Table of Contents

Secure Shell (SSH)

Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. The best known example application is for remote login to computer systems by users.

SSH uses public-key cryptography to authenticate the remote computer and allow it to authenticate the user, if necessary. There are several ways to use SSH one is to use automatically generated public-private key pairs to simply encrypt a network connection, and then use password authentication to log on.

Another is to use a manually generated public-private key pair to perform the authentication, allowing users or programs to log in without having to specify a password. In this scenario, anyone can produce a matching pair of different keys (public and private). The public key is placed on all computers that must allow access to the owner of the matching private key (the owner keeps the private key secret). While authentication is based on the private key, the key itself is never transferred through the network during authentication. SSH only verifies whether the same person offering the public key also owns the matching private key. In all versions of SSH it is important to verify unknown public keys, i.e. associate the public keys with identities, before accepting them as valid. Accepting an attacker’s public key without

validation will authorize an unauthorized attacker as a valid user.

History and development

Version 1.x

In 1995, Tatu Ylönen, a researcher at Helsinki University of Technology, Finland, designed the first version of the protocol (now called SSH-1) prompted by a password-sniffing attack at his university network. The goal of SSH was to replace the earlier rlogin, TELNET and rsh protocols, which did not provide strong authentication nor guarantee confidentiality.

Ylönen released his implementation as freeware in July 1995, and the tool quickly gained in popularity. Towards the end of 1995, the SSH user base had grown to 20,000 users in fifty countries.

In December 1995, Ylönen founded SSH Communications Security to market and develop SSH. The original version of the SSH software used various pieces of free software, such as GNU libgmp, but later versions released by SSH Communications Security evolved into increasingly proprietary software.

It is estimated that, as of 2000, there were 2 million users of SSH.

Version 1.99

In January 2006, well after version 2.1 was established, RFC 4253 specified that an SSH server which supports both 2.0 and prior versions of SSH should identify its protoversion as 1.99. This is not an actual version but a method to identify backward compatibility.

OpenSSH and OSSH

In 1999, developers, wanting a free software version to be available, went back to the older 1.2.12 release of the original SSH program, which was the last released under an open source license. Björn Grönvall's OSSH was subsequently developed from this codebase. Shortly thereafter, OpenBSD developers forked Grönvall's code and did extensive work on it, creating OpenSSH, which shipped with the 2.6 release of OpenBSD. From this version, a "portability" branch was formed to port OpenSSH to other operating systems.

As of 2005, OpenSSH was the single most popular SSH implementation, coming by default in a large number of operating systems. OSSH meanwhile has become obsolete. OpenSSH continues to be maintained and now supports both 1.x and 2.0 versions.

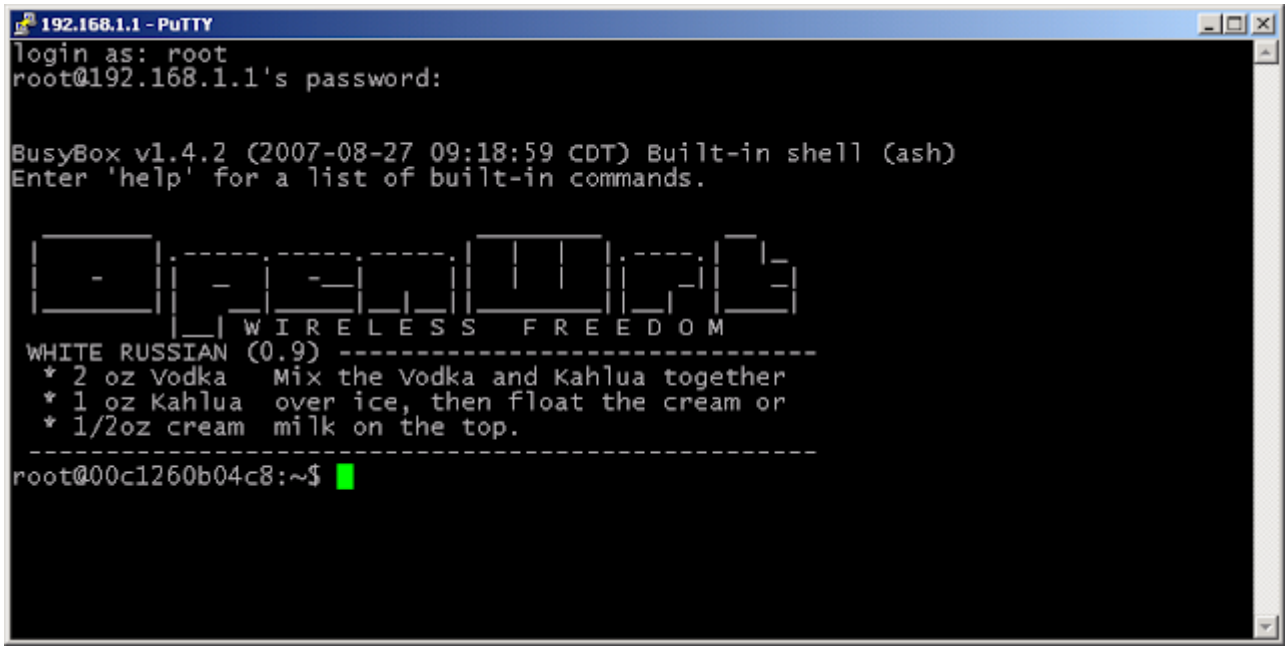
Version 2.x

"Secsh" was the official Internet Engineering Task Force's (IETF) name for the IETF working group responsible for version 2 of the SSH protocol.[19] In 2006, a revised version of the protocol, SSH-2, was adopted as a standard. This version is incompatible with SSH-1. SSH-2 features both security and feature improvements over SSH-1. Better security, for example, comes through Diffie–Hellman key exchange and strong integrity checking via message authentication codes. New features of SSH-2 include the ability to run any number of shell sessions over a single SSH connection. Due to SSH-2's superiority and popularity over SSH-1, some implementations such as Lsh and Dropbear support only the SSH-2 protocol.

Uses

1. For login to a shell on a remote host (replacing Telnet and rlogin)
2. For executing a single command on a remote host (replacing rsh)

3. For setting up automatic (passwordless) login to a remote server (for example, using OpenSSH)
4. Secure file transfer.
5. In combination with rsync to back up, copy and mirror files efficiently and securely.



Logging into OpenWrt via SSH using PuTTY running on Windows.

Related posts:

1. Types of Attack
2. Security threats
3. Computer and cyber security
4. Introduction to network security
5. Intrusion detection tool
6. Categories of security assessments
7. Security terminologies and principals
8. Intoduction to intrusion

9. Intrusion detection tool
10. Categories of security assessments
11. Intrusion terminology
12. Cryptography attacks
13. Cryptography
14. MD5
15. Message digest functions
16. Digital signature
17. Authentication Functions
18. One way hash function
19. Hash function in network web security
20. Digital signature standard
21. SSL Secure socket layer