SSH AGENT行于WINDOWS之上

张奇夫

背景

- Windows 上存在多种 ssh client 实现
  - Windows OpenSSH
  - Putty
  - Cygwin OpenSSH
  - WSL OpenSSH
  - ……
- ssh agent 在 *nix 上的实现使用到了 Unix 域套接字
- 由于缺少 Unix 域套接字，这些实现都几乎自己造了一套轮子来替代
- 这些轮子实现的通信方式几乎相互不兼容

AGENT 协议

- 基于 packet
- 请求-回应
- 由客户端单方面驱动，服务器永不请求数据

[draft-miller-ssh-agent-01]
3. Protocol Overview
The agent protocol is a packetised request-response protocol, solely driven by the client. It consists of a number of requests sent from the client to the server and a set of reply messages that are sent in response. At no time does the server send messages except in response to a client request. Replies are sent in order.

All values in the agent protocol are encoded using the SSH wire representations specified by [RFC4251]. Messages consist of a length, type and contents.

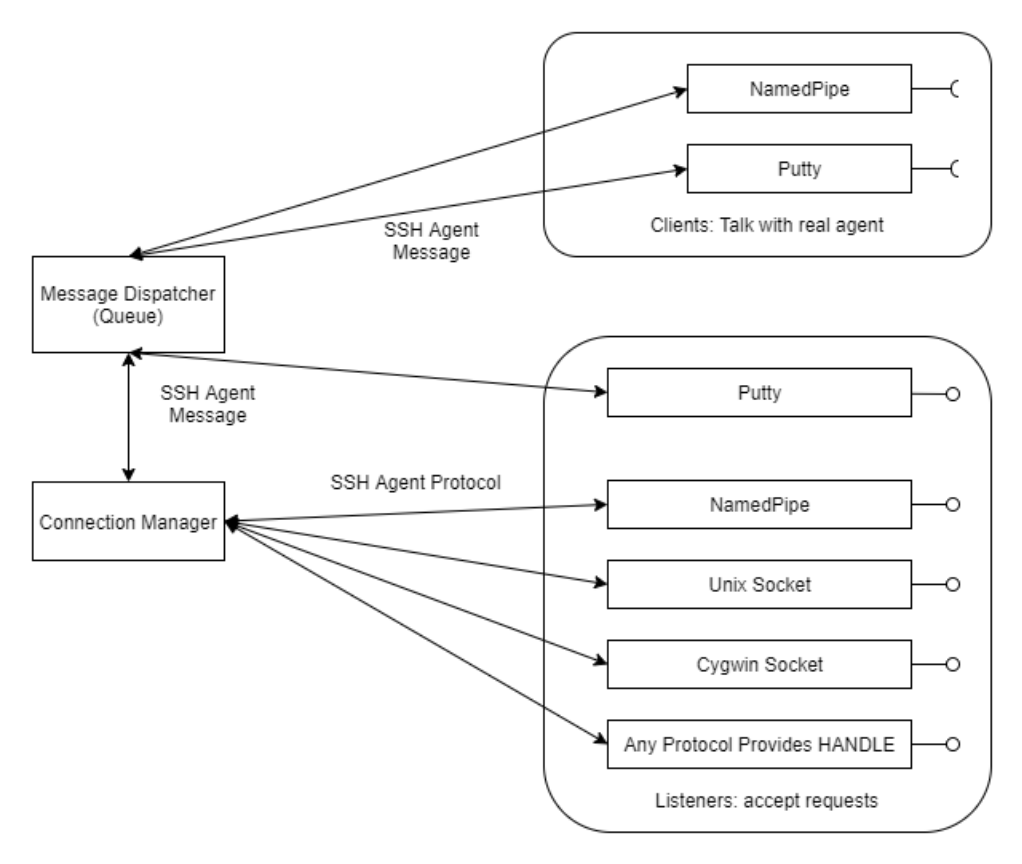
```
uint32                  message length
byte                    message type
byte[message length - 1]  message contents
```

