

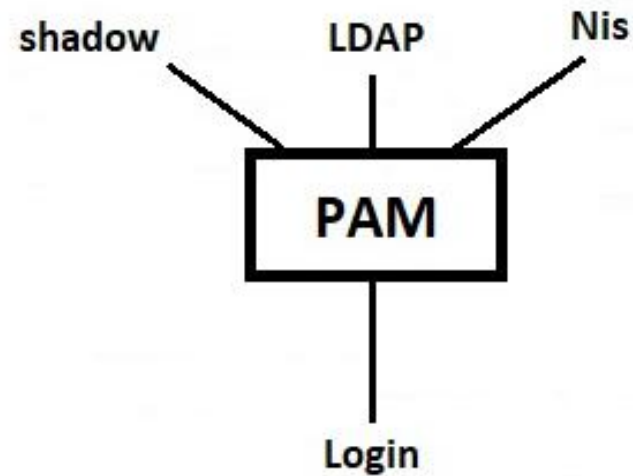
# PAM(Pluggable Authentication Modules)을 이용한 인증관리

# Linux 주요 인증

- 시스템에 로그인해 Shell을 이용하기 위한 인증
- SMTP/POP3/IMAP4 등 메일을 사용하기 위한 인증
- 애플리케이션을 이용하기 위한 인증
- 네트워크 공유 디렉터리에 접근하기 위한 인증

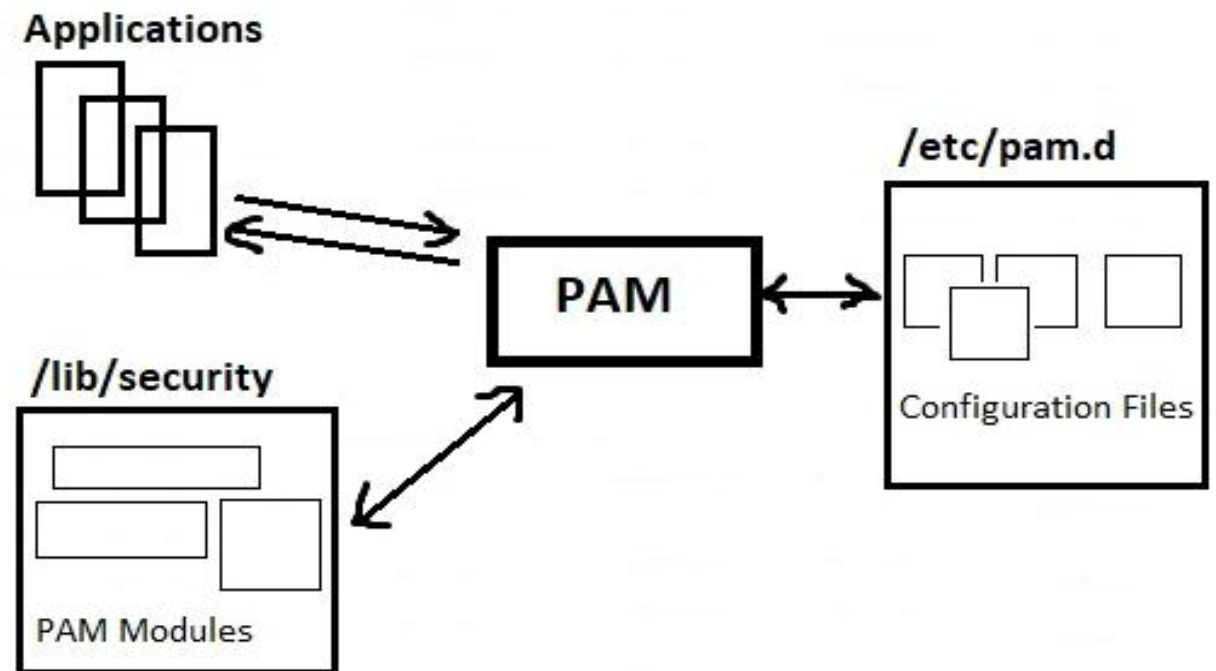
# Linux 인증 방식

- Password 인증
  - LDAP 인증
  - SSH 공개키 인증
- PAM 인증(중앙 집중형 인증 시스템)



## ◆ PAM의 구성

- 라이브러리
  - **/lib64/security**(또는 /lib/security)
- PAM을 이용하는 서비스 디렉터리
  - /etc/pam.d
  - /etc/pam.d/other



인증 설정 파일

/etc/pam.d

/etc/pam.d/remote

```
#auth      required pam_securetty.so
#password  requisite pam_pwquality.so
```

