https://blog.csdn.net/Y94639997/article/details/141057481

下载地址

openssl：https://ftp.openssl.org/source/old/1.1.1/ openssh：
https://ftp.openbsd.org/pub/OpenBSD/OpenSSH/portable/

测试版本 #openssl https://www.openssl.org/source/old/1.1.1/openssl-1.1.1w.tar.gz

https://www.openssl.org/source/old/3.3.1/openssl-3.3.1.tar.gz

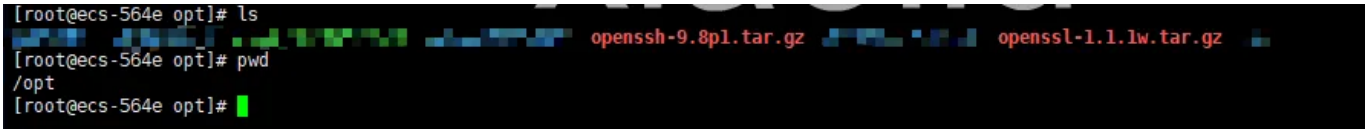
https://ghproxy.cn/https://github.com/openssl/openssl/releases/download/openssl-3.3.1/openssl-3.3.1.tar.gz

#openssh https://cdn.openbsd.org/pub/OpenBSD/OpenSSH/portable/openssh-9.8p1.tar.gz

https://cdn.openbsd.org/pub/OpenBSD/OpenSSH/portable/openssh-9.9p2.tar.gz

安装准备(这是前提)

查看当前openssh版本，并上传openssl-1.1.1w.tar.gz和 openssh-9.5p1.tar.gz 至 /opt/ 目录



一键脚本

centos升级

执行命令 bash ssh_update.sh

```
#https://dfs.zjlll.net/download/ssh_update.sh
#!/bin/bash

# 安装yum依赖
echo -e "\n \033[33m 安装yum依赖... \033[0m \n"
yum -y install gcc make perl zlib zlib-devel pam pam-devel  perl-IPC-Cmd

# 备份原有的openssl文件
echo -e "\n \033[33m 开始备份OpenSSL服务文件... \033[0m \n"
mv /usr/bin/openssl /usr/bin/openssl.bak
mv /usr/lib64/openssl /usr/lib64/openssl.bak
mv /usr/include/openssl /usr/include/openssl.bak

# 解压OpenSSL安装包
echo -e "\n \033[33m 开始解压OpenSSL安装包... \033[0m \n"
cd /opt/openssl
tar -zxvf openssl-1.1.1w.tar.gz
cd openssl-1.1.1w/
```

```
mkdir -p /usr/local/openssl

# 编译安装OpenSSL
echo -e "\n \033[33m 开始初始化、编译、安装OpenSSL服务... \033[0m \n"
./config --prefix=/usr/local/openssl --shared
make && make install

if [ $? -ne 0 ]; then
    echo -e "\n \033[30m OpenSSL服务安装失败!!! \033[0m \n"
    exit 1
else
    echo -e "\n \033[33m OpenSSL服务安装成功... \033[0m \n"
fi

# 进行相关使用配置
echo -e "\n \033[33m 开始为OpenSSL服务进行相关配置... \033[0m \n"
ln -s /usr/local/openssl/bin/openssl /usr/bin/openssl
ln -s /usr/local/openssl/include/openssl /usr/include/openssl
echo "/usr/local/openssl/lib" >> /etc/ld.so.conf
ldconfig

# 查看服务是否安装成功
echo -e "\n \033[33m 开始检测OpenSSL服务是否安装成功... \033[0m \n"
openssl version
openssl version -a

# 暂停，询问是否继续安装OpenSSH
echo -e "\n \033[33m 是否继续安装OpenSSH服务? (yes/no): \033[0m \n"
read user_input

if [ "$user_input" == "yes" ]; then
    # 备份原有的OpenSSH文件
    echo -e "\n \033[33m 开始备份OpenSSH服务文件... \033[0m \n"
    mv /etc/ssh/ /etc/ssh.bak
    mv /usr/bin/ssh /usr/bin/ssh.bak
    mv /usr/sbin/sshd /usr/sbin/sshd.bak
    mv /etc/init.d/sshd /etc/init.d/sshd.bak

    # 解压OpenSSH安装包
    echo -e "\n \033[33m 开始解压OpenSSH安装包... \033[0m \n"
    cd /opt/
    tar -zxf openssh-9.8p1.tar.gz
    cd openssh-9.8p1/

    # 编译安装OpenSSH
    echo -e "\n \033[33m 开始初始化、编译、安装OpenSSH服务... \033[0m \n"
    ./configure --prefix=/usr --sysconfdir=/etc/ssh --with-md5-passwords --with-
pam --with-tcp-wrappers --with-ssl-dir=/usr/local/openssl --with-privsep-
path=/var/lib/sshd --without-hardening
    make && make install

    if [ $? -ne 0 ]; then
        echo -e "\n \033[30m OpenSSH服务安装失败!!! \033[0m \n"
        exit 1
```

```
    else
        echo -e "\n \033[33m OpenSSH服务安装成功... \033[0m \n"
    fi

    # 进行相关使用配置
    echo -e "\n \033[33m 开始为OpenSSH服务进行相关配置... \033[0m \n"
    chmod 600 /etc/ssh/ssh_host_rsa_key /etc/ssh/ssh_host_ecdsa_key
/etc/ssh/ssh_host_ed25519_key
    echo "PasswordAuthentication yes" >> /etc/ssh/sshd_config
    echo "PermitRootLogin yes" >> /etc/ssh/sshd_config
    echo "UsePAM yes" >> /etc/ssh/sshd_config

    cp -pf /opt/openssh-9.8p1/contrib/redhat/sshd.init /etc/init.d/sshd
    chmod +x /etc/init.d/sshd
    chkconfig --add sshd
    chkconfig sshd on
    systemctl restart sshd

    # 查看服务是否安装成功
    echo -e "\n \033[33m 开始检测OpenSSH服务是否安装成功... \033[0m \n"
    ssh -V
    systemctl status sshd
else
    echo -e "\n \033[31m 已选择不安装OpenSSH服务。 \033[0m \n"
    exit 0
fi
```