AGENT 实现

- Windows OpenSSH 使用 Named Pipe 进行通信
- Putty 使用 WM_COPYDATA 窗口消息和共享内存进行通信
- Cygwin 使用 TCP 连接进行通信
- WSL 中的 OpenSSH 使用标准 Unix 域套接字
  - WSL1 中可以与 Windows 侧创建的套接字互操作，但 WSL2 移除了这项支持
- GPG4Win 提供的 ssh agent 有两种方式
  - libassuan 实现的套接字，但是 broken 很久了
  - Putty 兼容的方式，替代前者

- 虽然通信方式千差万别，但是以上几种使用的协议都是**相同**的

程序结构



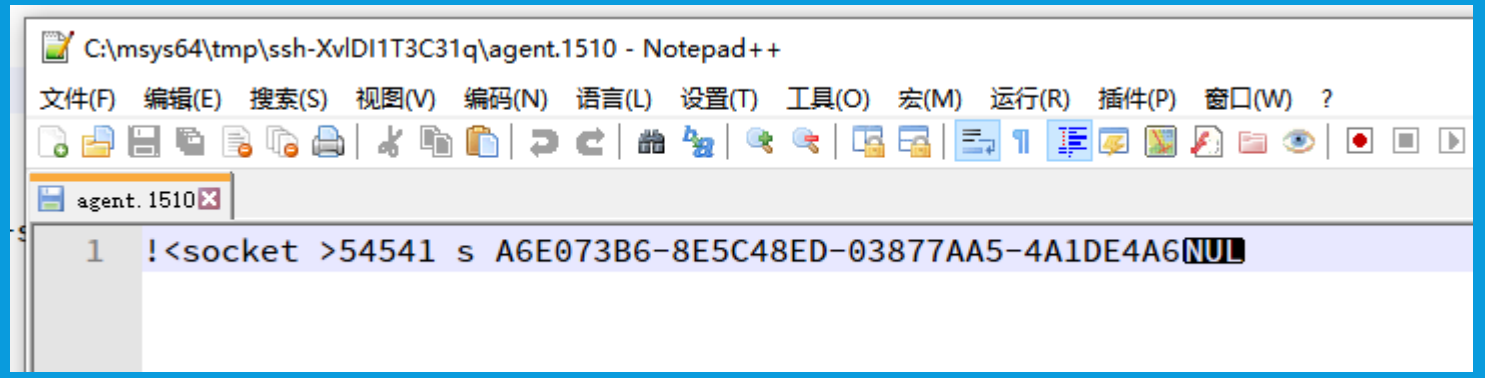
- 除了 Putty 以外，其他方式存在并发的可能
- 某些方式在建立底层连接过后需要额外步骤才能进行正常通信

实现细节：命名管道

- 位置：\\.\pipe\openssh-ssh-agent
- 连接后立即可以进行标准 agent 协议的通信
- 硬编码
- 全局
  - 在 Windows OpenSSH 的实现中，Impersonate 到对应的用户的上下文后读取注册表中存储的私钥

