목차

kickstart 자동 설치 구성

ISO 자동설치 구성방법

먼저 일반 Legacy 부팅일 경우에 대해서 적용한다. /isolinux/isolinux.cfg 내용 수정 커널 파라메터에 inst.ks=cdrom:/mysettings.ks 줄 추가 mysettings.ks 는 미리 작성한 kickstart 파일이며 ISO의 / 경로에 존재하여야 한다.

```
label linux
  menu label ^Install Rocky Linux 9.5
  kernel vmlinuz
  append initrd=initrd.img inst.stage2=hd:LABEL=Rocky-9-5-x86_64-dvd quiet
inst.ks=cdrom:/mysettings.ks
```

다음으로 EFI 부팅일 경우에는 아래 파일을 수정한다. /EFI/BOOT/grub.cfg 파일 수정 커널 파라메터에 inst.ks=cdrom:/mysettings.ks 줄 추가 mysettings.ks 는 미리 작성한 kickstart 파일이며 ISO의 / 경로에 존재하여야 한다.

```
### BEGIN /etc/grub.d/10_linux ###
menuentry 'Install Rocky Linux 9.5' --class fedora --class gnu-linux --class
gnu --class os {
    linuxefi /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=Rocky-9-5-x86_64-
dvd quiet inst.ks=cdrom:/mysettings.ks
    initrdefi /images/pxeboot/initrd.img
}
```

보안취약점 적용된 kickstart 파일

아래는 보안취약점을 적용한 kickstart 파일이다.

```
#############################################################
# LinuxDataSystem Secured kickstart file                    #
# Version  : v1.1                                           #
# Date     : 2025-06-10                                     #
# Author   : kwlee2@linuxdata.co.kr                         #
#############################################################

# Keyboard layouts
keyboard --xlayouts='us'
# System language
lang en_US.UTF-8 --addsupport=ko_KR.UTF-8
```

```
# SELinux configuration
selinux --disabled

%packages
@^minimal-environment
@Core
@Standard
vim
unzip
wget
psmisc
sysstat
net-tools
tar
bash-completion
telnet
bind-utils
traceroute
tuned
chrony
nfs-utils
%end

# System timezone
timezone Asia/Seoul --utc
#timesource --ntp-disable

# Root password
#rootpw --iscrypted
$6$BtR.Du9aLQ4TTwQw$.lyVLOQolJrZGa4I4nlk6NqZHFUTzPWc9wrn4r4JFGEUkGWj5cYsaJcI
QgfuaG5H8M2VFbeC/YknXjYHdcr/p1
#user --groups=wheel --name=lds --
password=$6$Jaa4gvCGp1yD9Zcy$oKUWX/6xr/IRXV4UZJjeBOTqyvdJTn518fDtD/eDZK.z5yW
hQiqe/M5Dq7rvR4JFei0qblfSAdjRmI0U.7dlA0 --iscrypted --gecos="lds"

%post --log=/root/kickstart-post.log

########################################
# Disable and stop firewalld
systemctl stop firewalld
systemctl disable firewalld
systemctl mask firewalld
echo "Firewalld service has been stopped and disabled."

echo "Starting RHEL 9 Security Hardening Post-Install Script"

# --- 2.1. 계정 및 접근 통제 ---
echo "Applying Account and Access Control settings..."

# 2.1.2. 패스워드 복잡성 설정
```