ubt升级

```
#!/bin/bash

# 安装依赖
echo -e "\n \033[33m 安装所需依赖... \033[0m \n"
apt-get update && apt-get -y install gcc make perl zlib zlib-devel pam pam-devel

# 备份原有的openssl文件
echo -e "\n \033[33m 开始备份OpenSSL服务文件... \033[0m \n"
mv /usr/bin/openssl /usr/bin/openssl.bak
mv /usr/lib64/openssl /usr/lib64/openssl.bak
mv /usr/include/openssl /usr/include/openssl.bak

# 解压OpenSSL安装包
echo -e "\n \033[33m 开始解压OpenSSL安装包... \033[0m \n"
cd /opt/openssl
tar -zxvf openssl-1.1.1w.tar.gz
cd openssl-1.1.1w/
mkdir -p /usr/local/openssl

# 编译安装OpenSSL
echo -e "\n \033[33m 开始初始化、编译、安装OpenSSL服务... \033[0m \n"
./config --prefix=/usr/local/openssl --shared
```

```
make && make install

if [ $? -ne 0 ]; then
    echo -e "\n \033[30m OpenSSL服务安装失败!!! \033[0m \n"
    exit 1
else
    echo -e "\n \033[33m OpenSSL服务安装成功... \033[0m \n"
fi

# 设置正确的环境变量
echo -e "\n \033[33m 设置环境变量，确保 OpenSSL 正确链接... \033[0m \n"
export LD_LIBRARY_PATH=/usr/local/openssl/lib:$LD_LIBRARY_PATH

# 进行相关使用配置
echo -e "\n \033[33m 开始为OpenSSL服务进行相关配置... \033[0m \n"
ln -s /usr/local/openssl/bin/openssl /usr/bin/openssl
ln -s /usr/local/openssl/include/openssl /usr/include/openssl
echo "/usr/local/openssl/lib" >> /etc/ld.so.conf
ldconfig

# 查看服务是否安装成功
echo -e "\n \033[33m 开始检测OpenSSL服务是否安装成功... \033[0m \n"
openssl version
openssl version -a

# 暂停，询问是否继续安装OpenSSH
echo -e "\n \033[33m 是否继续安装OpenSSH服务? (yes/no): \033[0m \n"
read user_input

if [[ "$user_input" == "yes" ]]; then
    # 备份原有的OpenSSH文件
    echo -e "\n \033[33m 开始备份OpenSSH服务文件... \033[0m \n"
    mv /etc/ssh/ /etc/ssh.bak
    mv /usr/bin/ssh /usr/bin/ssh.bak
    mv /usr/sbin/sshd /usr/sbin/sshd.bak
    mv /etc/init.d/sshd /etc/init.d/sshd.bak

    # 解压OpenSSH安装包
    echo -e "\n \033[33m 开始解压OpenSSH安装包... \033[0m \n"
    cd /opt/
    tar -zxf openssh-9.8p1.tar.gz
    cd openssh-9.8p1/

    # 编译安装OpenSSH
    echo -e "\n \033[33m 开始初始化、编译、安装OpenSSH服务... \033[0m \n"
    ./configure --prefix=/usr --sysconfdir=/etc/ssh --with-md5-passwords --with-
pam --with-tcp-wrappers --with-ssl-dir=/usr/local/openssl --with-privsep-
path=/var/lib/sshd --without-hardening
    make && make install

    if [ $? -ne 0 ]; then
        echo -e "\n \033[30m OpenSSH服务安装失败!!! \033[0m \n"
        exit 1
    else
```

```
        echo -e "\n \033[33m OpenSSH服务安装成功... \033[0m \n"
    fi

    # 进行相关使用配置
    echo -e "\n \033[33m 开始为OpenSSH服务进行相关配置... \033[0m \n"
    chmod 600 /etc/ssh/ssh_host_rsa_key /etc/ssh/ssh_host_ecdsa_key
/etc/ssh/ssh_host_ed25519_key
    echo "PasswordAuthentication yes" >> /etc/ssh/sshd_config
    echo "PermitRootLogin yes" >> /etc/ssh/sshd_config
    echo "UsePAM yes" >> /etc/ssh/sshd_config

    cp -pf /opt/openssh-9.8p1/contrib/redhat/sshd.init /etc/init.d/sshd
    chmod +x /etc/init.d/sshd
    chkconfig --add sshd
    chkconfig sshd on
    systemctl restart sshd

    # 查看服务是否安装成功
    echo -e "\n \033[33m 开始检测OpenSSH服务是否安装成功... \033[0m \n"
    ssh -V
    systemctl status sshd
else
    echo -e "\n \033[31m 已选择不安装OpenSSH服务。 \033[0m \n"
    exit 0
fi
```