实现细节：CYGWIN SOCKET

- 位置：自行指定
- 连接后需要额外的通信交换信息
- 将所需信息存储在文件中
  - TCP 监听的端口
  - 认证使用的 nonce



实现细节：UNIX DOMAIN SOCKET

- 位置：自行指定
- 连接后立即可以进行标准 agent 协议的通信
- 在系统上呈现为一个文件
- 仅 WSL1 可用

救救 WSL2

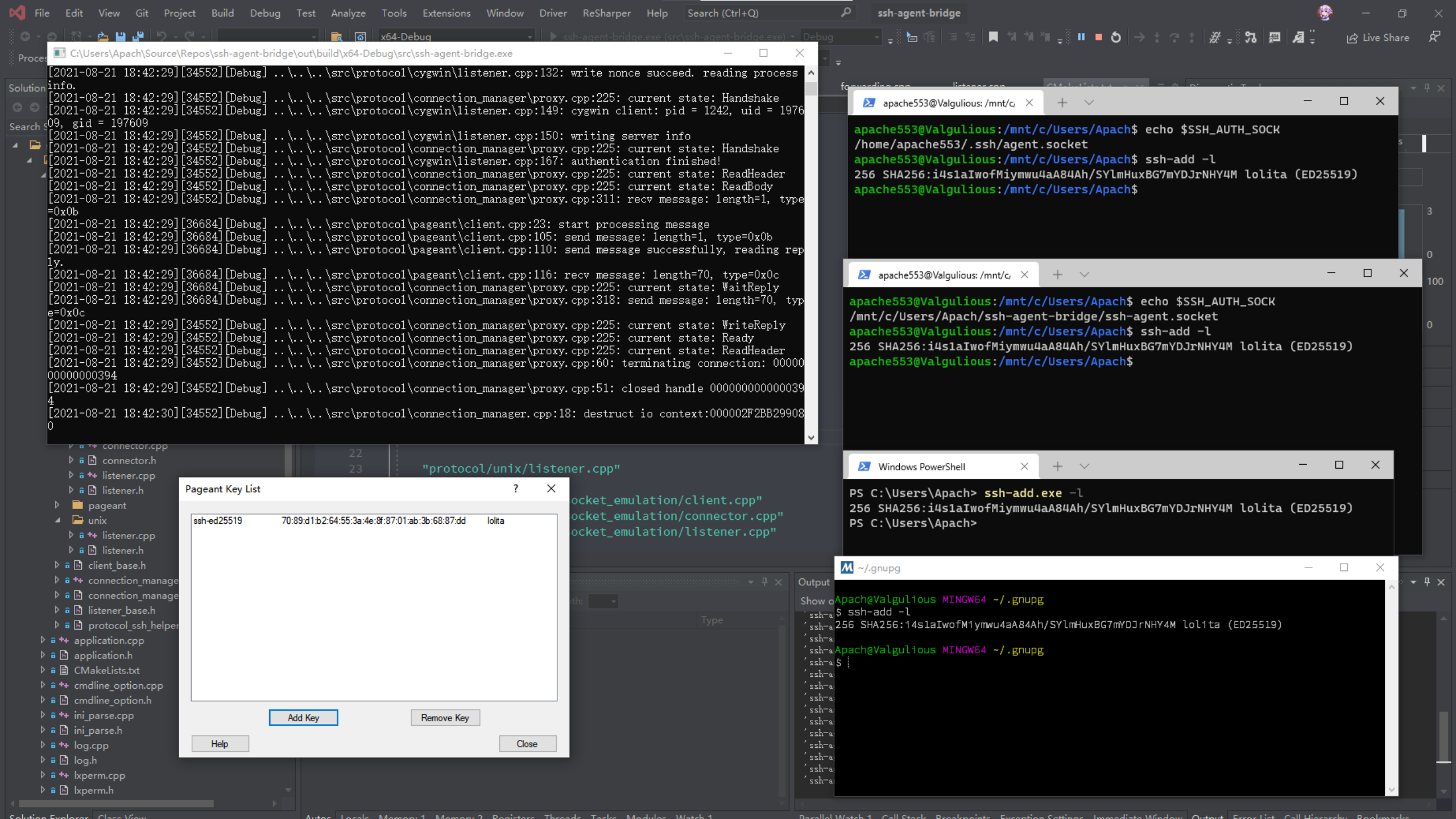
- 运行在虚拟机内，几乎隔离
- 如何与宿主机通信?
  - 网络
    - 使用类似 Cygwin Socket 的方式，读取某文件内容后获取必要信息进行连接和验证
    - 需要找到宿主机的 IP 地址
  - Hyper-V Socket
    - 需要写注册表
    - 需要找到 WSL2 虚拟机的 VmId
      - WSL 是每个用户独有的，系统中可能同时存在多个 WSL 虚拟机
- 需要一个运行在 WSL 中的辅助 daemon

实现细节：HYPERV SOCKET

- 位置：自行指定
- 连接后立即可以进行标准 agent 协议的通信
- 寻找 VmId
  - hcsdiag.exe list 输出结果中存在虚拟机的 VmId ✕
    - 无法分辨虚拟机的用户归属
    - 难以检测虚拟机的创建与删除
  - wslhost.exe 的命令行中可能存在虚拟机的 VmId
    - wslhost.exe 属于对应的用户
    - 可以通过 WMI 事件通知来监测

