목차

Last update: 2018/12/04 06:30	보안취약점 점검	https://atl.kr/dokuwiki/doku.php/%EB%B3%B4%EC%95%88%EC%B7%A8%EC%95%BD%EC%A0%90_%EC%A0%90%EA%B2%80?rev=1543905011

보안취약점 점검

— *이강우 2018/10/22 11:06*

RHEL 사용시 일반적인 보안 취약점 점검 사항 조치를 위한 조치방안에 대해 안내드리는 페이지 입니다.

아래 사항은 각 RHEL 버전에서 테스트 된 것이며 참조링크를 확인하시면 더욱 정확하게 사용하실 수 있습니다. 모든 내용에 대한 문의는 koovis@gmail.com으로 문의 바랍니다.

root계정 원격 접속 제한

- 위험도 : 상

조치방안

모든 콘솔 root 로그인 차단

콘솔 root 로그인 차단을 하게되면 물리 장비의 console에서도 root로그인이 불가능해진다. 방법은 /etc/securetty 의 내용을 모두 삭제하면 된다.

```
echo > /etc/securetty
```

이 기능을 적용하려면 아래의 내용을 관련 설정에 적용하여야 한다.

```
auth [user_unknown=ignore success=ok ignore=ignore default=bad]
pam_securetty.so
```

위 설정을 적용하기 위한 파일들은 아래와 같다.

```
/etc/pam.d/gdm
/etc/pam.d/gdm-autologin
/etc/pam.d/gdm-fingerprint
/etc/pam.d/gdm-password
/etc/pam.d/gdm-smartcard
/etc/pam.d/kdm
/etc/pam.d/kdm-np
/etc/pam.d/xdm
```

위 사항을 적용한다고 하여도 원격에서 root권한 접속(SSH)을 막을수는 없다. 원격 root 접속을 차단하기 위해서는 아래의 SSH root login 차단방법을 참조한다.

root SSH login 비활성화

/etc/ssh/sshd_config 파일에 아래 설정을 적용하도록 한다.

```
PermitRootLogin no
```

참조링크

- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/chap-Security_Guide-Securing_Your_Network.html#sect-Security_Guide-Workstation_Security-Administrative_Controls
- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/sec-Controlling_Root_Access.html#sec-Disallowing_Root_Access

Password complexity 암호 복잡성 설정

인터넷에 여러가지 방법에 대해 안내된 문서가 많지만 대부분이 맞지 않으며 제대로 된 설정이 아니다. 잘못설정하는경우 정책이 적용되지 않거나 문제가 발생하므로 아래 내용을 보고 정확하게 따라서 적용하기 바란다.

- 위험도 : 상

조치방안

RHEL6

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        sufficient    pam_unix.so try_first_pass nullok
auth        required      pam_deny.so

account     required      pam_unix.so

password    requisite     pam_cracklib.so try_first_pass retry=3 type=
minlen=8 minclass=4        # <-- 수정
password    sufficient    pam_unix.so try_first_pass use_authtok nullok
sha512 shadow
password    required      pam_deny.so
```

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

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        sufficient    pam_unix.so try_first_pass nullok
auth        required      pam_deny.so

account     required      pam_unix.so

password    requisite     pam_cracklib.so try_first_pass retry=3 type=
minlen=8 minclass=4     # <-- 수정
password    sufficient    pam_unix.so try_first_pass use_authtok nullok
sha512 shadow
password    required      pam_deny.so

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

위 두부분에만 적용하고 /etc/pam.d/passwd 는 수정하지 않는다. passwd 정책까지 변경하면 중복 설정되어 암호 변경시 3회 물어본다. pam_cracklib 관련 옵션은 아래 페이지 참조

- https://linux.die.net/man/8/pam_cracklib

RHEL7

RHEL7에서는 pam_cracklib 대신에 pam_pwquality 모듈이 사용된다. 기본적으로 적용되어있으며 정책파일만 수정해주면 된다. 복잡성 옵션에 대한 설정은 /etc/security/pwquality.conf 파일을 변경하여야 한다.

```
minlen = 8
minclass = 4
```

또는 아래와 같이 authconfig 명령어를 이용해도 된다.

