인증을 필요로 하는 대부분의 서비스에 **2-factor 인증**을 가능하게 할 수 있다.

```
[root] /usr/baropam > vi /etc/pam.d/su
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so forward_pass secret=/usr/baropam/.baro_curl
encrypt=no
```

/etc/pam.d/su 파일에 **BaroPAM** 모듈을 최 상단에 추가하면 "su" 명령어로 일반계정이 "root"로 권한 상승을 시도하는 경우에도 **2차 인증(추가 인증)** 입력을 적용할 수 있어서 보안이 한층 더 향상된다.

```
$ su - root
Password & verification code:
```

Desktop Linux 인 경우 GUI 로그인 화면에서 **BaroPAM**을 사용하고자 하는 경우 설정하는 방법은 다음과 같이 최 상단에 입력해 준다.

예) Debian, Ubuntu, SUSE, fedora Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/gdm-password or gdm-autologin
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so forward_pass secret=/usr/baropam/.baro_curl
encrypt=no
```

gdm-password, **gdm-autologin** 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 **gdm-password**의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart gdm-password
```

예) 하모니카OS, 구름OS, Mint Linux인 경우

```
[root] /usr/baropam > vi /etc/pam.d/lightdm or logtmdm-autologin
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so forward_pass secret=/usr/baropam/.baro_curl
encrypt=no
```

lightdm, **lightdm-autologin** 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 **lightdm**의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > systemctl restart lightdm
```

예) xrdp를 이용한 원격 데스크톱 접속을 하는 경우

```
[root] /usr/baropam > vi /etc/pam.d/xrdp-sesman
#%PAM-1.0
auth      required      /usr/baropam/pam_baro_curl.so forward_pass secret=/usr/baropam/.baro_curl
encrypt=no
```

forward_pass를 이용하여 암호 입력창(Password)에 **일회용 인증키**를 입력하면 된다. 예를 들어, **일회용 인증키**가 "123456" 이라면 "123456"만 입력하면 된다.

3) sshd 데몬의 환경 설정

sshd 데몬 설정을 위한 설정 파일인 **"/etc/ssh/sshd_config"** 파일의 내용 중 다음과 같은 인자는 변경이 필요하다.

인자	기존	변경	비고
PasswordAuthentication	yes	No	
Redhat 9.x 이상		yes	
ChallengeResponseAuthentication or KbdInteractiveAuthentication	no	yes	
UsePAM	no	yes	

sshd 설정이 끝나면 반드시 PAM 모듈이 제대로 추가되었는지 확인한 후 SSH Server의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > service sshd restart or systemctl restart sshd
```

```
sshd 를 정지 중: [ OK ]
sshd (을)를 시작 중: [ OK ]
```

Ubuntu, Debian or Linux Mint, Fedora:

```
$ systemctl restart ssh
```

만약, Ubuntu, Mint의 경우 ssh 재기동 후 접속이 되지 않으면 방화벽 설정의 문제이기 때문에 다음과 같은 명령어를 사용하여 방화벽 설정을 해제하고 재기동해야 한다.

```
$ sudo ufw disable
$ sudo service ufw restart
```

CentOS or RHEL:

```
$ service sshd restart or systemctl restart sshd
```

4) ACL(Access Control list) 설정

1) PAM 인증(환경 설정 정보를 File에 설정)인 경우 BaroPAM 모듈 사용 시 **2차 인증**에서 제외할 계정에 대한 ACL에 제외해야 하는 경우 BaroPAM 환경 설정 시 설정한 디렉토리에 ACL 파일을 생성한 후 제외할 계정을 다음과 같이 입력한다. (.baro_acl에 대한 파일 접근권한을 444로 설정해야 한다.)

```
[root] /usr/baropam > vi .baro_acl
barokey
baropam
```

2) PAM 인증(환경 설정 정보를 MariaDB에 설정)인 경우는 Mariadb의 ACL 설정 테이블을 사용해야 한다.

5) NTP(Network Time Protocol) 설정

BaroPAM은 시간 동기화 방식이므로 서버의 시간이 현재 시간과 다를 경우 **일회용 인증키**가 서로 일치하지 않아서 서버에 로그인을 못하는 경우가 발생할 수 있다.

최근에는 정보자산에 대한 시간 동기화(타임서버 시간 동기화)하는 방법으로 NTP(Network Time Protocol)을 이용하여 루트 계정에서 시스템의 시각을 현재 시각으로 설정할 수 있다.

NTP를 사용하기 위해서는 기본적으로 NTP 패키지가 반드시 설치되어 있어야 한다. 설치 확인은 다음의 명령어를 실행하여 확인한다. 만약, 설치되어 있지 않으면 Redhat, CentOS 8 이하 버전은 "yum install ntp" 그외는 "sudo apt-get install ntp" 명령어로 설치하면 된다.

```
[root]# rpm -qa | grep ntp
ntp-4.2.2p1-18.el5.centos
chkfontpath-1.10.1-1.1
```

ntpd 서비스를 서버 부팅 시 시작프로그램에 등록 및 ntp 활성화 여부 확인은 다음과 같은 명령어로 확인할 수 있다.

```
[root]# chkconfig ntpd on
[root]# chkconfig --list | grep ntp
ntpd          0:해제 1:해제 2:활성 3:활성 4:활성 5:활성 6:해제
```

chkconfig 이용하여 서버 부팅시 ntpd 데몬 활성화 여부 확인 3, 5 level에 off(해제) 가 되어 있으면 자

동 활성화 되지 않는다. 자동 활성화 하기 위해서는 3, 5에 on(활성)으로 다음과 같은 명령어로 변경해야 한다.

```
[root]# chkconfig --level 3 ntpd on
[root]# chkconfig --level 5 ntpd on
```

우리나라에서 운영되고 있는 NTP 서버는 다음과 같다.

```
server kr.pool.ntp.org
server time.bora.net
```

우리나라에서 운영되고 있는 NTP 서버를 ntpd 데몬 설정을 위한 설정 파일인 `"/etc/ntp.conf"`에 다음과 같이 설정한다.

```
[root]# vi /etc/ntp.conf
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#server 0.centos.pool.ntp.org
#server 1.centos.pool.ntp.org
#server 2.centos.pool.ntp.org
#server 3.centos.pool.ntp.org
server kr.pool.ntp.org iburst
server time.bora.net iburst
```

iburst 옵션은 일종의 옵션 설정으로써 동기화 하는데 걸리는 시간을 짧게 줄여주는 옵션임.

ntpд 데몬 설정을 위한 설정이 끝나면 반드시 NTP 설정이 제대로 추가되었는지 확인한 후 NTP 데몬의 Restart 작업이 반드시 필요하다.

```
[root]# /etc/init.d/ntpд restart
ntpд를 종료 중: [ OK ]
ntpд (을)를 시작 중: [ OK ]
```

ntpд 시간 확인은 다음과 같은 명령어로 확인할 수 있다.

```
[root]# ntpq -p
remote          refid          st t when poll reach  delay  offset  jitter
=====
*121.174.142.82 220.73.142.66 3 u 791 1024 377  9.333  -4.250  0.428
+time.bora.net  58.224.35.2   3 u 654 1024 367  2.926 -27.295 24.481
183.110.225.61 .INIT.         16 u  - 1024  0  0.000  0.000  0.000
LOCAL(0)       .LOCL.         10 l  39  64 377  0.000  0.000  0.001
```

* 표시된 ip 가 현재 시간을 가져오고 있는 ntp 서버임

NTP를 사용하기 위해서는 기본적으로 NTP 패키지가 반드시 설치되어 있어야 한다. 설치 확인은 다음의 명령어를 실행하여 확인한다. 만약, 설치되어 있지 않으면 Redhat, CentOS 8 이상 버전은 `"dnf install chrony"` 명령어로 설치하면 된다.

```
[root@baropam ~]# rpm -qa | grep chrony
chrony-3.5-1.el8.x86_64
```

우리나라에서 운영되고 있는 NTP 서버는 다음과 같다.

```
server kr.pool.ntp.org
server time.bora.net
```

우리나라에서 운영되고 있는 NTP 서버를 ntpd 데몬 설정을 위한 설정 파일인 `"/etc/chrony.conf"`에 다음과 같이 설정한다.

```
[root@baropam ~]# vi /etc/chrony.conf

# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#pool 2.centos.pool.ntp.org iburst
server kr.pool.ntp.org iburst
server time.bora.net iburst

# Record the rate at which the system clock gains/losses time.
driftfile /var/lib/chrony/drift

# Allow the system clock to be stepped in the first three updates
# if its offset is larger than 1 second.
makestep 1.0 3

# Enable kernel synchronization of the real-time clock (RTC).
rtcsync

# Enable hardware timestamping on all interfaces that support it.
#hwtimestamp *

# Increase the minimum number of selectable sources required to adjust
# the system clock.
#minsources 2

# Allow NTP client access from local network.
allow 192.168.0.0/16

# Serve time even if not synchronized to a time source.
#local stratum 10

# Specify file containing keys for NTP authentication.
keyfile /etc/chrony.keys

# Get TAI-UTC offset and leap seconds from the system tz database.
leapsectz right/UTC

# Specify directory for log files.
logdir /var/log/chrony

# Select which information is logged.
#log measurements statistics tracking
```

ntpd 데몬 설정을 위한 설정이 끝나면 반드시 NTP 설정이 제대로 추가되었는지 확인한 후 NTP 데몬의 Restart 작업이 반드시 필요하다. (chrony 서비스 시작 및 부팅시 구동 등록)

```
[root@baropam ~]# sudo systemctl enable chronyd
```

```
[root@baropam ~]# sudo systemctl restart chronyd
```

ntpd 시간 확인은 다음과 같은 명령어로 확인할 수 있다.