```
# 기본 주석 처리된 값을 변경하거나, 라인이 없으면 추가합니다.
# 예시 값이며, 실제 보안 정책에 맞게 조정하십시오.
if grep -q "^# minlen" /etc/security/pwquality.conf; then
    sed -i 's/^# minlen =.*/minlen = 9/' /etc/security/pwquality.conf
elif ! grep -q "^minlen" /etc/security/pwquality.conf; then
    echo "minlen = 9" >> /etc/security/pwquality.conf
fi

if grep -q "^# dcredit" /etc/security/pwquality.conf; then
    sed -i 's/^# dcredit =.*/dcredit = -1/' /etc/security/pwquality.conf
elif ! grep -q "^dcredit" /etc/security/pwquality.conf; then
    echo "dcredit = -1" >> /etc/security/pwquality.conf
fi

if grep -q "^# ucredit" /etc/security/pwquality.conf; then
    sed -i 's/^# ucredit =.*/ucredit = -1/' /etc/security/pwquality.conf
elif ! grep -q "^ucredit" /etc/security/pwquality.conf; then
    echo "ucredit = -1" >> /etc/security/pwquality.conf
fi

if grep -q "^# lcredit" /etc/security/pwquality.conf; then
    sed -i 's/^# lcredit =.*/lcredit = -1/' /etc/security/pwquality.conf
elif ! grep -q "^lcredit" /etc/security/pwquality.conf; then
    echo "lcredit = -1" >> /etc/security/pwquality.conf
fi

if grep -q "^# ocredit" /etc/security/pwquality.conf; then
    sed -i 's/^# ocredit =.*/ocredit = -1/' /etc/security/pwquality.conf
elif ! grep -q "^ocredit" /etc/security/pwquality.conf; then
    echo "ocredit = -1" >> /etc/security/pwquality.conf
fi

# 2.1.3. 계정 잠금 임계값 설정
# 시스템 환경에 맞는 프로파일 선택 (예: sssd)
authselect select sssd --force
# faillock 기능 활성화
authselect enable-feature with-faillock

# /etc/security/faillock.conf 설정
# deny: 잠금 임계값 (예: 5회)
if grep -q "^deny\s*=" /etc/security/faillock.conf; then
  sed -i 's/^deny\s*=.*/deny = 5/' /etc/security/faillock.conf
else
  echo "deny = 5" >> /etc/security/faillock.conf
fi

# unlock_time: 잠금 해제 시간(초) (예: 600초 = 10분)
if grep -q "^unlock_time\s*=" /etc/security/faillock.conf; then
  sed -i 's/^unlock_time\s*=.*/unlock_time = 600/'
/etc/security/faillock.conf
else
```

```
  echo "unlock_time = 600" >> /etc/security/faillock.conf
fi

# even_deny_root: root 계정에도 잠금 정책 적용
if ! grep -q "^even_deny_root" /etc/security/faillock.conf; then
 # 주석 처리된 even_deny_root가 있다면 주석 해제, 없다면 추가
  if grep -q "^#\s*even_deny_root" /etc/security/faillock.conf; then
    sed -i 's/^#\s*even_deny_root/even_deny_root/'
/etc/security/faillock.conf
  else
    echo "even_deny_root" >> /etc/security/faillock.conf
  fi
fi

# silent: 잠금 관련 메시지를 사용자에게 자세히 알리지 않음 (선택 사항)
if ! grep -q "^silent" /etc/security/faillock.conf; then
  if grep -q "^#\s*silent" /etc/security/faillock.conf; then
    sed -i 's/^#\s*silent/silent/' /etc/security/faillock.conf
  else
    echo "silent" >> /etc/security/faillock.conf
  fi
fi



# 2.1.6. root 계정 su 제한
# /etc/pam.d/su 파일에서 pam_wheel.so 활성화
# 해당 라인이 이미 존재하고 주석 처리되어 있다면 주석 해제
if grep -q "^#\s*auth\s*required\s*pam_wheel.so use_uid" /etc/pam.d/su; then
  sed -i 's/^#\s*\(auth\s*required\s*pam_wheel.so use_uid\)/\1/'
/etc/pam.d/su