/etc/pam.d/su

```
#auth      required pam_securetty.so
#password  requisite pam_pwquality.so
```

모듈 파일

/lib64/security

pam\_securetty.so

pam\_nologin.so

pam\_wheel.so

pam\_pwquality.so

/etc/securetty

/etc/nologin

/etc/group

/etc/security/pwquality.conf

정책설정파일

## PAM 설정 파일 구성

<u>type</u>	<u>control</u>	<u>module_name</u>	<u>module-arguments</u>
①	②	③	④

auth	include	postlogin
account	required	pam_nologin.so
session	include	system-auth
session	required	pam_namespace.so
password	include	system-auth

/etc/pam.d/login

## ① Type

- 어떤 타입의 인증이 사용될 것인지를 알려주는 항목

type	설명
Account (계정정책)	<p><b>사용자 계정을 확인</b>하는 절차 제공(계정의 접근 통제 및 계정 정책을 관리)</p> <ul style="list-style-type: none"> <li>- 사용자가 해당 서비스에 접근이 허용되는지 여부</li> <li>- 계정 활성화/비활성화 여부 확인</li> <li>- 패스워드 기간 만료 여부를 검사</li> <li>- 특정 시간대에 접속 시도 가능여부 확인</li> </ul>
Auth (비밀번호확인)	<p>사용자 <b>패스워드 유효성 검사</b>와 같은 서비스 인증 절차에 사용</p> <ul style="list-style-type: none"> <li>- 다중 모듈(Kerberos Ticket과 같은 연동도 가능)</li> </ul> <p>사용자를 인증하고 자격증명 절차 제공</p> <ul style="list-style-type: none"> <li>- <b>패스워드를 통해 인증</b>(사용자에게 패스워드를 요청하고 입력받은 정보가 유효한지 검사)</li> </ul>
Password (패스워드 정책)	<p>사용자가 패스워드 등의 인증 방법을 변경하도록 할 때 제공하는 방법</p> <p><b>패스워드 변경 시 최소길이/복잡도 설정 등과 관련(사용자 패스워드를 변경할 수 있도록 비밀번호 갱신 관련)</b></p>
session	<p><b>사용자가 인증 받기 전/후에 해야 할 것을 지정</b></p> <p>홈 디렉터리 마운트/언마운트, 로그인/로그아웃 서비스 제한 등 포함</p>

## ② control

- PAM이 무엇을 해야 할 지를 알려줌

Required (필수)	인증 결과와 관계 없이 <b>다음 인증 수행</b>	·인증결과가 성공일 경우, 다른 모듈들이 실패하지 않은 한 요청 허용 ·인증결과가 실패일 경우, 다른 인증을 수행 후 요청 거부
Requisite (필요)	인증결과가 실패일 경우 <b>즉시 인증 종료</b>	·인증결과가 성공일 경우, 다음 모듈 실행(최종 결과에 미반영) ·인증결과가 실패일 경우, 즉시 인증 실패를 반환(다른모듈실행안함)
Sufficient (충분)	인증결과가 성공일 경우 <b>즉시 인증 종료</b>	·인증결과가 성공일 경우, 즉시 인증 성공 반환 ·인증결과가 실패일 경우, 다음 인증 모듈실행(최종 인증결과에 미반영)
Optional (선택)	<b>최종 인증결과에 반영되지 않음</b>	



## Play with PAM

 required	<b>X</b>	✓	✓	✓
 required	✓	✓	✓	✓
 requisite	✓	<b>X</b>	✓	✓
 optional	✓		<b>X</b>	<b>X</b>
 sufficient	✓		✓	<b>X</b>
 requisite				✓
<b>Results</b>	<b>X</b>	<b>X</b>	✓	✓

**\* /etc/pam.d/system-auth : 로컬 로그인 인증 설정**

```
[root@localhost pam.d]# cat system-auth

auth      required      pam_env.so
auth      sufficient    pam_fprintd.so
auth      sufficient    pam_unix.so nullok try_first_pass
auth      requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth      required      pam_deny.so

account    required      pam_unix.so
account    sufficient    pam_localuser.so
account    sufficient    pam_succeed_if.so uid < 1000 quiet
account    required      pam_permit.so

password   requisite     pam_pwquality.so try_first_pass local_users_only retry=3 authtok_type=
password   sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password   required      pam_deny.so

session    optional      pam_keyinit.so revoke
session    required     pam_limits.so
-session   optional      pam_systemd.so
session    [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session    required     pam_unix.so
```