安全

- 目标
  - 防止其他用户未经授权的访问
  - 防止低权限程序访问
  - 阻止 WSL 中的其他用户访问（如 nobody ）
- 实现
  - Windows ACL
    - 设置对其他用户的拒绝权限
    - 设置完整性等级标记
  - LXSS POSIX 权限
    - 以 NTFS Extended Attributes 形式存储
    - Unix 套接字的例外情况


File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Driver ReSharper Help
Search (Ctrl+Q)
ssh-agent-bridge
Live Share
x64-Debug
C:\Users\Apach\Source\Repos\ssh-agent-bridge\out\build\x64-Debug\src\ssh-agent-bridge.exe
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\cygwin\listener.cpp:132: write nonce succeed. reading process info.
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: Handshake
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\cygwin\listener.cpp:149: cygwin client: pid = 1242, uid = 197609, gid = 197609
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\cygwin\listener.cpp:150: writing server info
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: Handshake
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\cygwin\listener.cpp:167: authentication finished!
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: ReadHeader
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: ReadBody
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:311: recv message: length=1, type=0x0b
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\pageant\client.cpp:23: start processing message
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\pageant\client.cpp:105: send message: length=1, type=0x0b
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\pageant\client.cpp:110: send message successfully, reading reply.
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\pageant\client.cpp:116: recv message: length=70, type=0x0c
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: WaitReply
[2021-08-21 18:42:29][36684][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:318: send message: length=70, type=0x0c
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: WriteReply
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: Ready
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:225: current state: ReadHeader
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:60: terminating connection: 0000000000000394
[2021-08-21 18:42:29][34552][Debug] ..\..\..\src\protocol\connection_manager\proxy.cpp:51: closed handle 0000000000000394
[2021-08-21 18:42:30][34552][Debug] ..\..\..\src\protocol\connection_manager.cpp:18: destruct io context:000002F2BB299080
apache553@Valgulious: /mnt/c/
apache553@Valgulious:/mnt/c/Users/Apach$ echo $SSH_AUTH_SOCK
/home/apache553/.ssh/agent.socket
apache553@Valgulious:/mnt/c/Users/Apach$ ssh-add -l
256 SHA256:i4s1aIwofMiymwu4aA84Ah/SYlmHuxBG7mYDJrNHY4M lolita (ED25519)
apache553@Valgulious:/mnt/c/Users/Apach$
apache553@Valgulious: /mnt/c/
apache553@Valgulious:/mnt/c/Users/Apach$ echo $SSH_AUTH_SOCK
/mnt/c/Users/Apach/ssh-agent-bridge/ssh-agent.socket
apache553@Valgulious:/mnt/c/Users/Apach$ ssh-add -l
256 SHA256:i4s1aIwofMiymwu4aA84Ah/SYlmHuxBG7mYDJrNHY4M lolita (ED25519)
apache553@Valgulious:/mnt/c/Users/Apach$
Windows PowerShell
PS C:\Users\Apach> ssh-add.exe -l
256 SHA256:i4s1aIwofMiymwu4aA84Ah/SYlmHuxBG7mYDJrNHY4M lolita (ED25519)
PS C:\Users\Apach>
~/.gnupg
Apach@Valgulious MINGW64 ~/.gnupg
$ ssh-add -l
256 SHA256:i4slaIwofMiymwu4aA84Ah/SYlmHuxBG7mYDJrNHY4M lolita (ED25519)
Apach@Valgulious MINGW64 ~/.gnupg
$
Pageant Key List
ssh-ed25519 70:89:d1:b2:64:55:3a:4e:8f:87:01:ab:3b:68:87:dd lolita
Add Key
Remove Key
Help
Close
"protocol/unix/listener.cpp"
ocket_emulation/client.cpp"
ocket_emulation/connector.cpp"
ocket_emulation/listener.cpp"
connector.cpp
connector.h
listener.cpp
listener.h
pageant
unix
listener.cpp
listener.h
client_base.h
connection_manage
connection_manage
listener_base.h
protocol_ssh_helper
application.cpp
application.h
CMakeLists.txt
cmdline_option.cpp
cmdline_option.h
ini_parse.cpp
ini_parse.h
log.cpp
log.h
lxperm.cpp
lxperm.h

谢谢