# 해당 라인이 없다면 추가 (일반적으로 include 라인들 위에 추가)
elif ! grep -q "^\s*auth\s*required\s*pam_wheel.so use_uid" /etc/pam.d/su;
then
  sed -i '/^auth\s*include\s*system-auth/i auth            required
pam_wheel.so use_uid' /etc/pam.d/su
fi

# /usr/bin/su 파일 권한 조정
chgrp wheel /usr/bin/su
chmod 4750 /usr/bin/su



# 2.1.7. 패스워드 최소 길이 설정 (login.defs)
# /etc/login.defs 파일의 PASS_MIN_LEN 값을 9로 설정 (pwquality.conf의 minlen과 일
치 권장)
if grep -q "^PASS_MIN_LEN" /etc/login.defs; then
  sed -i 's/^PASS_MIN_LEN.*/PASS_MIN_LEN    9/' /etc/login.defs
else
  echo "PASS_MIN_LEN    9" >> /etc/login.defs
fi
```

```
# 2.1.8. 패스워드 최대 사용기간 설정 (login.defs)
# /etc/login.defs 파일의 PASS_MAX_DAYS 값을 90으로 설정
if grep -q "^PASS_MAX_DAYS" /etc/login.defs; then
  sed -i 's/^PASS_MAX_DAYS.*/PASS_MAX_DAYS   90/' /etc/login.defs
else
  echo "PASS_MAX_DAYS   90" >> /etc/login.defs
fi

# (선택 사항) Kickstart로 생성된 일반 사용자에게 즉시 적용 (UID 1000 이상)
# for user in $(awk -F: '($3 >= 1000 && $1 != "nobody" && $1 != "nfsnobody")
{print $1}' /etc/passwd); do
#   chage -M 90 $user
# done

# 2.1.9. 패스워드 최소 사용기간 설정 (login.defs)
# /etc/login.defs 파일의 PASS_MIN_DAYS 값을 1로 설정
if grep -q "^PASS_MIN_DAYS" /etc/login.defs; then
  sed -i 's/^PASS_MIN_DAYS.*/PASS_MIN_DAYS   1/' /etc/login.defs
else
  echo "PASS_MIN_DAYS   1" >> /etc/login.defs
fi

# (선택 사항) Kickstart로 생성된 일반 사용자에게 즉시 적용 (UID 1000 이상)
# for user in $(awk -F: '($3 >= 1000 && $1 != "nobody" && $1 != "nfsnobody")
{print $1}' /etc/passwd); do
#   chage -m 1 $user
# done

# 2.1.15. Session Timeout 설정
# /etc/profile.d/custom_env.sh 파일 생성
cat <<'EOF' > /etc/profile.d/custom_env.sh
#!/bin/sh
# Shell history settings

# Session Timeout
export TMOUT=600
# readonly TMOUT

# 히스토리 파일에 저장할 명령어 개수
export HISTSIZE=5000

# 히스토리 목록에 표시할 시간 형식 (예: 2025-06-06 15:30:00 )
# 마지막의 공백은 명령어와 시간 구분을 위해 중요합니다.
export HISTTIMEFORMAT='%F %T '

# 히스토리 기록 제어
# ignoreboth: 바로 이전에 실행한 명령어와 동일하거나,
#         히스토리 목록 전체에 이미 있는 명령어가 중복 저장되는 것을 방지합니다.
# export HISTCONTROL=ignoreboth
EOF

# 생성된 스크립트 파일에 실행 권한 부여
```

```
chmod 644 /etc/profile.d/custom_env.sh

# --- 2.2. 파일시스템, 디렉토리 및 권한 ---
echo "Applying Filesystem, Directory, and Permission settings..."

# 2.2.5. /etc/hosts 파일 소유자 및 권한 설정
#chown root:root /etc/hosts
#chmod 644 /etc/hosts

# 2.2.6. /etc/rsyslog.conf 파일 소유자 및 권한 설정
chown root:root /etc/rsyslog.conf
chmod 640 /etc/rsyslog.conf
# find /etc/rsyslog.d/ -type f -exec chmod 640 {} \;
# find /etc/rsyslog.d/ -type f -exec chown root:root {} \;

# 2.2.8. SUID, SGID 파일 점검 (at 명령어 SUID 제거는 선택적)
# /usr/bin/at 의 SUID 제거 (atd 서비스를 사용하지 않거나, 사용을 엄격히 제한할 경우에만 고려)
# if [ -f /usr/bin/at ]; then
#   chmod u-s /usr/bin/at
# fi

# /sbin/unix_chkpwd 는 SUID 비트를 반드시 유지해야 합니다. 조치 없음.
# /usr/bin/newgrp 는 RHEL 9 기본적으로 SUID 비트가 없습니다. 조치 없음.

# 추가적으로, 시스템 전반의 불필요한 SUID/SGID 파일을 찾아 기록하거나 알릴 수 있습니다.
# (자동 제거는 위험하므로 권장하지 않음)
# echo "Searching for SUID/SGID files..." > /root/suid_sgid_check.log
# find / -xdev \( -perm -4000 -o -perm -2000 \) -type f -ls >>
/root/suid_sgid_check.log 2>/dev/null