시간을 받아오는 서버 리스트 / chrony.conf 파일에 등록된 server 리스트)

```
[root@baropam ~]# chronyc sources
210 Number of sources = 2
MS Name/IP address          Stratum Poll Reach LastRx Last sample
-----
^* ec2-54-180-134-81.ap-nor>  2  6  377  43  -349us[-1059us] +/-  24ms
^ time.bora.net              2  6  377  42  +1398us[+1398us] +/-  90ms
```

시간을 받아 오는 서버 정보)

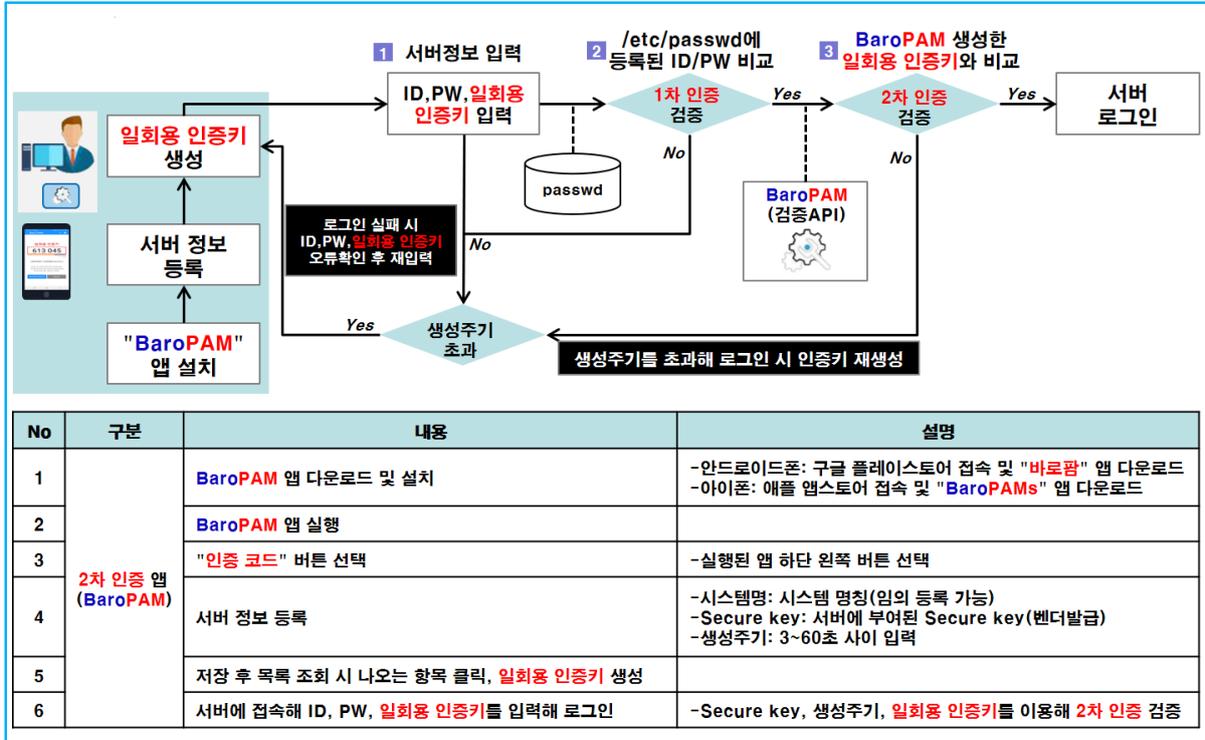
```
[root@baropam ~]# chronyc tracking
Reference ID      : 36B48651 (ec2-54-180-134-81.ap-northeast-2.compute.amazonaws)
Stratum          : 3
Ref time (UTC)   : Sun Mar 22 07:07:43 2020
System time      : 0.000130027 seconds slow of NTP time
Last offset      : -0.000710122 seconds
RMS offset       : 0.000583203 seconds
Frequency        : 19.980 ppm fast
Residual freq    : +0.142 ppm
Skew             : 3.235 ppm
Root delay       : 0.013462566 seconds
Root dispersion  : 0.017946836 seconds
Update interval  : 65.0 seconds
Leap status      : Normal
```

시간 상태 및 동기화 등 정보 확인)

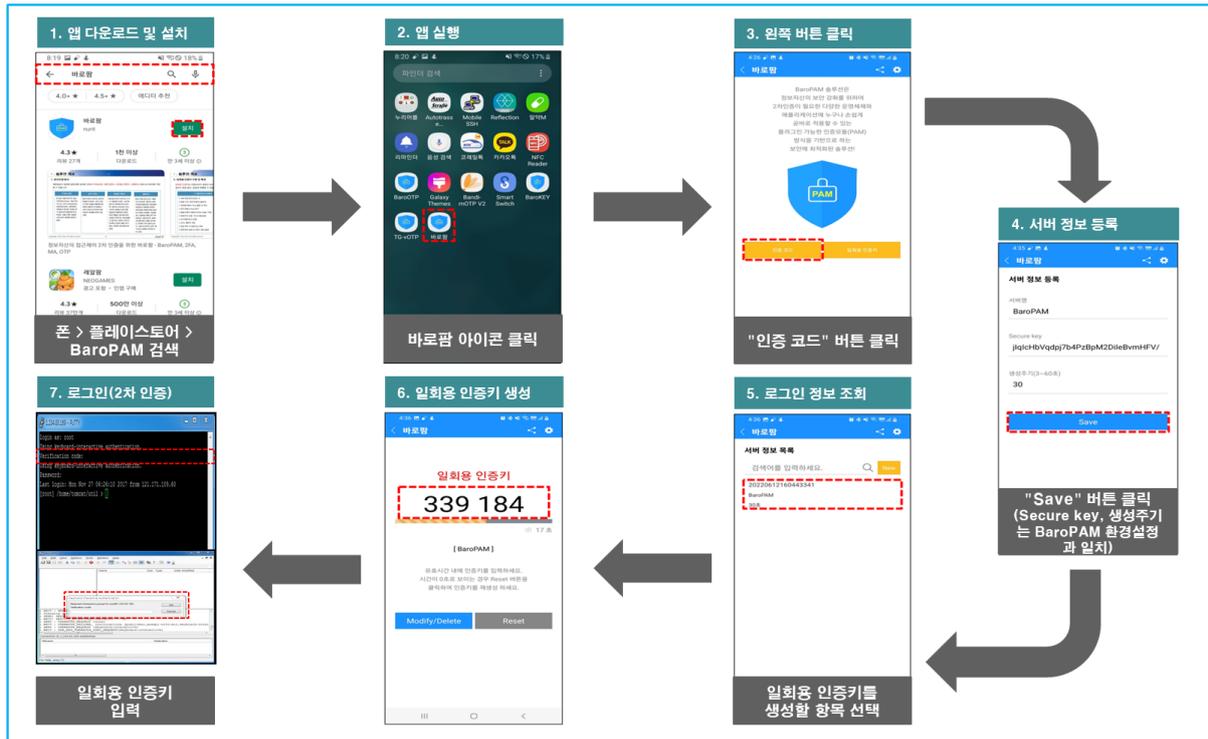
```
[root@baropam ~]# timedatectl status
Local time: Sun 2020-03-22 16:08:45 KST
Universal time: Sun 2020-03-22 07:08:45 UTC
RTC time: Sun 2020-03-22 07:08:44
Time zone: Asia/Seoul (KST, +0900)
System clock synchronized: yes
NTP service: active
RTC in local TZ: no
```