**\* /etc/pam.d/password-auth : 원격 로그인(ssh, ftp)의 인증 설정**

```
[root@localhost pam.d]# cat password-auth
auth      required      pam_env.so
auth      sufficient    pam_unix.so nullok try_first_pass
auth      requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth      required      pam_deny.so

account   required      pam_unix.so
account   sufficient    pam_localuser.so
account   sufficient    pam_succeed_if.so uid < 1000 quiet
account   required      pam_permit.so

password  requisite     pam_pwquality.so try_first_pass local_users_only retry=3 authtok_type=
password  sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password  required      pam_deny.so

session   optional      pam_keyinit.so revoke
session   required     pam_limits.so
-session  optional      pam_systemd.so
session   [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session   required     pam_unix.so
[root@localhost pam.d]#
```

Debug	시스템 로그 파일에 디버그 정보를 남기도록 지정
No_warn	모듈이 경고 메시지를 보내지 않도록 지정
Use_first_pass	사용자에게 패스워드 입력을 요구하지 않도록 지정 이전모듈에서 입력 받은 패스워드가 존재하지 않을 경우 인증 실패 반환
Try_first_pass	이전모듈에서 입력 받은 패스워드로 인증 시도 이전에 입력 받은 패스워드가 존재하지 않을 경우 사용자 입력 요구

### ③ module\_name

- 사용하는 모듈명을 명기하는 부분
- 절대경로를 입력하거나 /lib/security에 있는 모듈명 기입

### ④ module-arguments

- 지정한 모듈이 사용하는 인수를 기입
- 여러 인수를 사용하는 경우에는 공백으로 구분
- 인수에 공백을 포함시키려면 대괄호([])를 사용해서 묶음

```
[root@localhost pam.d]# head -4 remote
#%PAM-1.0
auth      required      pam_securetty.so
auth      required      pam_faillock.so deny=3 unlock time=60
auth      substack      password-auth
```

auth required pam\_faillock.so deny=3 unlock\_time=60

- 인증을 처리하는데 pam\_faillock.so 모듈 사용
- 실패 3회(deny=3)이면 30초동안 계정 잠금(unlock)

## pam\_wheel.so

- root 권한을 얻을 수 있는 사용자를 wheel(또는 group-ID=0)이라는 그룹으로 묶어서 사용하도록 지원하는 모듈
- su 명령과 관련된 /etc/pam.d/su에 사용하면 매우 유용하다.

argument	설명
debug	디버깅 관련 정보를 출력한다.
group=그룹명	wheel 또는 GID 0번 그룹을 검사하는 대신에 해당 그룹명으로 인증을 수행한다.
deny	모듈의 동작을 반대가 되도록 설정한다. 만약 wheel 그룹에 속한 사용자가 uid=0을 얻는 시도를 하면 접근을 거부한다.
trust	wheel 그룹에 속한 사용자가 root 권한을 요구한 경우 PAM_SUCCESS를 리턴값으로 준다. 즉, wheel 그룹에 속한 사용자들은 암호를 입력하지 않고도 root 권한을 획득할 수 있다.
use_uid	로그인할 때의 사용자명 대신에 현재의 UID를 사용한다. 다른 계정으로 로그인한 뒤에 su 명령을 사용한 경우가 해당된다.
root_only	단지 wheel 그룹에 속한 사용자 여부만 검사한다.

## pam\_listfile.so

- 특정 서비스에 대해 허가 목록이나 거부 목록을 만들 때 사용
- /etc/pam.d/vsftpd 파일에 설정되어 ftp 사용자 거부 목록 파일로 이용

```
[root@www ~]# cat /etc/pam.d/vsftpd
#%PAM-1.0
session    optional    pam_keyinit.so      force revoke
auth       required    pam_listfile.so    item=user sense=deny file=/etc/
vsftpd/ftpusers onerr=succeed
auth       required    pam_shells.so
auth       include     password-auth
account    include     password-auth
session    required    pam_loginuid.so
session    include     password-auth
```

## [모듈 인자(module-argument)]

argument	설명
item=	목록 파일에 이용할 항목을 지정하는데, 사용자인 경우에 item=user로 설정한다. 사용자 이외에도 group, tty, shell, rhost, ruser의 설정이 가능하다.
sense=	목록 파일을 허가 또는 거부로 설정하는 항목이다. 허가이면 allow, 거부이면 deny로 설정한다.
file=	목록 파일의 경로를 지정한다. 해당 파일에 아이템 등록은 한 줄에 하나씩 적어야 한다.
onerr=	succeed 또는 fail이라고 설정하는데, 일반적으로 sense에 설정하는 값의 반대로 지정한다. succeed면 PAM_SUCCESS를 리턴하고, fail이면 PAM_AUTH_ERR 또는 PAM_SERVICE_ERR을 리턴한다.
apply=	특정 사용자(user) 또는 특정 그룹(@group)으로 적용을 제한할 때 사용한다. item 항목이 tty, rhost, shell인 경우에만 의미 있는 제한이 된다.



