

Remotely and Securely Controlling Other Computers From Your Mac for Administrative or Support Purposes

Aaron Adams

Who is this guy?

Why are we doing this?

What we'll learn

- Remote desktop technologies overview
- Concentration on included or free tools
- Server applications
- Client applications
- Secure connections via VPN
- Secure connections via SSH tunnel

Remote Desktop Technologies

Microsoft Remote Desktop Protocol

- Based on ITU-T T.128 (aka T.SHARE)
- Included with Windows 2000 Server and above (**not** Windows 2000 Professional)
- Controls the console or a separate login session
- Unsecured

Virtual Network Computing

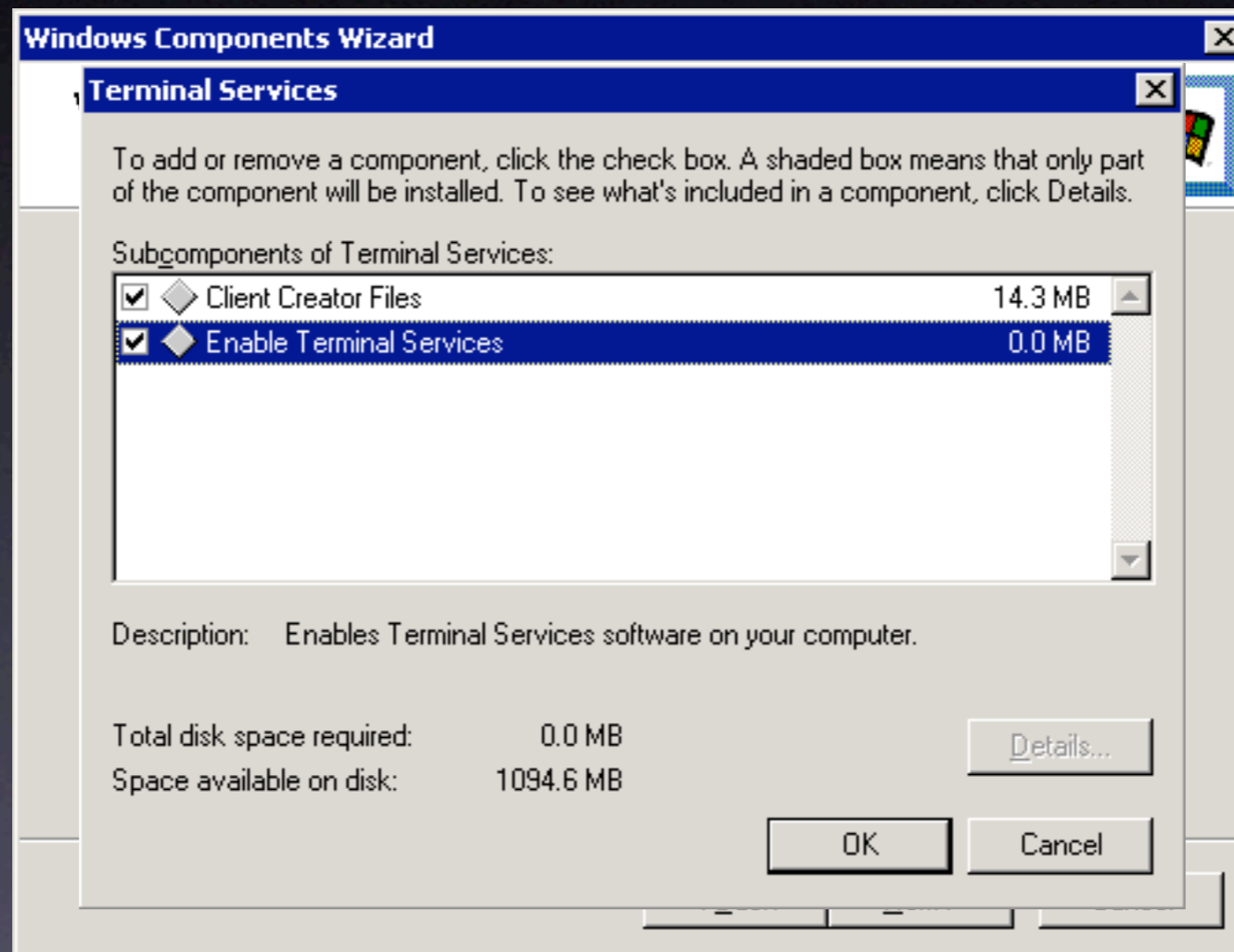
- Screen-scrubber, bandwidth intensive
- Cross-platform (including Windows 2000 Professional)
- Controls the console, which can be shared among multiple users
- Unsecured