2. BaroPAM 적용

2.1 BaroPAM 적용 프로세스



2.2 BaroPAM 적용 화면



2.3 Linux 로그인 방법

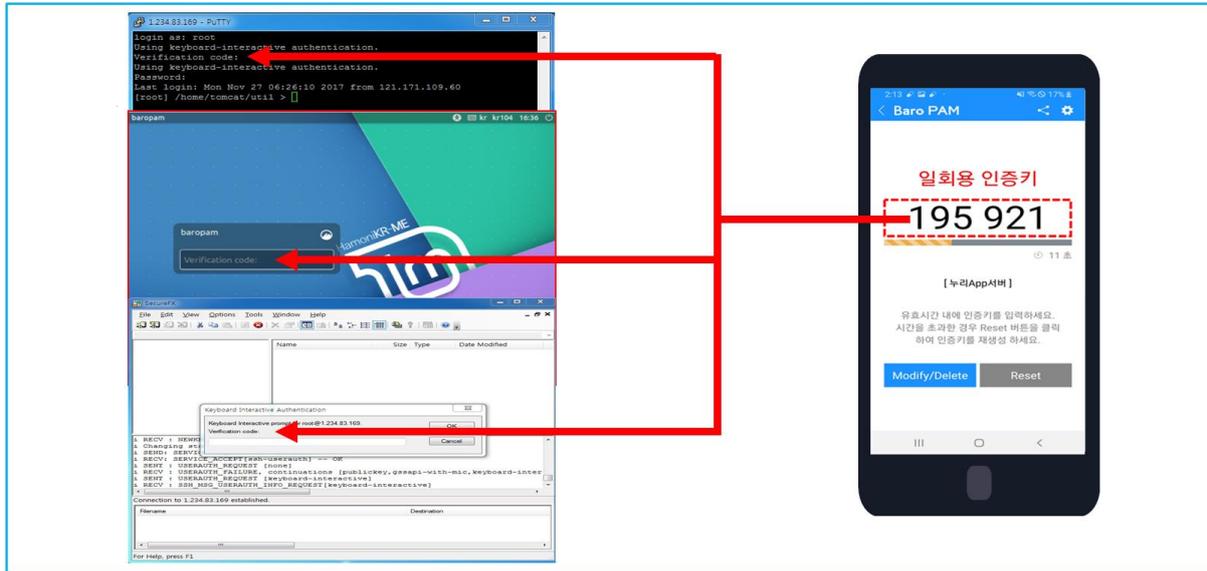
먼저, "BaroPAM Setup" 화면에서 입력한 "인증주기, Secure key, 서버명"을 "BaroPAM" 앱의 "서버 정보 등록" 화면에서 동일하게 입력해야 한다.

```

$ cat .baro_auth
" AUTH_KEY
" RATE_LIMIT 3 30
" KEY_METHOD app512
" CYCLE_TIME 30
" SECURE_KEY j1q1cHbVqdpj7b4PzBpM2DileBvmHFV/
" ACL_NAME /usr/baropam/.baro_acl
" ACL_TYPE deny
" HOSTNAME BaroPAM
" DISALLOW_REUSE
33458936
19035576
15364353
54649370
84342192
                
```

Linux/Unix 환경에 로그인 시 사용자 계정(Username)을 입력하고, 스마트 폰의 "BaroPAM" 앱에서 일회용

인증키를 생성한 후 "Verification code:"에 생성한 일회용 인증키와 "Password"를 입력한 후 "Enter" 버튼을 클릭하면 BaroPAM 모듈에 인증을 요청하여 검증이 성공하면 Linux/Unix의 로그인 인증 정책이 적용된다.



Linux/Unix의 로그인 화면에서 입력한 일회용 인증키를 BaroPAM 검증모듈에서 인증에 실패하면 "Access denied." 메시지가 로그인 화면에 나타난다. BaroPAM의 인증과 관련한 각종 메시지는 syslog에 남긴다.

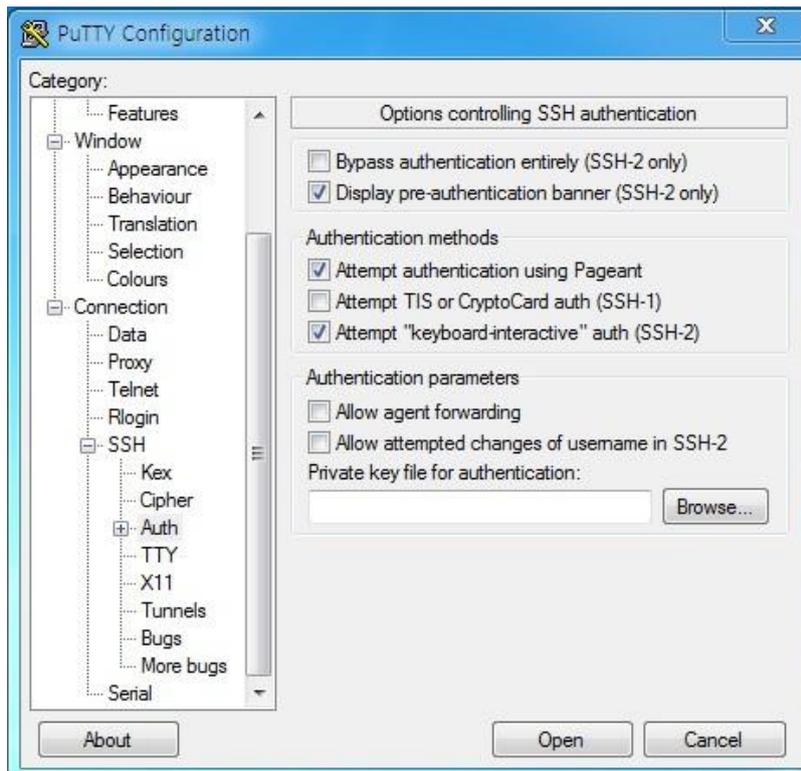
```

Mar 25 11:10:42 qsh-0415 sshd[27482]: pam_unix(sshd:session): session closed for user root
Mar 25 13:52:25 qsh-0415 sshd(pam_baro_auth)[2052]: Try to update RATE_LIMIT line.[3 30 1648183945]
Mar 25 13:52:45 qsh-0415 sshd[2050]: Accepted keyboard-interactive/pam for root from 222.108.117.41 port 49835 ssh2
Mar 25 13:52:45 qsh-0415 sshd[2050]: pam_unix(sshd:session): session opened for user root by (uid=0)
Mar 25 15:25:47 qsh-0415 sshd(pam_baro_auth)[14119]: Try to update RATE_LIMIT line.[3 30 1648189547]
Mar 25 15:25:49 qsh-0415 sshd(pam_baro_auth)[14119]: Verification code generation failed.[Success]
Mar 25 15:25:49 qsh-0415 sshd(pam_baro_auth)[14119]: Invalid verification code
Mar 25 15:25:51 qsh-0415 sshd[14118]: Received disconnect from 222.108.117.41: 13: The user canceled au
    
```

2.4 ssh/sftp 접속 틀

putty인 경우)

Putty로 접속할 때 보통 접속 과정과 동일하게 해주시면 되는데, 하나 설정해 주어야 할 것이 있다. 환경 설정에서 "connection -> SSH -> auth"에서 attempt "Keyboard-Interactive" auth(SSH-2)를 선택한 후 SSH 접속을 하면 된다.

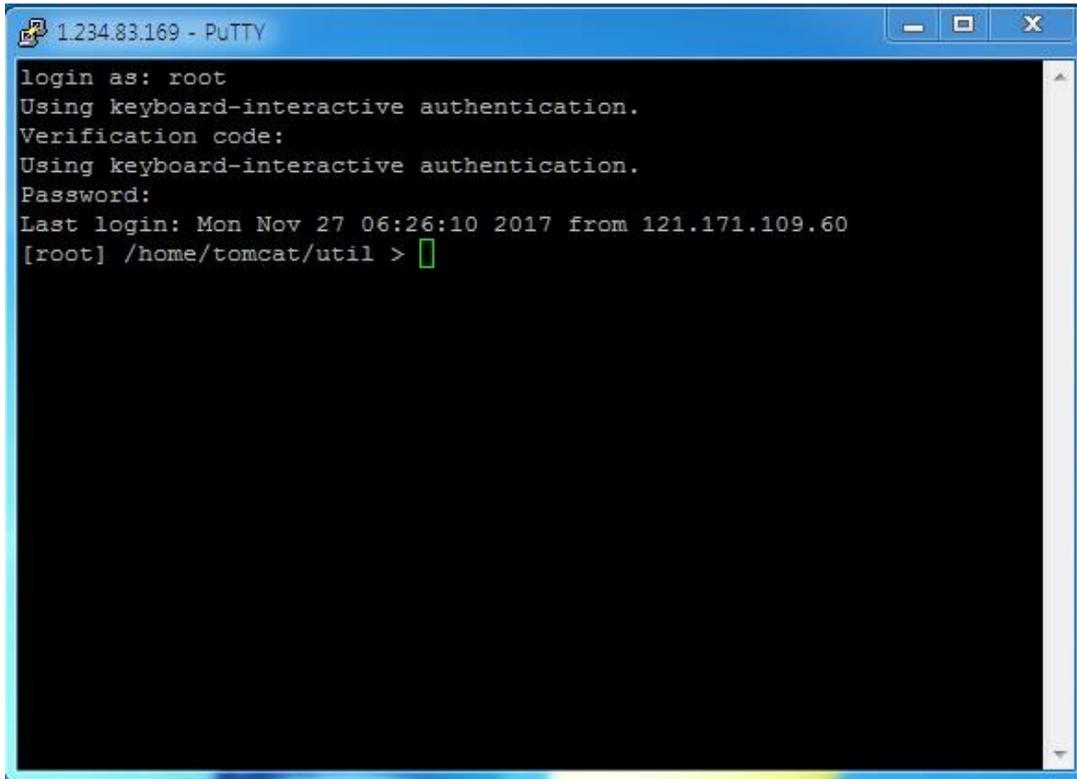


PuTTY Download 및 Documentation 관련 자료는 다음 URL에서 찾을 수 있다.