# 2.2.12. UMASK 설정 관리 (RHEL 9 기본 0022, 변경 시 주석 해제)
# RHEL 9 기본 umask는 0022이므로, 0022를 목표로 한다면 별도 조치 불필요.
# 만약 더 엄격한 umask 027 로 변경하고자 할 경우:
# echo "umask 027" > /etc/profile.d/custom_umask.sh
# chmod +x /etc/profile.d/custom_umask.sh
#
# # /etc/login.defs 파일의 UMASK 값도 변경 (pam_umask 모듈에 영향)
# if grep -q "^UMASK" /etc/login.defs; then
#   sed -i 's/^UMASK.*/UMASK           027/' /etc/login.defs
# else
#   echo "UMASK           027" >> /etc/login.defs
# fi

# 2.2.16. cron 파일 소유자 및 권한 설정
# 주요 cron 파일 및 디렉토리 권한 설정
chown root:root /etc/crontab
chmod 600 /etc/crontab

chown -R root:root /etc/cron.d
```

```
chmod -R 700 /etc/cron.d
find /etc/cron.d/ -type f -exec chmod 600 {} \;

chown -R root:root /etc/cron.hourly
chmod -R 700 /etc/cron.hourly
find /etc/cron.hourly/ -type f -exec chmod 600 {} \; # 또는 700으로 실행 가능하게

chown -R root:root /etc/cron.daily
chmod -R 700 /etc/cron.daily
find /etc/cron.daily/ -type f -exec chmod 700 {} \; # 실행 가능해야 함

chown -R root:root /etc/cron.weekly
chmod -R 700 /etc/cron.weekly
find /etc/cron.weekly/ -type f -exec chmod 700 {} \; # 실행 가능해야 함

chown -R root:root /etc/cron.monthly
chmod -R 700 /etc/cron.monthly
find /etc/cron.monthly/ -type f -exec chmod 700 {} \; # 실행 가능해야 함

# crontab 명령어 사용 제어
rm -f /etc/cron.deny
echo "root" > /etc/cron.allow
# 필요한 다른 사용자 추가 가능
# echo "adminuser" >> /etc/cron.allow
chown root:root /etc/cron.allow
chmod 600 /etc/cron.allow

# /usr/bin/crontab 권한 조정 (SUID 유지, other 권한 제거)
chown root:root /usr/bin/crontab
chmod 4750 /usr/bin/crontab

# --- 2.3. 서비스 및 네트워크 ---
echo "Applying Service and Network settings..."

# 2.3.2. $HOME/.rhosts, hosts.equiv 사용 금지
# 시스템 전체에서.rhosts 파일 검색 및 삭제
find /home -name .rhosts -type f -delete
find /root -name .rhosts -type f -delete # root 홈도 확인

# /etc/hosts.equiv 파일 삭제
rm -f /etc/hosts.equiv

# 2.3.3. 접속 IP 및 포트 제한 (firewalld) - Kickstart 명령어 섹션에서 기본 설정 권장
# Kickstart 명령어 섹션 (예시)
# firewall --enabled --service=ssh
# 또는 더 제한적으로, 특정 zone에만 ssh 허용하고 기본 zone은 drop (CIS 권장)
# firewall --default-zone=drop --service=ssh --zone=public # 예시, 실제 zone 정책에 맞게 조정

# %post 섹션 (예시: 특정 IP 대역 192.168.1.0/24 에서만 SSH 접근 허용)
# 이 경우, 위의 명령어 섹션에서 --service=ssh 를 빼거나,
```