## PAM의 사용 예

예제 1. 모든 계정에 대해 콘솔(Console) 로그인을 막는다.

① /etc/pam.d/login 파일에 다음의 설정을 추가한다.

```
#cat /etc/pam.d/login  
account required pam_deny.so
```

② 확인 방법

[CTRL]+[ALT]+[F2]키를 눌러서 2번째 터미널을 호출한 뒤에 로그인을 시도해보면 로그인이 되지 않는 것을 알 수 있다.

## 예제 2. 일반 계정 사용자 gildong의 텔넷 로그인을 막는다.

- ① /etc/pam.d/remote 파일에 다음의 설정을 추가

```
# vi /etc/pam.d/remote
auth    required pam_listfile.so item=user sense=deny file=/etc/loginusers
onerr=succeed
```

- ② /etc/loginusers 파일을 생성하고, 한 줄에 한 계정 추가

```
# cat /etc/loginusers
gildong
```

### 예제 3. 일반 계정 사용자 gildong의 ssh 로그인 허용

[설정]

① /etc/pam.d/sshd 파일에 다음의 설정을 추가

```
# vi /etc/pam.d/sshd
```

```
auth    required  pam_listfile.so item=user sense=allow file=/etc/ssh_users onerr=fail
```

② /etc/ssh\_users 파일을 생성하고, 한 줄에 한 계정씩 추가

```
# vi /etc/ssh_users
```

```
gildong
```

예제 4. 사용자 패스워드의 길이를 최소 12자로 설정하고, 새로운 패스워드 입력할 때 LINUX라는 문구열이 출력되도록 설정

① /etc/pam.d/system-auth 파일에 pam\_cracklib.so 항목의 모듈인자 수정

# vi /etc/pam.d/system-auth

password requisite pam\_cracklib.so try\_first\_pass retry=3 authok\_type=LINUX minlen=12

② 확인 예

A terminal window titled 'lin@www:~' showing the execution of the 'passwd' command. The output indicates a successful password change for user 'lin' with the new password 'LINUX'.

```
lin@www:~  
[lin@www ~]$ passwd  
Changing password for user lin.  
Changing password for lin.  
(current) UNIX password:  
New LINUX password:  
Retype new LINUX password:  
passwd: all authentication tokens updated successfully.  
[lin@www ~]$
```

## 예제 5. su 명령어의 사용 gildong 에게만 허용한다.

- ① /etc/group 파일의 wheel 그룹 항목의 4번째 필드에 gildong 를 추가

```
# vi /etc/group
```

```
wheel:x:10: gildong
```

- ② /etc/pam.d/su에 다음의 항목을 추가

```
# vi /etc/pam.d/su
```

```
auth required pam_wheel.so use_uid debug group=wheel
```

## 예제 6. 텔넷 로그인 시 패스워드 입력이 3회 이상 틀리면 3분 동안 로그인 제한

① /etc/pam.d/remote에 다음의 항목을 추가

```
# vi /etc/pam.d/remote
```

```
auth required pam_tally2.so deny=3 unlock_time=180
```

→ 보통 기본 1회에 deny 설정 값을 더해서 패스워드 총 4회까지 입력

② 로그인 시도 후 관련 기록 확인

A terminal window titled 'root@www:~' showing the output of the 'pam\_tally2' command. The output is a table with four columns: 'Login', 'Failures', 'Latest failure', and 'From'. The first row shows 'lin' with 4 failures, the latest failure at '03/07/16 19:46:30', and 'localhost' as the source.

```
root@www:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[root@www ~]# pam_tally2  
Login          Failures Latest failure    From  
lin            4      03/07/16 19:46:30 localhost  
[root@www ~]#
```

