

??????yum?

???????? DNF/YUM???? AlmaLinux/CentOS/Fedora/Rocky

```
curl -fsSL https://www.nat.ac.cn/shell/collect_system_centos.sh | sudo env INSTALL_DEPS=0
INSTALL_AIDE=0 bash -s
```

???????? a???? f????????

```
# 检查pwquality
curl -fsSL https://www.nat.ac.cn/shell/collect_system_centos.sh | sudo env INSTALL_DEPS=1
INSTALL_AIDE=0 bash -s

# 检查
curl -fsSL https://www.nat.ac.cn/shell/collect_system_centos.sh | sudo env INSTALL_DEPS=1
INSTALL_AIDE=1 bash -s
```

????????

```
#!/bin/bash

set -euo pipefail

if [[ "$(uname)" != "Linux" ]]; then
    echo "不是 Linux 系统"; exit 1
fi

heartbeat() { while true; do echo "[heartbeat] $(date '+%F %T') 心跳..."; sleep 5; done; }
heartbeat & HEARTBEAT_PID=$!

cleanup() { kill "$HEARTBEAT_PID" 2>/dev/null || true; }
trap cleanup EXIT

SYSTEM_IP=$(ip addr show | grep -w inet | grep -v 127.0.0.1 | awk '{print $2}' | cut -d/ -f1 |
head -n1)
[ -z "$SYSTEM_IP" ] && SYSTEM_IP="unknown"

TIMESTAMP=$(date +%Y%m%d%H%M%S)

OUTPUT_FILE="$HOME/${SYSTEM_IP}_${TIMESTAMP}.txt"
```

```

touch "$OUTPUT_FILE"

run_command() {
    local cmd="$1" desc="$2"
    echo "$desc"
    {
        echo; echo "echo \"\$desc\""; echo "echo \"Running: \$cmd\""; echo "$cmd"
        eval "$cmd" 2>&1 || echo "XXXXXXXXXX"
        echo "-----"
    } >> "$OUTPUT_FILE"
}

echo "####..." > "$OUTPUT_FILE"
echo "####: $OUTPUT_FILE" >> "$OUTPUT_FILE"
echo "####: $(date)" >> "$OUTPUT_FILE"
echo "-----" >> "$OUTPUT_FILE"

# XXXXXXXXXXXXXXXX
INSTALL_DEPS="${INSTALL_DEPS:-0}" # sudo/util-linux/audit/pwquality/google-authenticator
INSTALL_SELINUX="${INSTALL_SELINUX:-0}" # XXXXX SELinux XXXXX

# XXXXXXXXXCentOS 8+ XX dnfCentOS 7 XX yum
if command -v dnf >/dev/null 2>&1; then
    PKG_MGR="dnf"
elif command -v yum >/dev/null 2>&1; then
    PKG_MGR="yum"
else
    PKG_MGR="yum"
fi

if [[ "$INSTALL_DEPS" == "1" || "$INSTALL_SELINUX" == "1" ]]; then
    run_command "$PKG_MGR makecache || true" "XXXXXXXXXXXXXXXXXX"
fi

if [[ "$INSTALL_DEPS" == "1" ]]; then
    run_command "$PKG_MGR install -y sudo util-linux audit libpwquality google-authenticator ||
true" "XXXXXXXXXXXXXXXXXX"
fi

if [[ "$INSTALL_SELINUX" == "1" ]]; then
    run_command "$PKG_MGR install -y policycoreutils policycoreutils-python-utils selinux-policy
selinux-policy-targeted || true" "XXXXX SELinux XXXXXXXXX"

```

```

fi

if command -v sudo >/dev/null 2>&1; then SUDO="sudo"; else SUDO=""; fi

#####  #####

run_command "hostnamectl" " "
run_command "timedatectl" " / "
run_command "ip addr show" "IP "
run_command "uname -a" " "
run_command "cat /etc/redhat-release 2>/dev/null || cat /etc/os-release" " "

#####  #####

run_command "cat /etc/passwd" " "
run_command "cat /etc/group" " "
run_command "${SUDO} cat /etc/shadow" " root"
run_command "${SUDO} cat /etc/sudoers | grep -v ^# && ${SUDO} ls -l /etc/sudoers.d" "sudo "
run_command "w" " "
run_command "who" " "
run_command "last | head" " util-linux  /var/log/wtmp"
run_command "lastlog | head" " "
run_command "${SUDO} grep -v ^# /etc/login.defs" " / "
run_command "chage -l $(whoami)" " "

#####  /PAM #####

run_command "${SUDO} grep -v ^# /etc/pam.d/system-auth" "PAM "
run_command "${SUDO} grep -v ^# /etc/pam.d/password-auth" "PAM "
run_command "${SUDO} grep pam_pwquality /etc/pam.d/system-auth /etc/pam.d/password-auth" " pwquality"
run_command "${SUDO} grep -v '^#' /etc/security/pwquality.conf" "pwquality "
run_command "${SUDO} grep -v '^#' /etc/security/pwquality.conf.d/*.conf 2>/dev/null"
"pwquality "
run_command "rpm -qa | grep libpwquality || yum list installed 2>/dev/null | grep
libpwquality" "pwquality "
run_command "${SUDO} grep -H 'pam_google_authenticator.so' /etc/pam.d/sshd /etc/pam.d/system-
auth /etc/pam.d/password-auth 2>/dev/null" "PAM  Google Authenticator"

#####  SSH  #####

run_command "${SUDO} grep -E
'^(Protocol|PermitRootLogin|PasswordAuthentication|PermitEmptyPasswords|ChallengeResponseAuthe
ntication|AuthenticationMethods|ClientAliveInterval|ClientAliveCountMax)'"