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

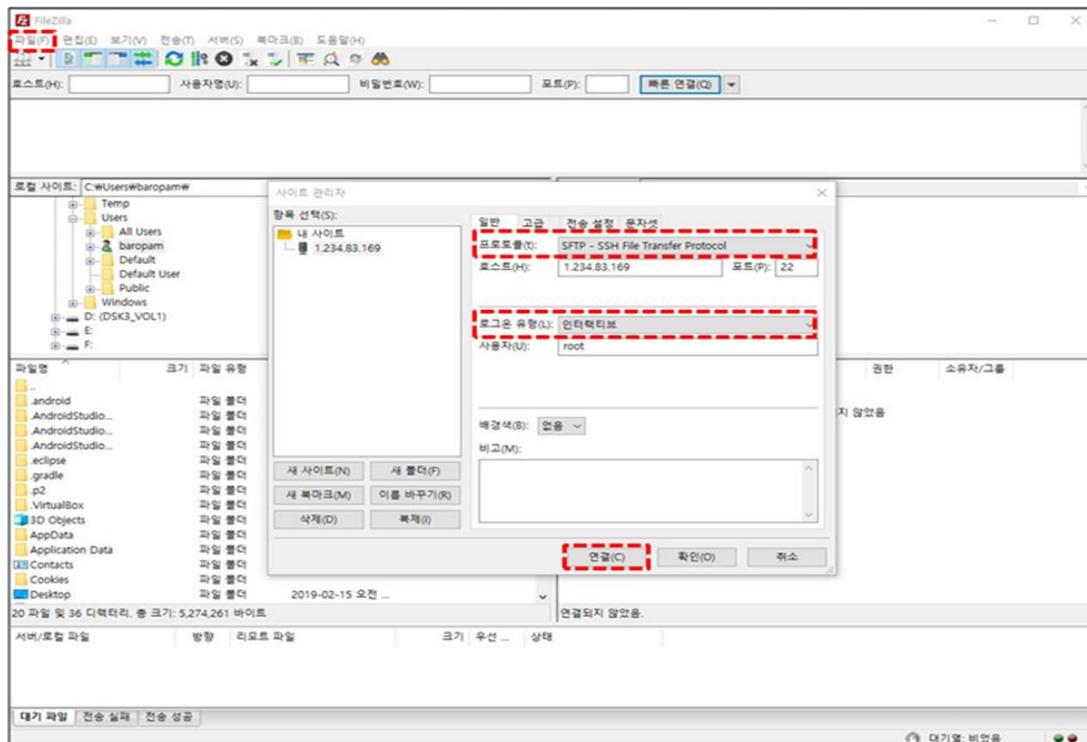
"Verification code:"를 입력하라는 메시지가 표시되면 BaroPAM 앱에서 생성한 일회용 인증기를 입력한다.

인증에 성공하면 다음과 같이 SSH 로그인 비밀번호를 입력 할 수 있다.

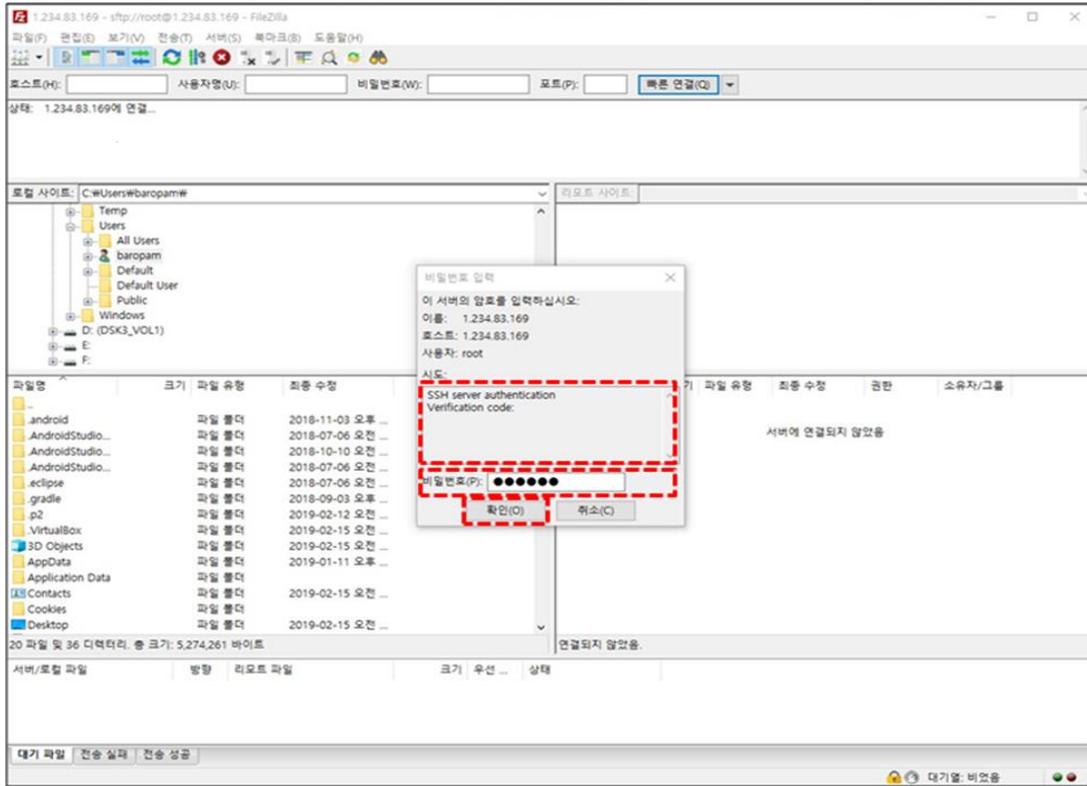


FileZilla인 경우)

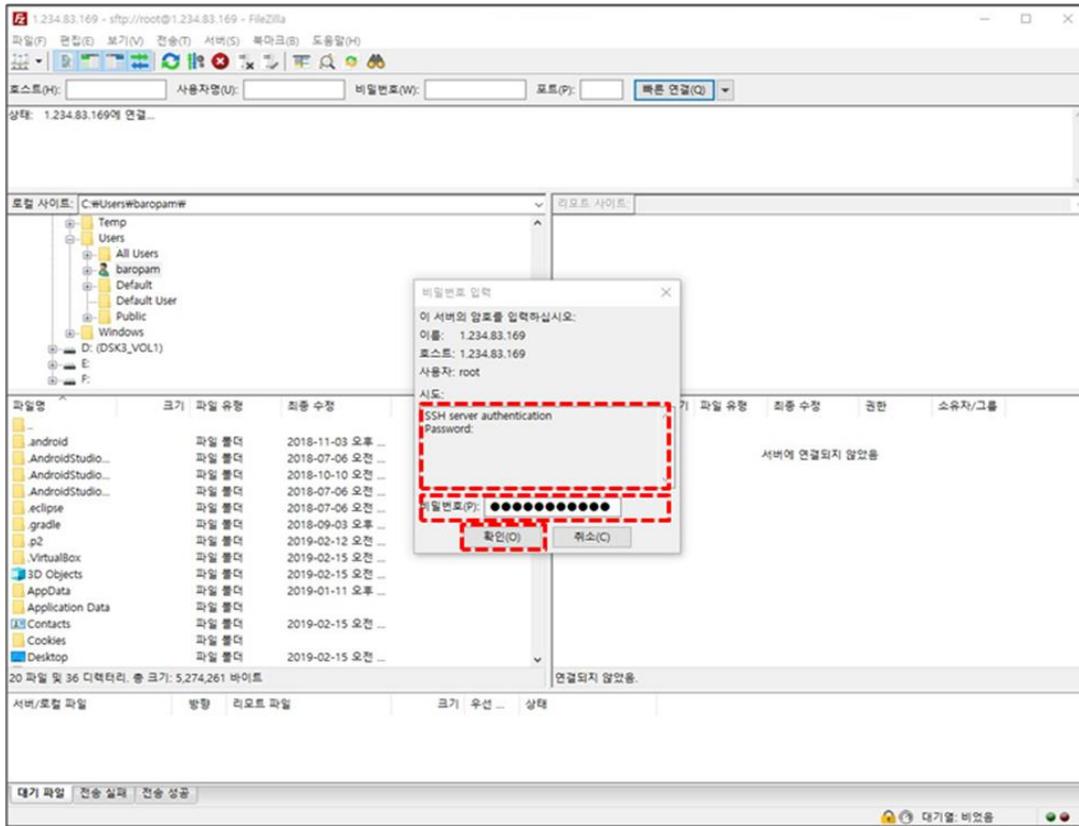
FileZilla로 접속할 때 보통 접속 과정과 다른데, 상단 왼쪽 메뉴에서 "파일(F) -> 사이트 관리자(S)"를 선택하여 일반 탭 화면에서 "프로토콜(t):" 항목에서 "SFTP - SSH File Transfer Protocol"과 "로그온 유형(L):" 항목에서 "인터랙티브"를 선택한 후 다음과 같이 "연결(C)" 버튼을 클릭한다.