Server Applications

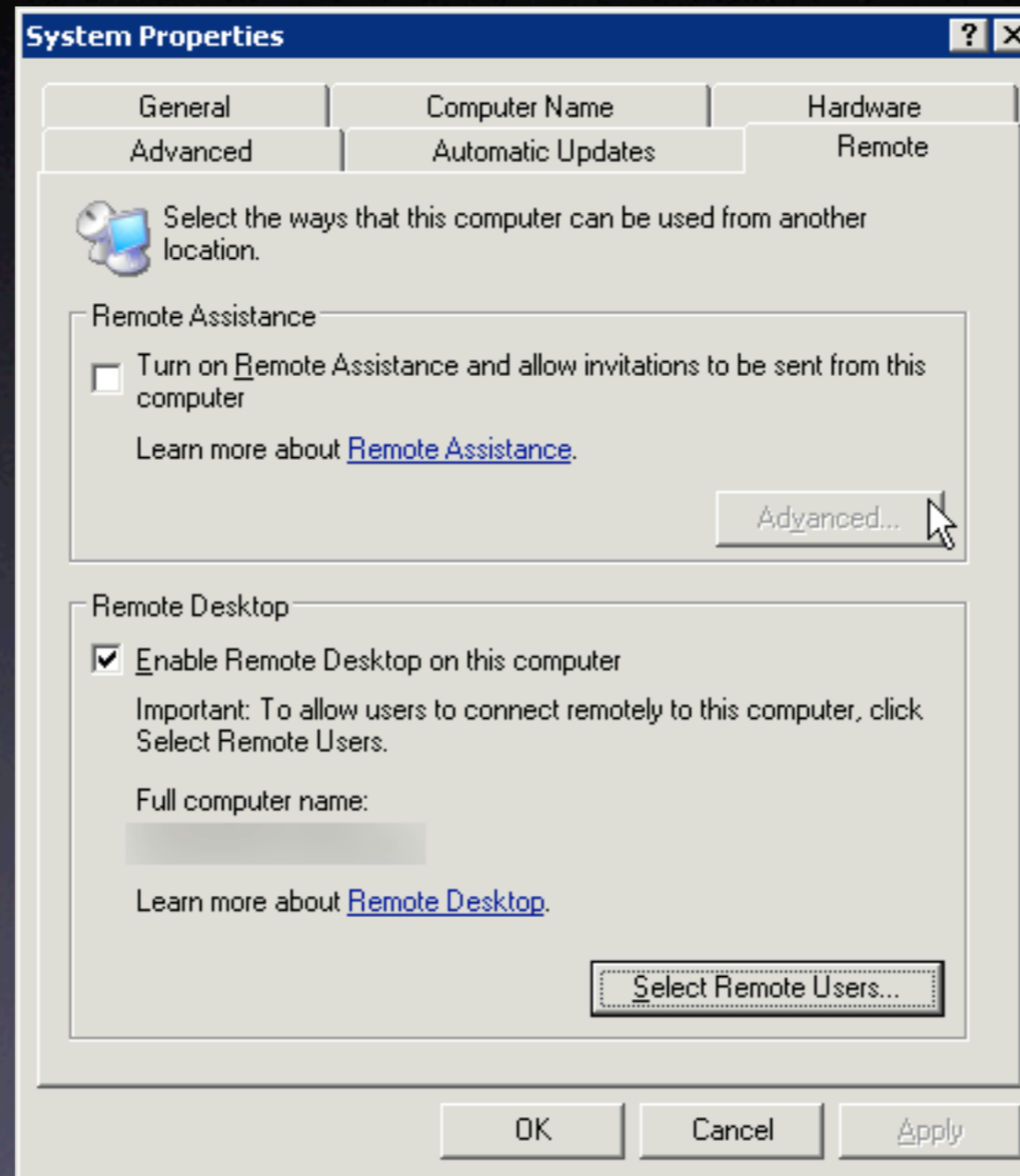
Windows Terminal Services

- Install via Control Panel (Windows 2000 Server)
- Enable in System Properties (Windows Server 2003 and Windows XP)

