OpenSSH Server
开发手册

江天文

2024/06/17

摘要

- OpenSSH Server在开发中可以被Qt Creator和Eclipse用来进行应用程序上传和远程调试，极大地提高了开发效率。同时它还可以远程复制和删除目标系统上的文件及登陆shell控制台操作目标机器。

NuDeveloper Ecosystem – Make engineers' jobs easier.

nuvoTon

Qt Creator远程调试中使用OpenSSH Server

- 在Qt Creator中远程调试中使用OpenSSH Server时，需要配置Buildroot，在Toolchain下，打开Build cross gdb for the host及其下面的Python support。

   NuDeveloper Ecosystem – Make engineers' jobs easier.

Qt Creator远程调试中使用OpenSSH Server

- 在Target packages →
  Networking applications
  下打开openssh和rsync

NuDeveloper Ecosystem – Make engineers' jobs easier.

配置OpenSSH Server

- 通过SSH远程登陆目标机器之前，需要事先配置OpenSSH Server。配置文件位于目标文件系统/etc/ssh/sshd_config，其配置如下

OpenSSH Server Configuration (/etc/ssh/sshd_config)	
PermitRootLogin	yes
PermitEmptyPasswords	yes
PasswordAuthentication	yes

上述配置存在"安全"问题，仅适用于在开发中"免密码输入"方便调试。接下来会将上述配置通过Post-Build脚本预先内置到目标根文件系统（rootfs）上，方便首次开机时，默认就能快速连接上目标机器。

 NuDeveloper Ecosystem – Make engineers' jobs easier.

nuvoTon

Post-Build脚本中预置OpenSSH Server配置文件

- Buildroot通过Post-Build 脚本在创建根文件系统 （rootfs）之前提供一个 机会让用户来修改文件 系统内容。

NuDeveloper Ecosystem – Make engineers' jobs easier.

nuvoTon

Post-Build脚本中预置OpenSSH Server配置文件

- 在Post-Build脚本中，添加下行

Post-Build Script (board/nuvoton/ma35d1/post-build.sh)

```
if grep -Eq "^BR2_PACKAGE_OPENSSH=y$" ${BR2_CONFIG}; then
    sed -i 's/^#PermitRootLogin.*/PermitRootLogin yes/'              ${TARGET_DIR}/etc/ssh/sshd_config
    sed -i 's/^#PermitEmptyPasswords.*/PermitEmptyPasswords yes/'    ${TARGET_DIR}/etc/ssh/sshd_config
    sed -i 's/^#PasswordAuthentication.*/PasswordAuthentication yes/' ${TARGET_DIR}/etc/ssh/sshd_config
fi
```

NuDeveloper Ecosystem – Make engineers' jobs easier.

nuvoTon

通过SSH公匙验证登陆系统

- 前面提到的是基于用户名和密码的方式登陆认证，这种方式存在"安全"隐患，下面将会介绍基于SSH公匙的登陆验证。
- SSH公匙认证允许多个用户以同一个系统用户名（root）但不使用同一个密码的方式登陆系统。
- 注销某个用户的SSH认证不会影响到其它的SSH用户。
- 允许单个用户登陆进多个账户而不需要管理多个密码。

NuDeveloper Ecosystem – Make engineers' jobs easier.

nuvoTon

通过**SSH**公匙验证登陆系统

- 通过SSH公匙登陆操作步骤如下
  1. 在Ubuntu主机（编译机器）上生成RSA密匙
     $ ssh-keygen -t rsa
  2. 使用SCP复制本地的公匙到远程从机
     $ scp ~/.ssh/id_rsa.pub root@192.168.1.102:
  3. 登陆远程从机
     $ ssh root@192.168.1.102
  4. 如果远程从机上不存在~/.ssh/authorized_keys文件，在命令行下创建
     # mkdir –p ~/.ssh
     # touch ~/.ssh/authorized_keys
  5. 在远程从机上，将公匙追加到文件~/.ssh/authorized_keys
     # cat ~/id_rsa.pub >> ~/.ssh/authorized_keys
  6. 执行步骤1到5添加多个SSH公匙

          NuDeveloper Ecosystem – Make engineers' jobs easier.   **nuvoTon**

通过**SSH**公匙验证登陆系统

- （接上）通过SSH公匙登陆操作步骤
  7. 在远程从机上，修改OpenSSH Server配置文件/etc/ssh/sshd_config

OpenSSH Server Configuration (/etc/ssh/sshd_config)	
PermitRootLogin	yes
PermitEmptyPasswords	no
PasswordAuthentication	no

  8. 在远程从机上，重启OpenSSH Server
     # /etc/init.d/S50sshd restart
  9. 在Ubuntu主机上，再次SSH登陆远程从机
     $ ssh root@192.168.1.102

   NuDeveloper Ecosystem – Make engineers' jobs easier.   **nuvoTon**

谢谢
謝謝
Děkuji
Bedankt
Thank you
Kiitos
Merci
Danke
Grazie
ありがとう
감사합니다
Dziękujemy
Obrigado
Спасибо
Gracias
Teşekkür ederim
Cảm ơn

Joy of innovation
nuvoTon