→ pam\_tally2 명령으로 관련 정보를 확인

③ 계정 잠금 해제

```
# pam_tally2 -r -u lin
```

→ lin 사용자의 계정 잠금 상태를 해제

## Lab 1. 계정 root로 콘솔 로그인 막기



```
[root@localhost pam.d] # ls gdm-password
gdm-password
[root@localhost pam.d] # cat gdm-password
auth      [success=done ignore=ignore default=bad] pam_selinux_permit.so
auth      substack      password-auth
auth      optional     pam_gnome_keyring.so
auth      include      postlogin

account   required     pam_nologin.so
account   include      password-auth


password  substack     password-auth
password  optional     pam_gnome_keyring.so use_auth_tok

session   required     pam_selinux.so close
session   required     pam_loginuid.so
session   optional     pam_console.so
-session  optional     pam_ck_connector.so
session   required     pam_selinux.so open
session   optional     pam_keyinit.so force revoke
session   required     pam_namespace.so
session   include      password-auth
session   optional     pam_gnome_keyring.so auto_start
session   include      postlogin
[root@localhost pam.d] #
```

```
auth required pam_succeed_if.so uid >= 1000
```

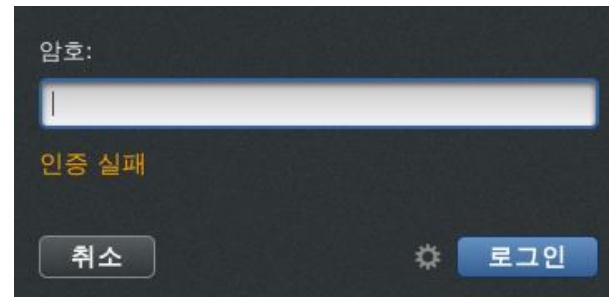
```
[root@localhost pam.d]# cat gdm-password
auth          required          pam_succeed_if.so uid >= 1000
auth          [success=done ignore=ignore default=bad] pam_selinux_permit.so
auth          substack           password-auth
auth          optional          pam_gnome_keyring.so
auth          include            postlogin
```

pam\_succeed\_if.so : UID가 1000일 경우에만 true 값을 반환하는 모듈 실패 시 아래 라인들을 수행함



사용자 이름:

취소 다음



암호:

인증 실패

취소 로그인



## Lab 2. 명령어 su 사용 시 wheel 그룹 멤버들은 패스워드 없이 로그인

```
[ root@localhost pam.d] # pwd
/etc/pam.d
[ root@localhost pam.d] # cat 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.
#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
[ root@localhost pam.d] #
```

```
[gildong@localhost ~]$ whoami
gildong
[gildong@localhost ~]$ su - test01
암호:
마지막 로그인 실패: 일 10월  8 11:33:53 KST 2023 일시 pts/1
[test01@localhost ~]$
[test01@localhost ~]$ whoami
test01
[test01@localhost ~]$
```

```
[root@localhost pam.d]# cat /etc/group | grep wheel
wheel:x:10:
[root@localhost pam.d]#
```

```
auth    sufficient pam_wheel.so trust use_uid
```

```
[root@localhost pam.d]# pwd
/etc/pam.d
[root@localhost pam.d]# cat su
#%PAM- 1.0
```

```
auth            sufficient      pam_wheel.so trust use_uid
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.
#auth           required        pam_wheel.so use_uid
auth            substack        system-auth
auth            include         postlogin
```

```
gpasswd -a gildong wheel
```

```
cat /etc/group | grep wheel
```

```
[root@localhost /]# gpasswd -a gildong wheel
사용자 gildong을(를) wheel 그룹에 등록 중
[root@localhost /]#
[root@localhost /]# cat /etc/group | grep wheel
wheel:x:10:gildong
[root@localhost /]#
```

```
[gildong@localhost /]$ su - test01
마지막 로그인: 일 10월 15 11:09:48 KST 2023 일시 pts/1
마지막 로그인 실패: 일 10월 15 11:18:10 KST 2023
마지막 로그인 후 1 번의 로그인 시도가 실패하였습니다.
[test01@localhost ~]$
```

```
[root@localhost ~]# gpasswd -a root wheel
사용자 root을(를) wheel 그룹에 등록 중
[root@localhost ~]#
[root@localhost ~]# cat /etc/group | grep wheel
wheel:x:10:gildong, root
[root@localhost ~]# exit
logout
[gildong@localhost ~]$ su - root
마지막 로그인: 일 10월 15 11:23:47 KST 2023 일시 pts/1
[root@localhost ~]#
```