```
# authconfig --passminlen=<number> --passminclass=<number> --
passmaxrepeat=<number>
```

- passminlen=<number> 최소 길이
- passminclass=<number> 최소 문자 종류(대문자,소문자,숫자,특수문자)
- passmaxrepeat=<number> 연속으로 쓸수있는 동일한 문자의 최대갯수 예) PasssssssWord 에서 s의 갯수
- passmaxclassrepeat=<number> 연속으로 쓸수있는 동일한 종류 문자 최대갯수 예) P@ssword 에서 ssword 는 연속된 같은 소문자

옵션에 대한 설정은 pwquality.conf를 참조. 또는 man authconfig

위 설정은 root사용자는 적용받지 않는다. root 사용자도 강제적으로 적용하기 위해서는 아래 내용을 추가로 설정하여야 한다.

root 사용자 pwquality 적용방법

root 사용자도 pwquality 적용을 하기 위해서는 /etc/pam.d/system-auth 파일과 /etc/pam.d/password-auth 파일의 내용을 수정하여야 한다. 해당 파일내에 local_users_only 라고 되어있는 부분을 enforce_for_root 로 변경해주면 된다.

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faildelay.so delay=2000000
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=  <-- 기존 내용
password    requisite     pam_pwquality.so try_first_pass enforce_for_root
retry=3 authtok_type=   <-- 이부분을 수정한다.
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
```

참조링크

- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/chap-Security_Guide-Securing_Your_Network.html#sect-Security_Guide-Workstation_Security-Password_Security
- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/chap-Hardening_Your_System_with_Tools_and_Services.html#sec-Password_Security

계정 잠금 임계값 설정

- 위험도 : 상

조치방안

무작위 대입 시도시 일정 횟수 이상 실패하면 차단하는 설정

RHEL 5.3 이전버전

pam.tally 모듈을 사용하여 적용한다. `/etc/pam.d/system-auth` 파일의 맨 윗부분에 아래의 내용을 추가한다.

```
auth        required      pam_tally.so deny=3 onerr=fail unlock_time=1200
no_magic_root
account     required      pam_tally.so no_magic_root reset
```

RHEL 5.4 이후버전

pam.tally2 모듈을 사용하여 적용한다. `/etc/pam.d/system-auth` 파일의 맨 윗부분에 아래의 내용을 추가한다.

```
auth        required      pam_tally2.so deny=3 onerr=fail unlock_time=1200
no_magic_root
```

RHEL 6

RHEL 6 부터는 `pam_faillock` 모듈을 사용한다.

1. root 사용자를 제외한 모든 사용자에 대해서 3회 이상 실패시 600초(10분) 동안 차단하는 설정이다. /etc/pam.d/system-auth 파일과 /etc/pam.d/password-auth 파일의 auth 섹션에 아래 내용을 추가한다.

```
auth        required      pam_faillock.so preauth silent audit deny=3
unlock_time=600
auth        sufficient    pam_unix.so nullok try_first_pass
auth        [default=die] pam_faillock.so authfail audit deny=3
unlock_time=600
```

2. account 섹션에도 아래 내용을 추가한다.

```
account     required      pam_faillock.so
```

3. root사용자도 동일하게 적용하기 위해서는 1번 항목 적용시 even_deny_root 속성을 추가하도록 한다.

```
auth        required      pam_faillock.so preauth silent audit deny=3
even_deny_root unlock_time=600
auth        sufficient    pam_unix.so nullok try_first_pass
auth        [default=die] pam_faillock.so authfail audit deny=3
even_deny_root unlock_time=600

account     required      pam_faillock.so
```

4. 로그조회는 faillock 명령어를 통해 조회 가능함.

```
[root@localhost ~]# faillock
john:
When                Type  Source
Valid
2013-03-05 11:44:14 TTY   pts/0
V
```

5. 차단된 사용자를 즉시 해제하려면 아래의 명령어를 사용하도록 한다.

```
faillock --user <username> --reset
```

RHEL 7

/etc/pam.d/system-auth 파일과 /etc/pam.d/password-auth 파일에 아래처럼 수정하도록 한다.