그러면 다음과 같이 비밀번호 입력 화면이 나타난다. 비밀번호 입력 화면에서 "시도:" 내용을 확인하고, 스마트 폰에서 생성한 일회용 인증키를 "비밀번호(P):" 입력 항목에 입력한 후 "확인(O)" 버튼을 클릭한다.



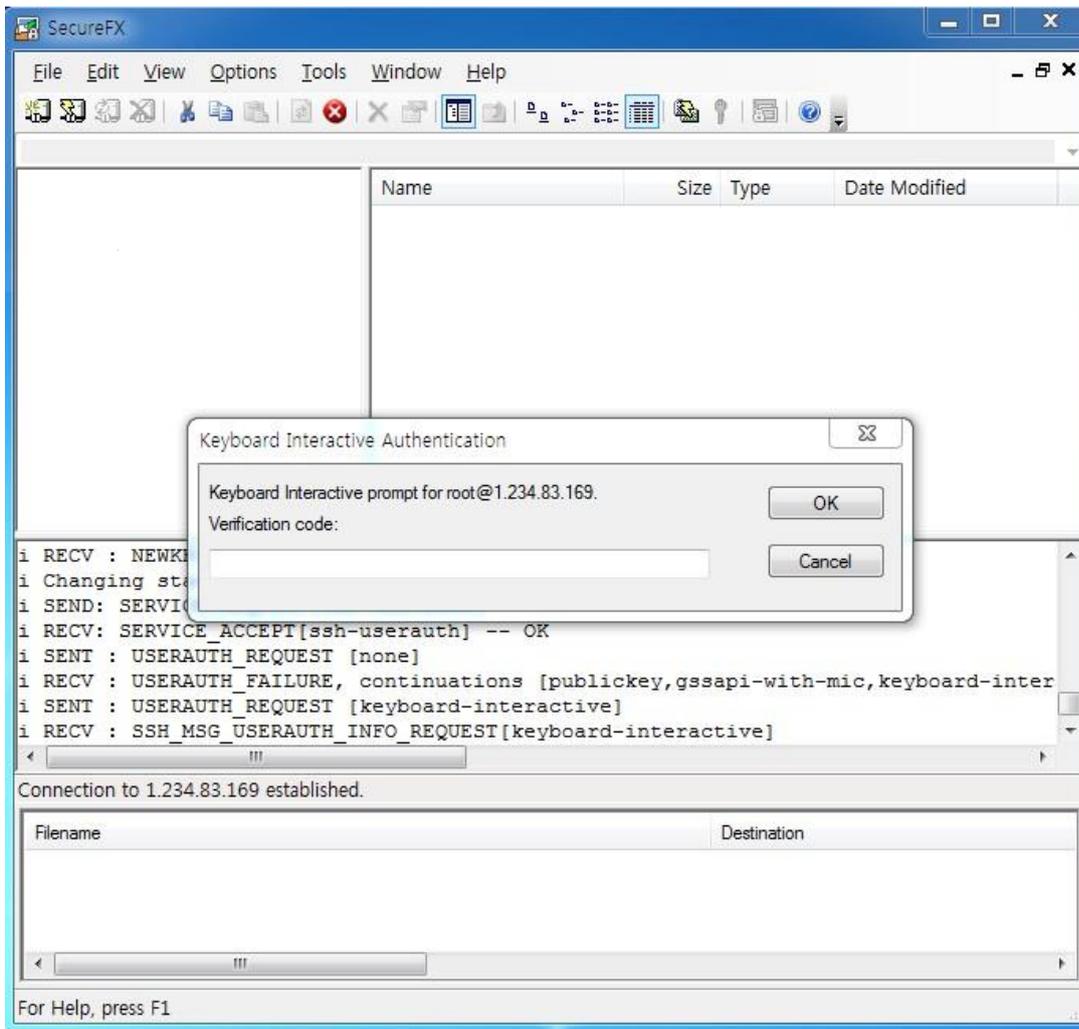
그러면 다음과 같이 비밀번호 입력 화면이 나타난다. 비밀번호 입력 화면에서 "시도:" 내용을 확인하고, 로그인 계정에 대한 비밀번호를 "비밀번호(P):" 입력 항목에 입력한 후 "확인(O)" 버튼을 클릭하여 서버에 접속한다.



SFTP인 경우)

"Verification code:" 를 입력하라는 메시지가 표시되면 BaroPAM 앱에서 생성한 일회용 인증기를 입력한다.

인증에 성공하면 다음과 같이 SFTP 로그인 비밀번호를 입력 할 수 있다.



SecureFX Download 및 Documentation 관련 자료는 다음 URL에서 찾을 수 있다.

<https://www.vandyke.com/>

결론적으로, **2차 인증**은 추가 보호 계층을 추가하여 암호 인증을 보호하는 효과적인 수단이 될 수 있으며, 사용 여부와 상관없이 사용자의 선택에 달려 있지만 **2차 인증**의 채택은 산업의 동향이다.

3. BaroPAM 제거

3.1 BaroPAM 환경 제거

BaroPAM이 설치된 상태에서 BaroPAM 모듈을 사용하지 않을 경우 sshd, su, sudo 파일 등에 설정한 내용을 제거하는 방법은 다음과 같이 주석(#) 처리나 삭제하면 된다.

```
[root] /usr/baropam > vi /etc/pam.d/sshd
#%PAM-1.0
#auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

sshd 데몬에 설정한 "/etc/ssh/sshd_config" 파일의 내용 중 다음과 같은 인자를 변경해야 한다.

인자	기존	변경	비고
PasswordAuthentication	no	yes	
ChallengeResponseAuthentication	yes	no	
UsePAM	yes	no	

sshd 설정이 끝나면 반드시 PAM 모듈이 제대로 제거되었는지 확인한 후 SSH Server의 Restart 작업이 반드시 필요하다.

```
[root] /usr/baropam > service sshd restart
sshd 를 정지 중: [ OK ]
sshd (을)를 시작 중: [ OK ]
```

Ubuntu, Debian or Linux Mint:

```
$ service ssh restart
```

Fedora:

```
$ systemctl restart sshd
```

CentOS or RHEL:

```
$ service sshd restart
```

4. BaroPAM FAQ

현상 : 일회용 인증키가 맞지 않아서 로그인을 하지 못하는 경우

원인 : BaroPAM은 시간 동기화 방식으로 폰과 Windows나 Server의 시간이 동일해야 함,

조치 : 폰과 Windows나 Server의 시간이 맞는지 확인.

현상 : Feb 7 07:59:09 eactive sshd(pam_baro_auth)[29657]: ACL file ".baro_acl" must only be accessible by user id root

원인 : .baro_acl 파일의 Permission이 다름.

조치 : .baro_acl 파일의 Permission를 444로 설정.

현상 : Feb 7 08:02:15 eactive sshd(pam_baro_auth)[29739]: Failed to acl file read ".baro_acl"

원인 : .baro_acl 파일이 존재하지 않는 경우에 발생.

조치 : baropam 홈 디렉토리에 .baro_acl 파일을 생성. (Permission를 444로 설정)

현상 : Cannot look up user id xxxxx

원인 : 사용자 ID xxxxx를 조회 할 수 없는 경우 발생.

조치 : /etc/passwd 파일에 user id xxxxx를 등록.

현상 : Failed to secret file read .baro_auth

원인 : Secret file이 존재하지 않은 경우에 발생.

조치 : Secret file의 존재여부를 확인.

현상 : Secret file .baro_auth must only be accessible by root

원인 : .baro_auth 파일의 Permission이 다른 경우에 발생.

조치 : .baro_auth 파일의 Permission를 444로 설정.

현상 : Invalid file size for .baro_auth

원인 : .baro_auth 파일의 크기가 1 < size < 64K 가 아닌 경우 발생.

조치 : .baro_auth 파일의 크기를 확인.

현상 : Could not read .baro_auth

원인 : .baro_auth 파일이 존재하지 않거나 파일의 Permission이 444가 아닌 경우 발생.

조치 : .baro_auth 파일의 존재여부 및 파일의 Permission를 확인.

현상 : Invalid file contents in .baro_auth

원인 : .baro_auth 파일의 내용(규칙)이 잘못된 경우에 발생.

조치 : .baro_auth 파일의 내용을 확인.

현상 : Failed to create tmp secret file[error message]

원인 : 임시 secret file을 생성하지 못한 경우에 발생.

조치 : 임시 secret file을 생성하지 못한 이유는 error message를 확인.

현상 : Failed to open tmp secret file .baro_auth~[error message]

원인 : 1. Redhat, CentOS인 경우 SELINUX를 비활성화하지 않아 보안문제로 막혀서 발생

2. 임시 secret file인 .baro_auth~을 오픈하지 못한 경우에 발생.

조치 : 1. "/etc/sysconfig/selinux"에서 SELINUX를 비활성화(SELINUX=enforcing → disabled)