## Lab 3. wheel 그룹 멤버들만 명령어 su 사용 ( 사용시 패스워드 입력 필수)

```
auth    required pam_wheel.so use_uid
```

```
[root@localhost pam.d]# cat su
```

```
#%PAM- 1.0
```

```
auth            required            pam_wheel.so use_uid
```

```
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.
```

```
#auth           required            pam_wheel.so use_uid
```

```
auth            substack            system-auth
```

```
auth            include             postlogin
```

```
[gildong@localhost ~]$ cat /etc/group | grep wheel
wheel:x:10:gildong,root
[gildong@localhost ~]$
[gildong@localhost ~]$ su - test01
암호:
마지막 로그인: 일 10월 15 11:44:15 KST 2023 일시 pts/2
[test01@localhost ~]$
[test01@localhost ~]$ whoami
test01
[test01@localhost ~]$ su - test02
암호:
su: 권한 부여 거부
[test01@localhost ~]$ su - root
암호:
su: 권한 부여 거부
[test01@localhost ~]$ exit
logout
[gildong@localhost ~]$ su - root
암호:
마지막 로그인: 일 10월 15 11:24:57 KST 2023 일시 pts/1
마지막 로그인 실패: 일 10월 15 11:45:26 KST 2023 일시 pts/2
마지막 로그인 후 1 번의 로그인 시도가 실패하였습니다.
[root@localhost ~]#
```

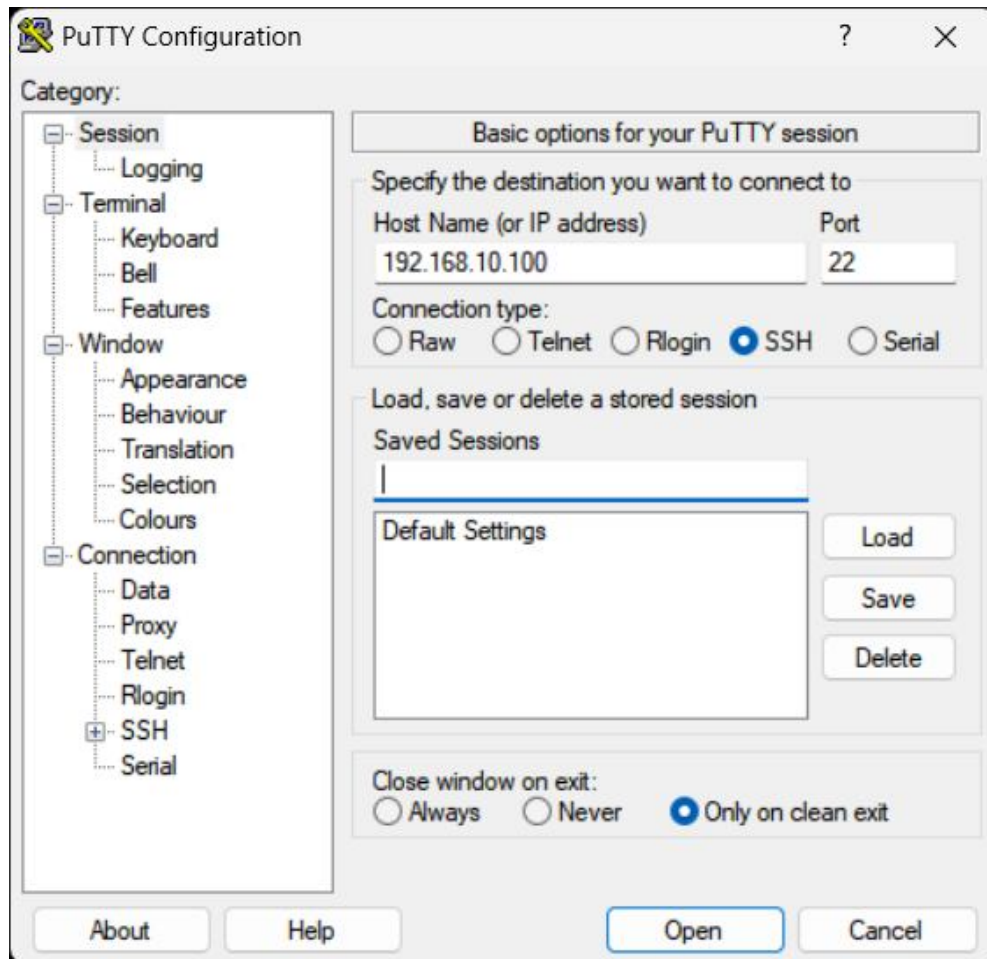
```
cat /etc/group | grep wheel
whoami
su - test01
whoami
su - test02
exit
su - root
```

