Remote Desktop Protocols in Linux

Remote desktop protocols provide graphical remote access to systems across Windows, Linux, and macOS. These protocols enable administrators to manage, troubleshoot, and update systems remotely using appropriate tools depending on the OS.

🔑 Common Protocols

RDP (Remote Desktop Protocol)

- **Platform**: Primarily Windows
- **Purpose**: Remote GUI access to Windows desktops
- **Analogy**: Key made for Windows buildings

VNC (Virtual Network Computing)

- **Platform**: Popular in Linux (cross-platform)
- **Purpose**: GUI access to remote desktops
- **Analogy**: Universal key often used for Linux systems
- **Port**: TCP/5900 + display number (e.g., :1 → 5901)

🖥️ X11 / XServer

What is X11?

- Protocol suite for GUI on Unix/Linux
- Predominant in Unix systems, also supported on other OSs
- **Local rendering**: Unlike VNC/RDP, rendering happens on the local machine
- **Port Range**: TCP/6000–6009 (e.g., :0 = 6000)

X11 Forwarding

- Use SSH to securely forward GUI apps
- Enable in `/etc/ssh/sshd_config`:

```
X11Forwarding yes
```

- Run remote GUI app:

```
ssh -X user@remote-ip /usr/bin/firefox
```

Security Concerns

- Default communication is unencrypted
- Vulnerable to packet sniffing and MITM attacks
- Tools like `xwd`, `xgrabsc` can capture window content
- Notable vulnerabilities:
  - CVE-2017-2624, CVE-2017-2625, CVE-2017-2626

🧭 XDMCP (X Display Manager Control Protocol)

- **Port**: UDP/177
- **Function**: Remote X sessions with full GUI redirection
- **Security**: Unencrypted; vulnerable to MITM
- **Use Case**: Access full desktop environments (GNOME, KDE) remotely

---

🖱️ VNC (Virtual Network Computing)

- **Protocol**: RFB (Remote Frame Buffer)
- **Usage**: Remote GUI access, screen sharing, troubleshooting
- **Port**: TCP/5900 + display (e.g., :1 = 5901)
- **Security**: Encrypted with authentication; best used with SSH tunnels

VNC Tools

- TigerVNC
- TightVNC
- RealVNC
- UltraVNC

Two VNC Modes

1. Share the actual host screen (local keyboard/mouse usable)
2. Login to virtual session (like terminal server)

TigerVNC Setup

Install:

```
sudo apt install xfce4 xfce4-goodies tigervnc-standalone-server -y
vncpasswd
```

Create config:

```
touch ~/.vnc/xstartup ~/.vnc/config

# xstartup
cat <<EOT >> ~/.vnc/xstartup
#!/bin/bash
```

```
unset SESSION_MANAGER
unset DBUS_SESSION_BUS_ADDRESS
/usr/bin/startxfce4
[ -x /etc/vnc/xstartup ] && exec /etc/vnc/xstartup
[ -r $HOME/.Xresources ] && xrdb $HOME/.Xresources
x-window-manager &
EOT

# config
cat <<EOT >> ~/.vnc/config
geometry=1920x1080
dpi=96
EOT

chmod +x ~/.vnc/xstartup
```

Start server:

```
vncserver
```

Check sessions:

```
vncserver -list
```

🔐 SSH Tunneling for VNC

Create a tunnel:

```
ssh -L 5901:127.0.0.1:5901 -N -f -l htb-student 10.129.14.130
```

Connect:

```
xtightvncviewer localhost:5901
```

🛡️ Protocol Comparison

Protocol	Default Port(s)	Encryption	Platform	Notes
RDP	3389	Yes	Windows	Full remote desktop
VNC	5900+display	Partial	Cross-platform	Use SSH for secure tunnel

Protocol	Default Port(s)	Encryption	Platform	Notes
X11	6000–6009	No	Unix/Linux	Use SSH X11 forwarding
XDMCP	UDP 177	No	Unix/Linux	Avoid in secure environments