直接替换url就行了

centos

```
#!/bin/bash
set -e

# ========== [0] 版本配置 ==========
OPENSSL_VERSION="1.1.1w"
OPENSSH_VERSION="9.8p1"
OPENSSL_URL="https://github.com/openssl/openssl/releases/download/OpenSSL_1_1_1w/o
penssl-${OPENSSL_VERSION}.tar.gz"
OPENSSH_URL="https://cdn.openbsd.org/pub/OpenBSD/OpenSSH/portable/openssh-${OPENSS
H_VERSION}.tar.gz"

OPENSSL_TAR="openssl-${OPENSSL_VERSION}.tar.gz"
OPENSSH_TAR="openssh-${OPENSSH_VERSION}.tar.gz"
OPENSSL_DIR="openssl-${OPENSSL_VERSION}"
OPENSSH_DIR="openssh-${OPENSSH_VERSION}"
OPENSSL_PREFIX="/usr/local/openssl"

# ========== [1] 安装依赖 ==========
echo -e "\n\033[33m[1/7] 安装编译依赖...\033[0m"
yum install -y gcc make perl zlib zlib-devel pam pam-devel perl-IPC-Cmd wget
tcp_wrappers-devel
```

```
# ========== [2] 备份旧版本 ==========
echo -e "\n\033[33m[2/7] 备份系统 OpenSSL...\033[0m"
mv /usr/bin/openssl /usr/bin/openssl.bak 2>/dev/null || true
mv /usr/include/openssl /usr/include/openssl.bak 2>/dev/null || true
mv /usr/lib64/openssl /usr/lib64/openssl.bak 2>/dev/null || true

echo -e "\n\033[33m备份系统 OpenSSH...\033[0m"
mv /etc/ssh /etc/ssh.bak 2>/dev/null || true
mv /usr/bin/ssh /usr/bin/ssh.bak 2>/dev/null || true
mv /usr/sbin/sshd /usr/sbin/sshd.bak 2>/dev/null || true
mv /etc/init.d/sshd /etc/init.d/sshd.bak 2>/dev/null || true

# ========== [3] 下载并编译安装 OpenSSL ==========
echo -e "\n\033[33m[3/7] 下载并安装 OpenSSL ${OPENSSL_VERSION}...\033[0m"
cd /opt
[ -f "$OPENSSL_TAR" ] || wget "$OPENSSL_URL"
rm -rf "$OPENSSL_DIR" && tar -zxf "$OPENSSL_TAR"
cd "$OPENSSL_DIR"
./config --prefix=$OPENSSL_PREFIX --shared zlib
make -j$(nproc)
make install

echo "/usr/local/openssl/lib" > /etc/ld.so.conf.d/openssl-${OPENSSL_VERSION}.conf
ldconfig

ln -sf /usr/local/openssl/bin/openssl /usr/bin/openssl
ln -sf /usr/local/openssl/include/openssl /usr/include/openssl

# ========== [4] 下载并编译安装 OpenSSH ==========
echo -e "\n\033[33m[4/7] 下载并安装 OpenSSH ${OPENSSH_VERSION}...\033[0m"
cd /opt
[ -f "$OPENSSH_TAR" ] || wget "$OPENSSH_URL"
rm -rf "$OPENSSH_DIR" && tar -zxf "$OPENSSH_TAR"
cd "$OPENSSH_DIR"

./configure --prefix=/usr \
  --sysconfdir=/etc/ssh \
  --with-md5-passwords \
  --with-pam \
  --with-tcp-wrappers \
  --with-ssl-dir=$OPENSSL_PREFIX \
  --with-privsep-path=/var/lib/sshd \
  --without-hardening

make -j$(nproc)
make install

# ========== [5] 配置 SSH ==========
echo -e "\n\033[33m[5/7] 配置 SSHD 服务...\033[0m"
ssh-keygen -A
sed -i 's/#\?PasswordAuthentication.*/PasswordAuthentication yes/'
/etc/ssh/sshd_config
sed -i 's/#\?PermitRootLogin.*/PermitRootLogin yes/' /etc/ssh/sshd_config
```

```
echo "UsePAM yes" >> /etc/ssh/sshd_config

cp -pf contrib/redhat/sshd.init /etc/init.d/sshd
chmod +x /etc/init.d/sshd
chkconfig --add sshd
chkconfig sshd on

# ========== [6] 启动 SSH 服务 ==========
echo -e "\n\033[33m[6/7] 启动 SSHD...\033[0m"
systemctl daemon-reexec || true
systemctl restart sshd || service sshd restart || /etc/init.d/sshd restart

# ========== [7] 验证结果 ==========
echo -e "\n\033[32m[7/7] 安装完成，验证信息如下：\033[0m"
ssh -V
/usr/local/openssl/bin/openssl version
ldd /usr/sbin/sshd | grep ssl

echo -e "\n\033[32m☑ OpenSSL ${OPENSSL_VERSION} + OpenSSH ${OPENSSH_VERSION} 安
装完成。\033[0m"
```