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth silent audit deny=3
unlock_time=600   # 추가
auth        sufficient    pam_unix.so nullok try_first_pass
auth        [default=die] pam_faillock.so authfail audit deny=3
unlock_time=600     # 추가
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so

account     required      pam_faillock.so                       # 추가
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
```

참조링크

- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/chap-Security_Guide-Securing_Your_Network.html#sect-Security_Guide-Workstation_Security-Administrative_Controls
- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/chap-Hardening_Your_System_with_Tools_and_Services.html#sect-Security_Guide-Workstation_Security-Account_Locking

su 사용 제한

- 위험도 : 하

조치방안

wheel 그룹에 속한 사용자만 su명령어를 사용할 수 있도록 설정

```
#%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          # 주석해
제 시 해당 그룹사용자는 무조건 신뢰(패스워드확인 안함)
# Uncomment the following line to require a user to be in the "wheel" group.
#auth           required        pam_wheel.so use_uid                # 주석해
제 시 해당 그룹사용자는 패스워드 확인 후 신뢰
auth            include         system-auth
account         sufficient      pam_succeed_if.so uid = 0 use_uid quiet
account         include         system-auth
password        include         system-auth
session         include         system-auth
session         optional        pam_xauth.so
```

상황에 따라 trust 옵션을 사용할 지 결정하도록 한다.

/etc/login.defs

login.defs 설정은 shadow-utils 패키지에 포함된 명령어에만 유효한 설정임을 명심한다. 해당 패키지에 포함된 명령어는 아래와 같다.

- /etc/default/useradd
- /etc/login.defs
- /usr/bin/chage
- /usr/bin/gpasswd
- /usr/bin/lastlog
- /usr/bin/newgrp
- /usr/bin/sg
- /usr/sbin/adduser
- /usr/sbin/chpasswd
- /usr/sbin/groupadd
- /usr/sbin/groupdel
- /usr/sbin/groupmems
- /usr/sbin/groupmod
- /usr/sbin/grpck
- /usr/sbin/grpconv

- /usr/sbin/grpunconv
- /usr/sbin/newusers
- /usr/sbin/pwck
- /usr/sbin/pwconv
- /usr/sbin/pwunconv
- /usr/sbin/useradd
- /usr/sbin/userdel
- /usr/sbin/usermod
- /usr/sbin/vigr
- /usr/sbin/vipw

따라서 `/usr/bin/passwd` 같이 패스워드 변경 명령어는 login.defs 의 내용에 영향을 받지 않는다. 이 부분은 `pam.d` 모듈과 관련이 있다

- https://access.redhat.com/solutions/66322

패스워드 최소 길이 설정

- 위험도 : 중

`/etc/login.defs` 파일은 새로운 사용자를 생성하는 경우에만 적용된다. 이미 만들어진 사용자들에 대한 정책은 pam.d 를 사용하여야 한다. pam.d 설정 없이 login.defs 내용만 변경한다고 해서 적용되지 않는다. 본 문서 위쪽의 암호 복잡성 설정 부분 참조.

- https://access.redhat.com/solutions/656833

조치방안

/etc/login.defs 파일에 패스워드 관련 기본 설정 변경

```
# Password aging controls:
#
#       PASS_MAX_DAYS   Maximum number of days a password may be used.
#       PASS_MIN_DAYS   Minimum number of days allowed between password
changes.
#       PASS_MIN_LEN    Minimum acceptable password length.
#       PASS_WARN_AGE   Number of days warning given before a password
expires.
#
PASS_MAX_DAYS   99999       # 패스워드 최대 사용기간 (일반적으로 기간을 제한하는것보다 복잡성을 설정하는것이 보안에 좋음)
PASS_MIN_DAYS   0
PASS_MIN_LEN    9           # 9글자 이상으로 변경
PASS_WARN_AGE   7
```

Session Timout 시간 설정

- 위험도 : 하

조치방안

/etc/profile 내에 TMOUT환경 변수 설정 아울러 히스토리

```
HOSTNAME=`/usr/bin/hostname 2>/dev/null`
HISTSIZE=5000                                   # 히스토리 저장 갯수 5000으로 변경
HISTTIMEFORMAT="%F %T "                         # 히스토리 저장시 날짜 시간 같이 저장
TMOUT=300                                       # 세션 타임아웃 시간 설정
if [ "$HISTCONTROL" = "ignorespace" ] ; then
    export HISTCONTROL=ignoreboth
else
    export HISTCONTROL=ignoredups
fi

export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL HISTTIMEFORMAT
TMOUT  # 환경변수 적용
```

참조링크

- https://linux-audit.com/configure-the-minimum-password-length-on-linux-systems/