```

```

/etc/ssh/sshd_config" "SSH 配置"
run_command "cat \${HOME}/.ssh/authorized_keys" "配置公钥"

##### 配置 su 和 sudo #####
run_command "${SUDO} grep -v ^# /etc/pam.d/su" "su 配置"
run_command "${SUDO} grep -v ^# /etc/pam.d/sudo" "sudo PAM 配置"
run_command "echo \${TMOUT:-unset}; ${SUDO} grep TMOUT /etc/profile /etc/bashrc 2>/dev/null" "
配置 bash"

##### 配置 audit #####
run_command "ps -eo user,pid,cmd | grep auditd" "auditd 进程"
run_command "${SUDO} systemctl status auditd" "auditd 状态"
run_command "${SUDO} auditctl -l" "auditctl 规则"
run_command "${SUDO} cat /etc/audit/audit.rules 2>/dev/null" "audit.rules 文件"
run_command "${SUDO} grep -v ^# /etc/audit/rules.d/*.rules 2>/dev/null" "audit.rules.d 文件"
run_command "ls -ltr /var/log | tail" "日志文件"
run_command "${SUDO} head -n 20 /var/log/messages" "messages 日志"
run_command "${SUDO} head -n 20 /var/log/secure" "secure 日志"
run_command "${SUDO} grep -v ^# /etc/logrotate.conf" "logrotate.conf 文件"
run_command "ls -l /etc/logrotate.d" "logrotate.d 文件"

##### 配置 firewalld #####
run_command "ss -lntp" "TCP 监听"
run_command "ip route" "路由"
run_command "ip neigh" "ARP 表"
run_command "${SUDO} firewall-cmd --list-all 2>/dev/null || echo 'firewalld 配置'"
"firewalld 配置"
run_command "${SUDO} systemctl status firewalld" "firewalld 状态"
run_command "${SUDO} iptables -L -n -v" "iptables 规则"

##### 配置 rsyslog 和 auditd #####
run_command "systemctl list-unit-files --type=service --state=enabled" "启用的服务"
run_command "systemctl show rsyslog.service -p User,Group,UID,GID" "rsyslog 配置"
run_command "systemctl show auditd.service -p User,Group,UID,GID" "auditd 配置"
run_command "ps -eo user,pid,cmd --sort=pid | head -n 30" "进程列表 30 个"

##### SELinux 配置 #####
if command -v getenforce >/dev/null 2>&1 && [[ "$(getenforce 2>/dev/null)" != "Disabled" ]];
then
    run_command "getenforce" "SELinux 配置"

```

```

run_command "${SUDO} sestatus" "SELinux 0000"
run_command "${SUDO} cat /etc/selinux/config" "SELinux 0000"
run_command "${SUDO} semodule -l | head -n 20" "SELinux 0000000200"
run_command "${SUDO} getsebool -a | head -n 30" "SELinux 0000000300"
run_command "${SUDO} ls -Z /etc/passwd /etc/shadow /etc/group 2>/dev/null" "0000 SELinux 000
"

run_command "${SUDO} ps -eZ | head -n 20" "00 SELinux 00000200"
run_command "${SUDO} ausearch -m avc -ts recent 2>/dev/null | head -n 20 || echo 'audit 000
AVC 00000 ausearch 000'" "SELinux 0000000"
else
    echo "SELinux 0000000000 SELinux 00" | tee -a "$OUTPUT_FILE"
fi

##### 00000000000 #####
run_command "${SUDO} sysctl -a | grep -E
'net.ipv4.conf.all.(accept_source_route|accept_redirects|send_redirects|rp_filter|log_martians
|forwarding)'" "0000000000"
run_command "mount | grep -E 'noexec|nodev|nosuid'" "0000000"

echo "00000000000 $OUTPUT_FILE"

```

Revision #14

Created 10 September 2025 16:54:44 by Forest

Updated 9 April 2026 05:12:55 by Forest