## Lab 4. 원격 접속 사용 시간 제한

```
[root@localhost pam.d]# pwd
/etc/pam.d
[root@localhost pam.d]# ls sshd
sshd
[root@localhost pam.d]# cat sshd
#%PAM- 1.0
auth      required      pam_sepermit.so
auth      substack      password-auth
auth      include      postlogin
account   required      pam_nologin.so
account   include      password-auth
password  include      password-auth
# pam_selinux.so close should be the first session rule
session   required      pam_selinux.so close
session   required      pam_loginuid.so
# pam_selinux.so open should only be followed by sessions to
text
session   required      pam_selinux.so open env_params
session   optional     pam_keyinit.so force revoke
session   include      password-auth
session   include      postlogin
[root@localhost pam.d]#
```

```
cd /etc/pam.d
cat sshd
```



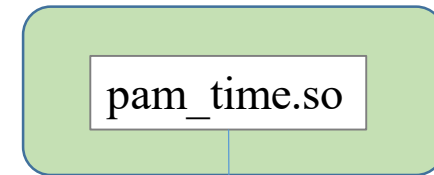


- 원격 접속 사용 시간을 평일 9시 30분부터 17 시 30분으로 제한

```
[root@localhost pam.d]# cat sshd
#%PAM-1.0
account      required      pam_time.so
auth         required      pam_sepermit.so
auth         substack      password-auth
auth         include       postlogin
account      required      pam_nologin.so
account      include       password-auth
password     include       password-auth
```

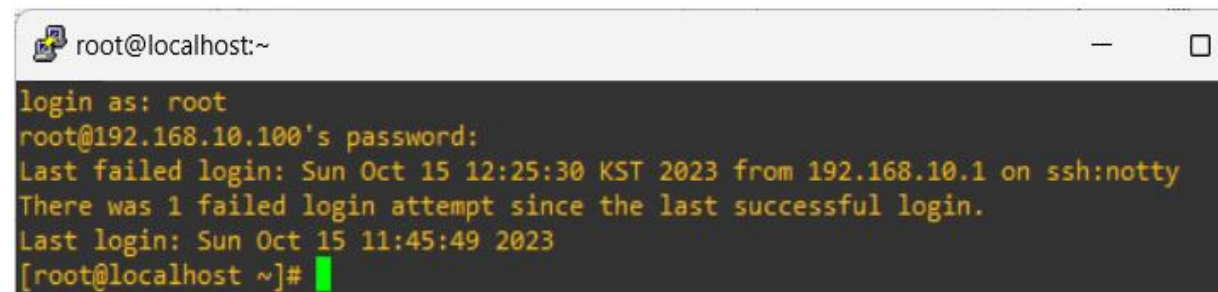
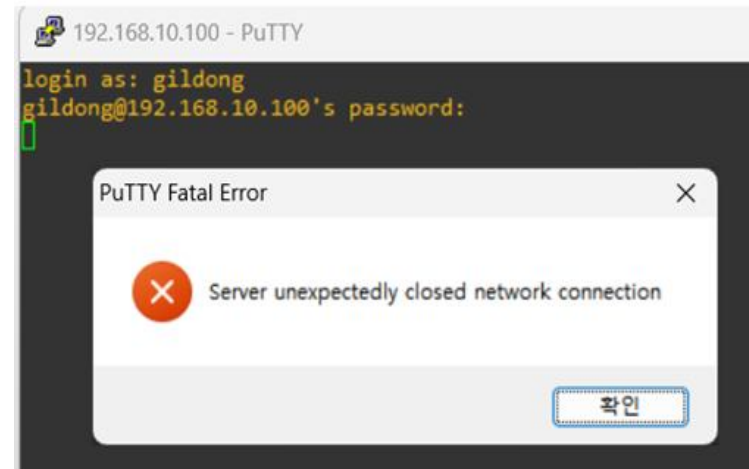
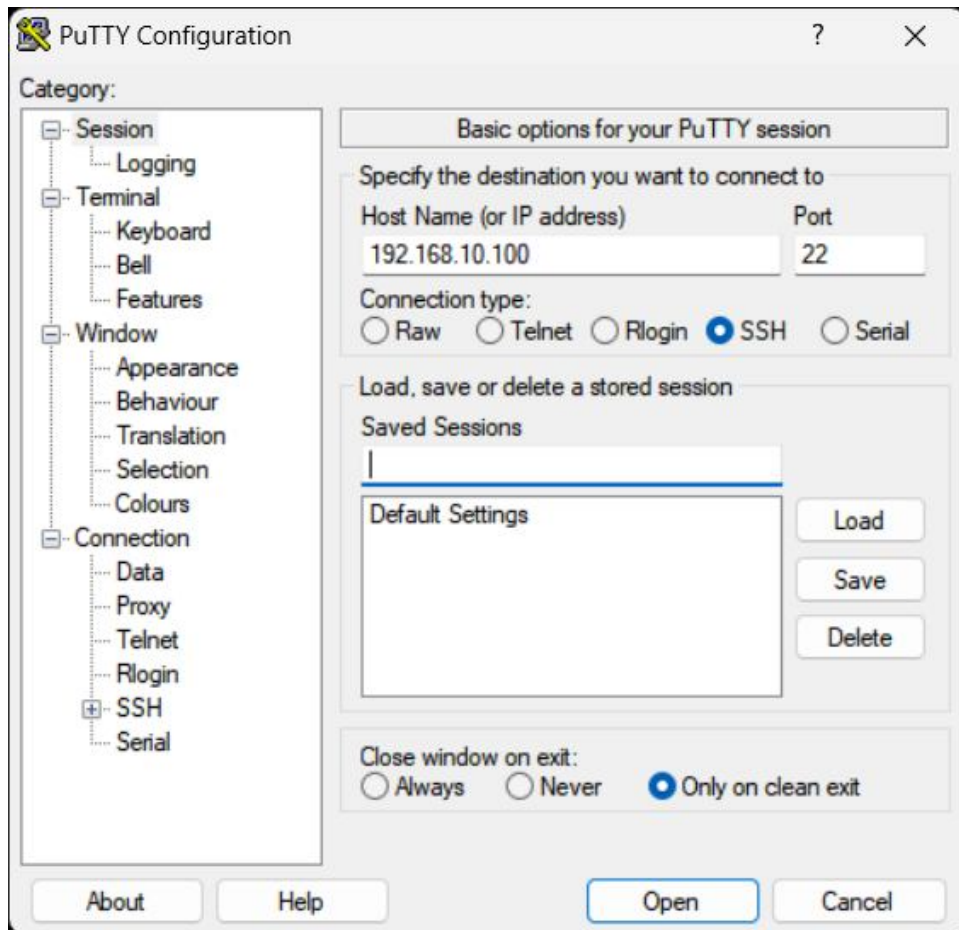
```
[root@localhost security]# pwd
/etc/security
[root@localhost security]# ls time.conf
time.conf
[root@localhost security]# tail -4 time.conf
# End of example file.
#
sshd; *; gildong; !Wd0930-1730
sshd; *; root; Wd0930-1730
[root@localhost security]#
```