```
[root] /etc > vi /etc/sysconfig/selinux
# 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 - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
# targeted - Only targeted network daemons are protected.
# strict - Full SELinux protection.
SELINUXTYPE=targeted

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

바로 적용은 되지 않으며 재부팅을 해야 적용이 된다. 재부팅을 하지 않고 현재 접속된 터미널에 한해 변경된 내용을 적용하고 싶을 경우 다음의 명령어를 실행하면 된다.

```
[root] /etc > /usr/sbin/setenforce 0
```

2. 임시 secret file인 .baro_auth~을 오픈하지 못한 이유는 error message를 확인.

현상 : Secret file .baro_auth changed while trying to use one-time authentication key

원인 : 일회용 인증키를 사용하는 동안 비밀 파일 .baro_auth가 변경된 경우 발생.

조치 : 다시 로그인을 시도.

현상 : Failed to update secret file .baro_auth[error message]

원인 : secret file을 변경하지 못한 경우에 발생.

조치 : secret file을 변경하지 못한 이유는 error message를 확인.

현상 : Invalid RATE_LIMIT option. Check .baro_auth

원인 : Secret file인 .baro_auth 파일의 내용 중 RATE_LIMIT 설정값이 잘못 설정되어 있는 경우 발생.

조치 : 제한 횟수($1 < \text{RATE_LIMIT} < 100$), 제한 시간($1 < \text{interval} < 3600$)의 설정 값을 확인.

현상 : Invalid list of timestamps in RATE_LIMIT. Check .baro_auth

원인 : Secret file인 .baro_auth 파일의 내용 중 RATE_LIMIT 옵션에 Update된 timestamps가 잘못된 경우 발생.

조치 : Secret file인 .baro_auth 파일의 RATE_LIMIT 옵션에 Update된 timestamps를 확인.

현상 : Try to update RATE_LIMIT line.

원인 : 정상적으로 로그인 한 경우 출력되는 메시지.

조치 : No action

현상 : Too many concurrent login attempts. Please try again.

원인 : Secret file인 .baro_auth 파일의 DISALLOW_REUSE 옵션(일회용 인증키 생성 주기 내에는 하나의 로그인만 가능)이 설정된 경우

로그인 성공 후 일회용 인증키 생성 주기 내에 로그인을 재 시도한 경우 발생.

조치 : 일회용 인증키 생성 주기 후에 로그인 재 시도.

현상 : Trying to reuse a previously used time-based code.

Retry again in 30 seconds.

Warning! This might mean, you are currently subject to a man-in-the-middle attack.

원인 : Secret file인 .baro_auth 파일의 DISALLOW_REUSE 옵션은 중간자 공격(man-in-the-middle)을 대비한 옵션.

중간자 공격(man-in-the-middle)은 권한이 없는 개체가 두 통신 시스템 사이에서 스스로를 배치하고 현재 진행 중인 정보의 전달을 가로채면서 발생.

간단히 말해서, 현대판 도청 시스템이라고 할 수 있는 것

조치 : No action

현상 : Failed to allocate memory when updating .baro_auth

원인 : Secret file인 .baro_auth를 업데이트 할 때 메모리 할당에 실패한 경우 발생.

조치 : Technical support

현상 : Can't find SECURE_KEY[error message]

원인 : Secret file인 .baro_auth 파일의 SECURE_KEY 옵션이나 설정값이 없는 경우에 발생.

조치 : Secret file인 .baro_auth 파일의 SECURE_KEY 옵션이나 설정값 확인.

현상 : Verification code generation failed.[error message]

원인 : 일회용 인증키 검증에 실패한 경우 발생.

조치 : 로그인 재 시도.

현상 : Invalid verification code

원인 : 일회용 인증키 검증에 실패한 경우 발생.

조치 : 로그인 재 시도.

현상 : Invalid verification code

Can not make/remove entry for session.

원인 : 서버의 시스템 시간이 맞지 않아서 발생.

조치 : date 명령어로 서버의 시스템 시간이 맞는지 확인하여 틀리면 시간을 맞춰줘야 함.

1. date 명령어 서버의 시스템 시간을 변경(임시 방편)

2. ntp가 설정되어 있는지 확인하여 설정되어 있으면 ntp 시간을 설정하는 주기를 줄여 주어야 하며, 설정되어 있지 않으면 ntp를 설정해야 함.

현상 : Mar 12 15:37:01 baropam gdm(pam_baro_auth)[1215]: [ID 128276 auth.error] No user name available when checking verification code

원인 : 인증 코드를 검증할 때 사용 가능한 사용자가 아닌 경우(등록된 사용자가 아닌 경우 발생).

조치 : 시스템 관리자에게 로그인-ID가 등록되어 있는지 확인.

현상 : Apr 3 13:06:13 kdn sshd[3577]: PAM unable to dlopen(/usr/baropam/pam_baro_auth.so):
/usr/baropam/pam_baro_auth.so: cannot open shared object file: No such file or directory
Apr 3 13:06:13 kdn sshd[3577]: PAM adding faulty module: /usr/baropam/pam_baro_auth.so

원인 : 1. /usr/baropam/pam_baro_auth.so 파일이 존재하지 않아서 발생.

2. 설치된 pam_baro_auth.so 모듈이 OS 버전과 맞지 않아서 발생.

조치 : 1. BaroPAM 모듈 파일(pam_baro_auth.so)의 존재하는지 확인하여 없으면 BaroPAM의 설치 파일에서 복사한다.

2. OS 버전을 확인한 후 OS 버전에 맞는 BaroPAM 모듈 다운로드하여 재설치해야 함.

현상 : mm_log_handler: write: Broken pipe

mm_request_send: write: Broken pipe

원인 : 이것은 몇 초 내에 서버에 keepalive 메시지를 보내야하는 빈도.

서버가 너무 오래 유휴 상태 인 연결을 닫을 수 있다. 클라이언트 (ServerAliveInterval) 또는 서버 (ClientAliveInterval)를 업데이트 할 수 있다.

조치 : 클라이언트 시스템의 /etc/ssh/ ssh_config에 ServerAliveInterval을 설정하거나 서버 시스템의 /etc/ssh/sshd_config에 ClientAliveInterval을 설정할 수 있다. 오류가 계속 발생하면 간격을 줄여야 한다.

ServerAliveInterval => 서버로부터 아무런 데이터도받지 못한 경우, ssh (1)는 암호화 된 채널을 통해 메시지를 보내 서버의 응답을 요청하는 시간 초과 간격을 초 단위로 설정한다. 기본값은 0이며이 메시지가 서버로 전송되지 않음을 나타낸다. 이 옵션은 프로토콜 버전 2에만 적용된다.

ClientAliveInterval => 클라이언트로부터 아무런 데이터도받지 못하면, sshd (8)는 클라이언트로부터 응답을 요청하기 위해 암호화 된 채널을 통해 메시지를 보낸다.
기본값은 0이다.이 메시지는 클라이언트에 전송되지 않음을 나타낸다.
이 옵션은 프로토콜 버전 2에만 적용된다.

To update your server(and restart your sshd) => 서버를 업데이트하고(sshd를 다시 시작하려면)
echo "ClientAliveInterval 60" | sudo tee -a /etc/ssh/sshd_config

Or client-side: => Or client-side:
echo "ServerAliveInterval 60" >> ~/.ssh/config

ClientAliveInterval: 클라이언트 살아있는지 확인하는 간격
ClientAliveCountMax: 클라이언트 응답 없어도 접속 유지하는 횟수
예를 들어, ClientAliveInterval=15, ClientAliveCountMax=3 이면 45초 후 접속 끊김

현상 : May 19 12:37:37 baropam sshd(pam_baro_auth)[1416]: Failed to acl file read "(null)"

원인 : acl file 존재여부 및 파일 permission 문제로 발생
조치 : 빈 acl 파일을 .baro_acl 파일을 444 권한으로 생성함.