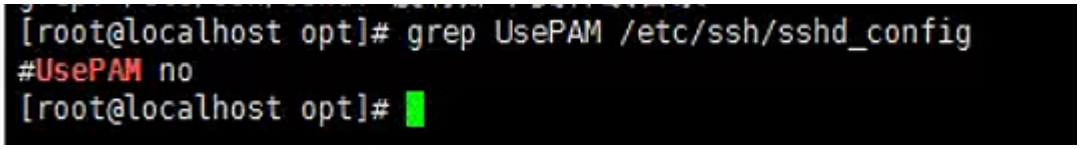
遇见的问题

centos7.5 OpenSSH_7.4p1, OpenSSL 1.0.2k-fips 26 Jan 2017 升级 OpenSSH_9.8p1, OpenSSL 1.1.1w 11 Sep 2023

升级后遇到，22端口通的，登录不上 排查 先看sshd配置文件中是否打开了pam

```
grep UsePAM /etc/ssh/sshd_config
```

升级前是没有开启pam的，执行脚本后开启了pam



如果开启了没有这个文件，就要创建并且编辑

初始版本，只限于登录

```
[root@localhost ~]# cat /etc/pam.d/sshd
# PAM configuration for the Secure Shell service
auth       required     pam_unix.so
```

```
account    required     pam_unix.so
password   required     pam_unix.so
session    required     pam_unix.so
```

升级版本，有次数和时间限制

```
[root@ksp-control-1 ~]# ssh -V
OpenSSH_7.4p1, OpenSSL 1.0.2k-fips  26 Jan 2017
[root@ksp-control-1 ~]# cat /etc/pam.d/sshd
#%PAM-1.0
auth       required     pam_sepermit.so
auth       substack     password-auth
auth       include      postlogin
# Used with polkit to reauthorize users in remote sessions
-auth      optional     pam_reauthorize.so prepare
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 be executed in the
user context
session    required     pam_selinux.so open env_params
session    required     pam_namespace.so
session    optional     pam_keyinit.so force revoke
session    include      password-auth
session    include      postlogin
# Used with polkit to reauthorize users in remote sessions
-session   optional     pam_reauthorize.so prepare
```