```
# public zone에서 ssh 서비스를 제거하고 새로운 trusted zone을 만들어야 합니다.

# 예시: public zone에서 ssh 제거 후, trusted_ssh zone 생성 및 특정 대역에 ssh 허용
# firewall-cmd --permanent --zone=public --remove-service=ssh
# firewall-cmd --permanent --new-zone=trusted_ssh
# firewall-cmd --permanent --zone=trusted_ssh --set-target=ACCEPT
# firewall-cmd --permanent --zone=trusted_ssh --add-source=192.168.1.0/24
# firewall-cmd --permanent --zone=trusted_ssh --add-service=ssh
# firewall-cmd --reload # 설치 마지막에 한 번만 실행하는 것이 좋음

# 2.3.15. Postfix 보안 설정 (Postfix 설치 시)
# Sendmail 패키지는 설치하지 않음 (-sendmail)
# Postfix 설치 시 (예: dnf install postfix) 보안 설정 적용
if rpm -q postfix; then
  postconf -e "disable_vrfy_command = yes"
 # 오픈 릴레이 방지 설정 (기본값일 수 있으나 확인)
  # postconf -e "smtpd_recipient_restrictions = permit_mynetworks,
reject_unauth_destination"
  # systemctl restart postfix # 설치 마지막에 한 번만
fi

# 2.3.18. ssh 원격접속 보안 강화
#
#############################################################################
##
# SSH 보안 설정 통합 (수정된 부분)
#
#############################################################################
##
echo "Applying consolidated SSH security settings..."

# /etc/ssh/sshd_config.d/ 디렉토리에 단일 설정 파일 생성
SSHD_CONFIG_DIR="/etc/ssh/sshd_config.d"
CUSTOM_SSH_CONFIG_FILE="$SSHD_CONFIG_DIR/99-kickstart-hardening.conf" # 우선
순위를 위해 99- prefix 사용

mkdir -p "$SSHD_CONFIG_DIR"

# 단일 설정 파일에 모든 SSH 관련 보안 설정을 기록합니다.
# 기존의 여러 파일로 나뉘어 있던 설정을 하나로 통합합니다.
cat <<EOF > "$CUSTOM_SSH_CONFIG_FILE"
# This file was generated by Kickstart and contains consolidated security
settings.

# [Root 계정 원격 접속 제한]
PermitRootLogin no

# [.rhosts 및 hosts.equiv 사용 금지]
IgnoreRhosts yes
HostbasedAuthentication no
```

```
# [SSH 원격접속 보안 강화 (CIS Benchmark 권장 사항 등)]
ClientAliveInterval 300
ClientAliveCountMax 0
LoginGraceTime 60
MaxAuthTries 4
LogLevel VERBOSE
PermitEmptyPasswords no
PermitUserEnvironment no
UsePAM yes

# [로그온 시 경고 메시지 제공]
Banner /etc/issue.net

# (선택 사항) 보다 강력한 암호화 알고리즘 설정 (필요시 주석 해제하여 사용)
# Ciphers aes256-ctr,aes192-ctr,aes128-ctr
# MACs hmac-sha2-512,hmac-sha2-256
# KexAlgorithms diffie-hellman-group-exchange-sha256

EOF

# 파일 권한 설정
chown root:root "$CUSTOM_SSH_CONFIG_FILE"
chmod 644 "$CUSTOM_SSH_CONFIG_FILE"
echo "Consolidated SSH settings applied to $CUSTOM_SSH_CONFIG_FILE"
#
################################################################################
##
# SSH 설정 통합 종료
#
################################################################################
##



# 2.3.20. at 파일 소유자 및 권한 설정
echo "Disabling atd service..."
# systemctl을 사용하여 atd 서비스를 비활성화 및 중지합니다.
# 서비스가 존재하지 않거나 이미 비활성화된 경우에도 오류를 내지 않고 넘어갑니다.
systemctl disable --now atd >/dev/null 2>&1 || true
echo "atd service hardening complete."

# at 명령어 사용자 제어
echo "Starting job scheduler security hardening..."

# --- at 명령어 보안 설정 ---
if [ -x "/usr/bin/at" ]; then
    echo "Configuring 'at' security..."
    # at.deny 파일이 있다면 삭제하여 at.allow 파일로만 접근 제어
    rm -f /etc/at.deny

    # at.allow 파일을 생성하고 root만 허용
    echo "root" > /etc/at.allow
    chown root:root /etc/at.allow
```

```
    chmod 600 /etc/at.allow

    # /usr/bin/at 실행 파일의 권한을 4750으로 설정 (소유자 외 실행 금지)
    chmod 4750 /usr/bin/at
else
    echo "'at' command not found, skipping 'at' configuration."