**/lib64/security**



/etc/security/time.conf

```
sshd; *; gildong; !wk0930-17:30
sshd; *; root; wk0930-17:30
```



## Lab 5. 콘솔로그인 시 패스워드 정책 설정



gildong

암호:

2 로그인 실패로 인해 계정이 잠김

취소 로그인



gildong

암호:

일시적으로 계정이 잠금되었습니다 (5 초 남음)

취소 로그인

```
#vi /etc/pam.d/password-auth  
auth    required pam_tally2.so unlock_time=10  
account required pam_tally2.so
```

<<확인>>

```
#pam_tally2
```

```
#pam_tally2 -r -u gildong
```


```
#pam_tally2 -r
```

## pam\_tally2.so

argument	설명
deny= <i>N</i>	로그인 시도가 <i>N</i> 번 실패하면 접근을 차단
lock_time= <i>N</i>	로그인 실패 후에 <i>N</i> 초 동안 접근을 차단
unlock_time= <i>N</i>	관리자가 정한 일정 횟수 이상 로그인에 실패했을 경우 <i>N</i> 초 동안 접근을 차단 해당 시간 동안은 관리자가 계정을 해제하기 전까지는 계정잠김
root_unlock_time= <i>N</i>	root 사용자가 일정 횟수 이상 로그인에 실패했을 경우 <i>N</i> 초 동안 접근 차단
file=경로	카운트 내역을 기록하는 파일 경로를 기록 기본 파일명은 /var/log/tallylog
no_log_info	syslog에 메시지를 전달하지 않음
silent	관련 정보를 출력하지 않음

```
[root@localhost /]# su root
[root@localhost /]# exit
exit
[root@localhost /]# exit
logout
[gildong@localhost 바탕화면]$ su root
암호:
```

auth	sufficient	pam_rootok.so
auth	substack	system-auth



auth	required	pam_env.so
auth	sufficient	pam_fprintd.so
auth	sufficient	pam_unix.so nullok try_first_pass
auth	requisite	pam_succeed_if.so uid >= 1000 quiet_success
auth	required	pam_deny.so