B.7: Remote Console - SSH/X Forwarding -

Console Redirection

Most Linux distributions can be configured for connections via the serial console port, which enables administration of a machine even if it has no keyboard, mouse, monitor, or network attached to it. A basic environment for this scenario is two machines connected using a serial cable (9-pin connector cable). The administering machine can be any Unix/Linux or Windows machine with a terminal emulator program (PuTTY or Minicom, for example).

In situations where a systems administrator may not have easy access to the Linux system, it can be configured to allow for remote administrative access, including booting the system, even providing access to the BIOS/UEFI. This configuration is often employed in a data center setting, where there may be numerous servers without a physical keyboard/monitor/mouse.

SSH Tunneling

Port forwarding via SSH (SSH tunneling) creates a secure connection between a local computer and a remote machine through which services can be relayed. Because the connection is encrypted, SSH tunneling is useful for transmitting information that uses an unencrypted protocol, such as IMAP, VNC, or IRC.

Types of Port Forwarding

SSH's port forwarding feature can smuggle various types of Internet traffic into or out of a network. This can be used to avoid network monitoring or sniffs, or bypass badly configured routers on the Internet. Note: You might also need to change the settings in other programs (like your web browser) in order to circumvent these filters.
There are three types of port forwarding with SSH:

- Local port forwarding: connections from the SSH client are forwarded via the SSH server, then to a destination server
- Remote port forwarding: connections from the SSH server are forwarded via the SSH client, then to a destination server
- Dynamic port forwarding: connections from various programs are forwarded via the SSH client, then via the SSH server, and finally to several destination servers

Local port forwarding is the most common type. For example, local port forwarding lets you bypass a company firewall that blocks Wikipedia.

Remote port forwarding is less common. For example, remote port forwarding lets you connect from your SSH server to a computer on your company's intranet.

Dynamic port forwarding is rarely used. For example, dynamic port forwarding lets you bypass a company firewall that blocks web access altogether. Although this is very powerful, it takes a lot of work to set up, and it's usually easier to use local port forwarding for the specific sites you want to access.

Port-forwarding is a widely supported technique and a feature found in all major SSH clients and servers, although not all clients do it the same way. For help on using a specific client, consult the client's documentation. For example, the PuTTY manual has a section on port forwarding in PuTTY.

To use port forwarding, you need to make sure port forwarding is enabled in your server. You also need to tell your client the source and destination port numbers to use. If you're using local or remote forwarding, you need to tell your client the destination server. If you're using dynamic port forwarding, you need to configure your programs to use a SOCKS proxy server. Again, exactly how to do this depends on which SSH client you use, so you may need to consult your documentation.

**X Forwarding**

The X Window System (also known as X11, or just X) is a software package and network protocol that lets you interact locally, using your personal computer's display, mouse, and keyboard, with the graphical user interface (GUI) of an application running on a remote networked computer.

X forwarding can be used in an SSH session on one computer to securely run graphical applications (X clients) located on remote systems.

**Accessibility Options**

It is possible to make the Linux operating system usable by as many people as possible across ages, languages and physical abilities. This includes providing an accessible platform with high quality assistive tools, and ensuring that other applications work well with these. The assistive tools on Linux are often part of the default installation.
Options Common to Linux

- screen reader - Orca is common
- high contrast themes, complete with custom icons and mouse cursor
- keyboard modifiers including on-screen keyboards
- it is possible to control applications using voice commands. Julius speech recognition engine can be used to recognize voice commands and execute pre-set commands.
- for users that cannot click with a hardware button, there is an utility named Mousetweaks that enables them to perform the various clicks by software

For systems that may not include accessibility tools as part of the default install, most of the tools are open source and freely available for download and install.

Adapted from:
"SSH/OpenSSH/PortForwarding" by Pasi Lallinaho, CommunityHelpWiki is licensed under CC BY-SA 4.0
"Accessibility" by Christopher M. Penalver, CommunityHelpWiki is licensed under CC BY-SA 4.0