fi

# 2.3.22. 로그온 시 경고 메시지 제공
BANNER_TEXT="
################################################################################
# WARNING: This system is for the use of authorized users only.                #
# Individuals using this computer system without authority, or in excess #
# of their authority, are subject to having all of their activities on   #
# this system monitored and recorded by system personnel.                      #
#                                                                              #
# In the course of monitoring individuals improperly using this system,  #
# or in the course of system maintenance, the activities of authorized   #
# users may also be monitored.                                                 #
#                                                                              #
# Anyone using this system expressly consents to such monitoring and     #
# is advised that if such monitoring reveals possible evidence of        #
# criminal activity, system personnel may provide the evidence of such   #
# monitoring to law enforcement officials.                                     #
################################################################################

"
echo "$BANNER_TEXT" > /etc/motd
echo "$BANNER_TEXT" > /etc/issue
echo "$BANNER_TEXT" > /etc/issue.net

# --- 2.4. 로그 및 감사, 패치 ---
echo "Applying Logging, Auditing, and Patching settings..."

# 2.4.1. 최신 보안패치 적용
#echo "Applying latest security patches..."
#dnf update -y
# (선택 사항) dnf-automatic 설정
# dnf install -y dnf-automatic
# sed -i 's/^apply_updates =.*/apply_updates = yes/' /etc/dnf/automatic.conf
# 예시
# systemctl enable --now dnf-automatic.timer

# 2.4.3. 시스템 로깅 설정 (rsyslog, auditd) - 기본 예시
# echo "authpriv.* /var/log/auth.log" > /etc/rsyslog.d/custom-auth.conf
# dnf install -y audit; systemctl enable --now auditd
# echo "-w /etc/passwd -p wa -k passwd_changes" > /etc/audit/rules.d/audit-
custom.rules
# augenrules --load
```

```
### 터미널 명령어 로깅
echo ">>> Starting command logging configuration..."

# 1. /etc/profile.d/ 에 명령어 로깅 스크립트 생성
# 'EOF'에 작은 따옴표를 사용하여 변수($USER, $PWD 등)가 지금 확장되지 않고
# 사용자가 로그인할 때 동적으로 확장되도록 합니다.
cat <<'EOF' > /etc/profile.d/command-logging.sh
#!/bin/bash
# Script to log user commands to syslog

# history -a: 현재 세션의 히스토리를 즉시 파일에 쓰도록 함
# logger: history의 마지막 명령어를 가져와 syslog로 전송
# -p local6.info: local6 facility, info 레벨로 로그 전송
# -t bash-commands: 로그에 'bash-commands' 태그 추가
export PROMPT_COMMAND='history -a; logger -p local6.info -t bash-commands
"USER=$USER IP=$(who -m | awk "{print \$5}" | tr -d "()") PWD=$PWD
CMD=$(history 1 | sed "s/^[ ]*[0-9]\+[ ]*//")"'
EOF

# 스크립트에 실행 권한 부여
chmod +x /etc/profile.d/command-logging.sh
echo ">>> Created /etc/profile.d/command-logging.sh"

# 2. rsyslog 설정 파일 생성
# local6 facility로 들어오는 로그를 /var/log/command.log 파일에 기록하도록 설정
cat <<EOF > /etc/rsyslog.d/command-logging.conf
# Rule for command logging
local6.* /var/log/command.log
EOF
echo ">>> Created /etc/rsyslog.d/command-logging.conf"

# 3. logrotate 설정 파일 생성
# /var/log/command.log 파일을 자동으로 관리(로테이트)하도록 설정
cat <<EOF > /etc/logrotate.d/command-log
/var/log/command.log {
    missingok
    notifempty
    create 0600 root root
    postrotate
        /bin/systemctl kill -s HUP rsyslogd.service >/dev/null 2>&1 || true
    endscript
}
EOF
echo ">>> Created /etc/logrotate.d/command-log"

echo ">>> Command logging configuration finished."



################################
# --- 서비스 재시작 (설치 마지막에 한 번만 수행 권장) ---
echo "Restarting services..."
```

```
# systemctl restart sshd
# systemctl restart rsyslog
# if rpm -q postfix; then systemctl restart postfix; fi
# if rpm -q auditd; then service auditd restart; fi # auditd는 service 명령어 사용 가능
# firewall-cmd --reload # 방화벽 규칙 적용

echo "RHEL 9 Security Hardening Post-Install Script Finished."

%end
```