현상 : Failed to compute location of secret file

원인 : pam에 설정된 secret file이 해당 디렉토리에 존재하지 않은 경우 발생.
조치 : pam에 설정된 secret file이 해당 디렉토리에 존재하지 않으면 secret file을 해당 디렉토리에 생성해 줘야 함.
ex) auth required /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no

현상 : Failed to compute location of encrypt flag

원인 : pam에 암호화 플래그가 존재하지 않은 경우 발생.
조치 : pam에 암호화 플래그(yes, no)을 설정해 줘야 함.
ex) auth required /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no

현상 : HamoniKR OS 설치 후 ssh 접속이 안된 경우

원인 : HamoniKR OS의 방화벽이 설정되어 있어서 발생
조치 : HamoniKR OS의 방화벽이 설정을 해제한 후 ufw를 재기동하면 됨.
> sudo ufw disable
> sudo service ufw restart

현상 : Grooroom OS 재부팅 후 Screen saver에 적용한 BaroPAM 이 해제되는 현상

원인 : Grooroom OS는 재부팅하면 Screen saver와 관련된 설정파일인 lightdm이 초기화되어 발생
조치 : 원복 파일인 "/usr/share/debian-system-adjustments/pam.d/lightdm" 파일에 BaroPAM을 설정하면 됨.

현상 : Oct 14 10:09:43 baropam sshd[18075]: PAM unable to dlopen(/usr/baropam/pam_baro_auth.so):
/usr/baropam/pam_baro_auth.so: undefined symbol: curl_easy_setopt

원인 : 웹 개발 툴인 cURL(Client for URLs) 관련 라이브러리가 존재하지 않아서 발생.
조치 : Redhat, CentOS는 "yum install curl" 그외는 "sudo apt-get install curl" 명령어로 설치

현상 : Did not receive verification code from user

error: ssh_msg_send: write: Broken pipe
원인 : Secure key가 잘못 설정된 경우에 발생
조치 : 설정된 Secure key를 확인.
번더에서 제공된 Secure key인지 확인.

현상 : PAM: authentication thread exited unexpectedly.

*** glibc detected *** su: free(): invalid pointer: 0x00002aede020c9e2 ***

원인 : BaroPAM 환경 설정 파일(.baro_nurit)이 존재하지 않는 경우에 발생.

조치 : BaroPAM 환경 설정 파일(.baro_nurit)의 존재하는지 확인하여 없으면 BaroPAM의 설치 파일에서 복사한다.

현상 : 개방형OS인 구름OS에서 비밀번호를 변경한 후 로그인에 실패하여 로그인이 안되는 현상 발생.

Jul 8 09:31:51 gooroom lightdm: pam_unix(lightdm-greeter:session): session closed for user lightdm

Jul 8 09:31:51 gooroom lightdm: pam_unix(lightdm:session): session opened for user baropam(uid=1000) by (uid=0)

Jul 8 09:31:51 gooroom systemd-logind[446]: Removed session c1.

Jul 8 09:31:51 gooroom systemd-logind[446]: New session 4 of user baropam.

Jul 8 09:31:51 gooroom lightdm: gkr-pam: gnome-keyring-daemon started properly and unlocked keyring

Jul 8 09:31:51 gooroom systemd-logind[446]: Removed session 4.

Jul 8 09:31:52 gooroom lightdm: pam_unix(lightdm-greeter:session): session opened for user lightdm(uid=104) by (uid=0)

Jul 8 09:31:52 gooroom systemd-logind[446]: New session c2 of user lightdm.

원인 : 약한 비밀번호로 변경한 경우 발생.

조치 : 대소문자를 포함해서 8자리 이상의 강한 비밀번호로 변경.

현상 : 개방형OS인 구름OS에서 BaroPAM을 적용한 뒤 로그인에 실패하는 현상 발생.

원인 : lightdm에 BaroPAM을 설정할 때 파라미터 중 nullok로 설정하여 발생.

조치 : lightdm에 BaroPAM을 설정할 때 파라미터 중 nullok를 forward_pass로 변경

현상 : No supported authentication methods available (server sent publickey,gssapt-keyex,gssapt-with-mic)

원인 : Interactive mode를 지원하지 않음.(/etc/pam.d/sshd 설정 시 nullok를 설정하지 말고 forward_pass로 설정해야 함)

조치 : "/etc/ssh/sshd_config" 파일에서 "PasswordAuthentication yes"로 변경 후 sshd restart

현상 : Linux 서버에 BaroPAM 적용 후 일회용 인증키를 입력하는 항목(Verification code: 또는 Password & Verification code:)을 스킵(Skip)하여 로그인이 안되는 현상 발생

서버 접근제어 솔루션이 적용되어 있는 경우 BaroPAM을 적용 했는데, 로그인 되지 않는 현상

원인 : 서버 접근제어 솔루션에서 /etc/pam.d/sshd 설정한 것 보다 BaroPAM 설정이 앞에 설정하여 발생함

조치 : 다음과 같이 /etc/pam.d/sshd 설정의 순서를 변경하면 됨.

변경 전)

```
auth      required      /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
auth      required      pam_sepermit.so
auth      include      password-auth
account   required      pam_nologin.so
account   include      password-auth
password  include      password-auth
```

변경 후)

```
auth      required      pam_sepermit.so
auth      substack      password-auth
account   required      pam_nologin.so
account   include      password-auth
password  include      password-auth
```

```
auth    required    /usr/baropam/pam_baro_auth.so nullok secret=/usr/baropam/.baro_auth
encrypt=no
```

PAM 설정 시 특정 모듈의 성공과 실패를 어떻게 처리할 것인지를 나타내는 것을 Control이라 한다.

Control 중 include과 substack은 다른 PAM 관련 모듈을 불러오는 것은 동일 하지만, substack은 substack의 동작 결과에 따라 나머지 모듈을 처리하지 않는다는 차이점이 있다.

5. MySQL/MariaDB 설치 및 구성

5.1 MariaDB 설치

```
[root@localhost ~]# dnf -y install mariadb-server
```

MariaDB 설치 작업이 정상적으로 끝나면 다음과 같은 명령어를 사용하여 MariaDB를 기동시킨다.

```
[root@localhost ~]# systemctl start mariadb
```

MariaDB와 함께 제공되는 스크립트를 실행하여 MariaDB 보안 옵션을 개선하기 위한 몇 가지 단계를 진행해야 한다.

```
[root@localhost ~]# mysql_secure_installation
```

일련의 프롬프트가 표시되며, 비밀번호를 설정한 것을 모르는 경우 프롬프트가 표시되면 Enter 키를 누른다.

```
Enter current password for root (enter for none): Enter
```

다음으로 새 root 비밀번호를 설정할 것인지 확인하고 강력한 비밀번호를 설정한다.

```
Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.
Set root password? [Y/n] y
New password: baropam
Re-enter new password: baropam
Password updated successfully!
Reloading privilege tables..
... Success!
```

다음으로 이어지는 프롬프트에 대해 Enter 키를 누르기만 하면 된다.

익명 사용자를 제거한다.

```
Remove anonymous users? [Y/n] y
... Success!
```

root 로그인을 원격으로 허용하지 않는다.

```
Disallow root login remotely? [Y/n] y
... Success!
```

테스트 데이터베이스를 제거한다.

```
Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
```

```
... Success!
```

권한 테이블을 다시 로드한다.

```
Reload privilege tables now? [Y/n] y
... Success!
Cleaning up...
```

5.2 MariaDB 구성

먼저 FreeRADIUS에 대한 데이터베이스와 데이터베이스 사용자를 생성한 다음 데이터베이스와 비밀번호로 식별되는 사용자를 생성한다.

예)

```
Database: baropamdb
User: nurit
Password: baropams
```

사용자와 비밀번호를 원하는 대로 바꿀 수 있지만 값을 적절하게 바꾸려면 나중에 수행할 구성에 주의를 기울여야 한다.

root로 MySQL/MariaDB 콘솔에 액세스하여 시작한다.

```
[root@baropam /]# mysql -uroot -p
Enter password: baropam
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.26 Source distribution

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\w' to clear the current input statement.
```

명령을 실행하여 데이터베이스 및 사용자를 생성한다.

```
MariaDB [(none)]> CREATE DATABASE baropamdb;
MariaDB [(none)]> CREATE USER 'nurit'@'localhost' IDENTIFIED BY 'baropams';
MariaDB [(none)]> GRANT ALL ON baropamdb.* TO 'nurit'@'localhost';
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> use baropamdb
```

다음으로 RADIUS MySQL 스키마를 새로 생성된 데이터베이스로 생성한다.

```
#
# Table structure for table 'TB_BARO_HOST'
#
```

```

CREATE TABLE IF NOT EXISTS TB_BARO_HOST (
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  RATE_CNT      VARCHAR(2)  NOT NULL default '5',
  RATE_SEC      VARCHAR(3)  NOT NULL default '30',
  RATE_TIME     VARCHAR(110) NULL default '',
  KEY_METHOD    VARCHAR(6)  NOT NULL default 'app512',
  CYCLE_TIME    VARCHAR(2)  NOT NULL default '60',
  SECURE_KEY    VARCHAR(32) NOT NULL default 'j1q1cHbVqdpj7b4PzBpM2Di1eBvmHFV/',
  ACL_TYPE      VARCHAR(5)  NOT NULL default 'deny',
  MIDDLE_TYPE   VARCHAR(14) NOT NULL default 'DISALLOW_REUSE',
  MIDDLE_TIME   VARCHAR(8)   NULL default '',
  ENV_TYPE      VARCHAR(8)  NOT NULL default 'share',
  PRIMARY KEY (HOSTNAME)
) ENGINE = INNODB;

#
# Table structure for table 'TB_HOST_EOTA'
#

CREATE TABLE IF NOT EXISTS TB_HOST_EOTA (
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  EMERGENCY_KEY VARCHAR(8)  NOT NULL default '',
  PRIMARY KEY (HOSTNAME,EMERGENCY_KEY)
) ENGINE = INNODB;

#
# Table structure for table 'TB_BARO_ACL'
#

CREATE TABLE IF NOT EXISTS TB_BARO_ACL (
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  USERNAME      VARCHAR(40) NOT NULL default '',
  PRIMARY KEY (HOSTNAME,USERNAME)
) ENGINE = INNODB;

#
# Table structure for table 'TB_BARO_USER'
#

CREATE TABLE IF NOT EXISTS TB_BARO_USER (
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  USERNAME      VARCHAR(40) NOT NULL default '',
  RATE_CNT      VARCHAR(2)  NOT NULL default '5',
  RATE_SEC      VARCHAR(3)  NOT NULL default '30',
  RATE_TIME     VARCHAR(110) NULL default '',
  KEY_METHOD    VARCHAR(6)  NOT NULL default 'app512',
  CYCLE_TIME    VARCHAR(2)  NOT NULL default '60',
  SECURE_KEY    VARCHAR(32) NOT NULL default 'j1q1cHbVqdpj7b4PzBpM2Di1eBvmHFV/',
  ACL_TYPE      VARCHAR(5)  NOT NULL default 'deny',
  MIDDLE_TYPE   VARCHAR(14) NOT NULL default 'DISALLOW_REUSE',
  MIDDLE_TIME   VARCHAR(8)   NULL default '',
  PRIMARY KEY (HOSTNAME,USERNAME)
) ENGINE = INNODB;

```

```

) ENGINE = INNODB;

#
# Table structure for table 'TB_USER_EOTA'
#

CREATE TABLE IF NOT EXISTS TB_USER_EOTA (
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  USERNAME      VARCHAR(40) NOT NULL default '',
  EMERGENCY_KEY VARCHAR(8)  NOT NULL default '',
  PRIMARY KEY (HOSTNAME,USERNAME,EMERGENCY_KEY)
) ENGINE = INNODB;

#
# Table structure for table 'TB_BARO_LOG'
#

CREATE TABLE IF NOT EXISTS TB_BARO_LOG (
  AUTH_DTTM     VARCHAR(10) NOT NULL default '',
  HOSTNAME      VARCHAR(30) NOT NULL default '',
  USERNAME      VARCHAR(40) NOT NULL default '',
  REMOTE_IP     VARCHAR(30)  NULL default '',
  AUTH_MSG      VARCHAR(200) NOT NULL default '',
  PRIMARY KEY (AUTH_DTTM, HOSTNAME, USERNAME)
) ENGINE = INNODB;
    
```

참고) 스키마 정보

1. 서버 설정 마스터 (TB_BARO_HOST)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	호스트명	HOSTNAME	VAR	30	Y	PK	서버의 호스트명(uname -n)
2	제한횟수	RATE_CNT	VAR	2	Y		인증키의 제한횟수(1~10)
3	제한기간	RATE_SEC	VAR	3	Y		인증키의 제한시간(초, 15~600초)
4	제한시간	RATE_TIME	VAR	110	N		인증키의 제한시간(내부적으로 사용)
5	인증방식	KEY_METHOD	VAR	6	Y		인증키의 인증방식(app1, app256, app384, app512)
6	인증주기	CYCLE_TIME	VAR	2	Y		인증키의 인증주기(초, 3~60초)
7	Secure키	SECURE_KEY	VAR	32	Y		벤더에서 제공하는 Secure key(라이선스 키)
8	ACL구분	ACL_TYPE	VAR	5	Y		2차 인증에서 허용(allow) 또는 제외(deny) 구분
9	중간자공격방어구분	MIDDLE_TYPE	VAR	14	Y		중간자(man-in-the-middle) 공격을 예방할 경우는 "DISALLOW_REUSE"을 설정한 경우 일회용 인증키의 인증주기 동안은 다른 사용자가 로그인 할 수 없으며, 만약 허용할 경우는 "ALLOW_REUSE"을 설정
10	중간자공격방어시간	MIDDLE_TIME	VAR	8	N		중간자공격을 방어할 경우에 사용(내부적으로 사용)
11	환경설정구분	ENV_TYPE	VAR	8	Y		환경설정 정보를 공유(share)해서 사용할 것인지, 사용자별(username)로 설정하지 구분

2. 서버 응급일회용인증키 마스터 (TB_HOST_EOTA)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	호스트명	HOSTNAME	VAR	30	Y	PK,FK	서버의 호스트명(uname -n)
2	응급일회용키	EMERGENCY_KEY	VAR	8	Y	PK	인증키 생성기인 BaroPAM 앱을 사용할 수 없을 때 분실한 경우를 대비하여 SSH 서버에 다시 액세스하는데 사용

3. ACL 설정 마스터 (TB_BARO_ACL)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	호스트명	HOSTNAME	VAR	30	Y	PK,FK	서버의 호스트명(uname -n)
2	계정(사용자)명	USERNAME	VAR	40	Y	PK	서버에 로그인하는 계정

4. 계정(사용자)별 설정 마스터 (TB_BARO_USER)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	호스트명	HOSTNAME	VAR	30	Y	PK,FK	서버의 호스트명(uname -n)
2	계정(사용자)명	USERNAME	VAR	40	Y	PK	서버에 로그인하는 계정
3	제한횟수	RATE_CNT	VAR	2	Y		인증키의 제한횟수(1~10)
4	제한기간	RATE_SEC	VAR	3	Y		인증키의 제한시간(초, 15~600초)
5	제한시간	RATE_TIME	VAR	110	N		인증키의 제한시간(내부적으로 사용)
6	인증방식	KEY_METHOD	VAR	6	Y		인증키의 인증방식(app1, app256, app384, app512)
7	인증주기	CYCLE_TIME	VAR	2	Y		인증키의 인증주기(초, 3~60초)
8	Secure키	SECURE_KEY	VAR	32	Y		벤더에서 제공하는 Secure key(라이선스 키)
9	ACL구분	ACL_TYPE	VAR	5	Y		2차 인증에서 허용(allow) 또는 제외(deny) 구분
10	중간자공격방어구분	MIDDLE_TYPE	VAR	14	Y		중간자(man-in-the-middle) 공격을 예방할 경우는 "DISALLOW_REUSE"을 설정한 경우 일회용 인증키의 인증주기 동안은 다른 사용자가 로그인 할 수 없으며, 만약 허용할 경우는 "ALLOW_REUSE"을 설정
11	중간자공격방어시간	MIDDLE_TIME	VAR	8	N		중간자공격을 방어할 경우에 사용(내부적으로 사용)

5. 계정(사용자)별 응급일회용인증키 마스터 (TB_USER_EOTA)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	호스트명	HOSTNAME	VAR	30	Y	PK,FK	서버의 호스트명(uname -n)
2	계정(사용자)명	USERNAME	VAR	40	Y	PK,FK	서버에 로그인하는 계정
3	응급일회용키	EMERGENCY_KEY	VAR	8	Y	PK	인증키 생성기인 BaroPAM 앱을 사용할 수 없을 때 분실한 경우를 대비하여 SSH 서버에 다시 액세스하는데 사용

6. 인증 로그 마스터 (TB_BARO_LOG)

NO	한글명	영문명	Type	길이	필수	Key	비고
1	인증시간	AUTH_DTTM	VAR	10	Y	PK	국제적인 표준시간
2	호스트명	HOSTNAME	VAR	30	Y	PK,FK	서버의 호스트명(uname -n)
3	사용자명	USERNAME	VAR	40	Y	PK,FK	서버에 로그인하는 계정
4	인증메시지	AUTH_MSG	VAR	200	Y		

6. About BaroPAM



Version 1.0 – Official Release – 2016.12.1
Copyright © Nurit corp. All rights reserved.
<http://www.nurit.co.kr>

제 조 사 : 주식회사 누리아이티
등록번호 : 258-87-00901
대표이사 : 이종일
대표전화 : 02-2665-0119(영업문의/기술지원)
이 메 일 : mc529@nurit.co.kr
주 소 : 서울시 강서구 마곡중앙2로 15, 913호(마곡동, 마곡테크노타워2)