Terminal Services Install (Windows 2000 Server)



Terminal Services Install (Windows XP and Server 2003)



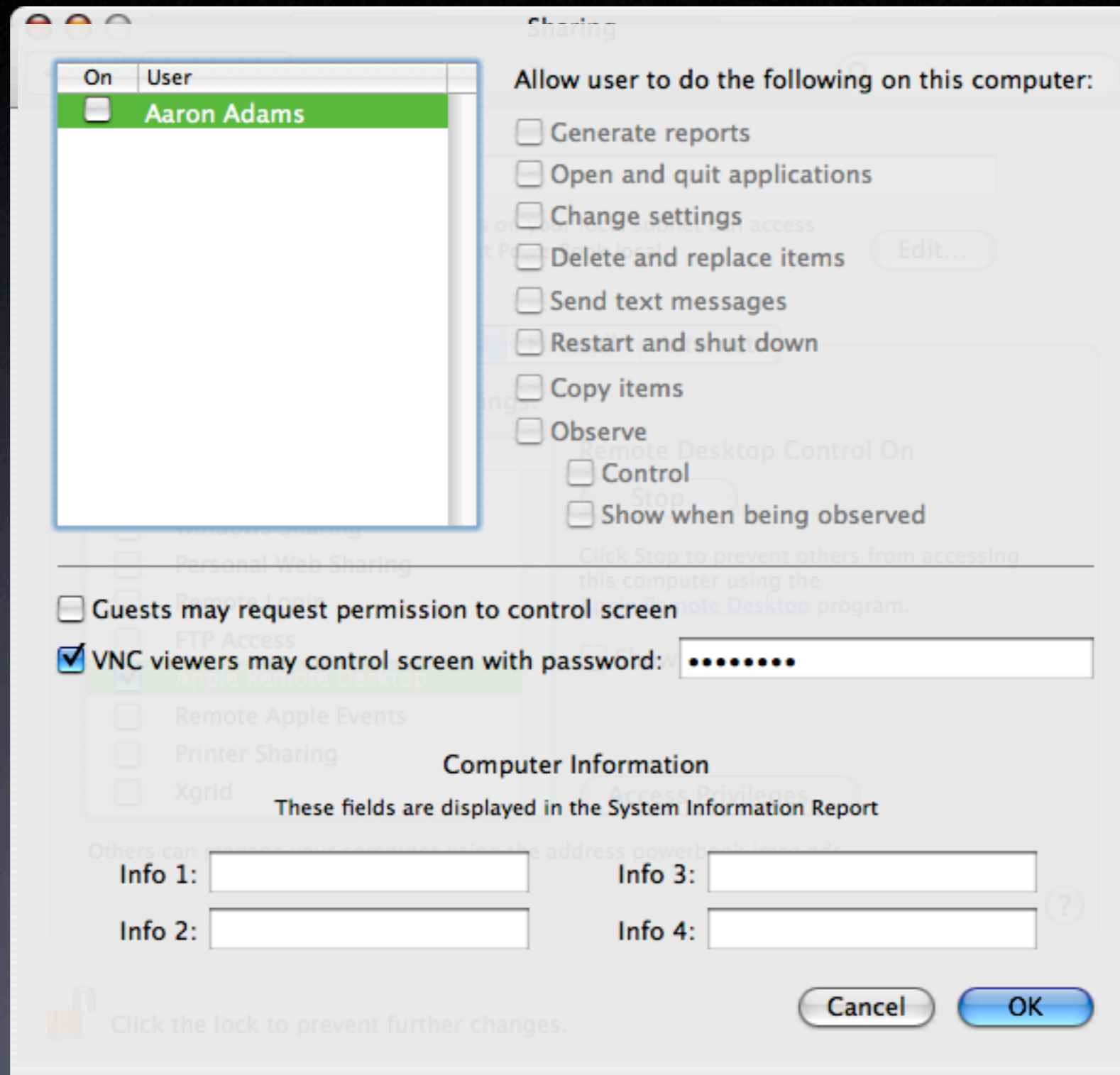
Virtual Network Computing (VNC)

- For Windows: RealVNC, TightVNC, UltraVNC, others.
- For Mac: Apple Remote Desktop, OSXvnc
- For Linux: VNC server included

Apple Remote Desktop

- Comes with Tiger
- Can be used with ARD application or as generic VNC server
- Enable in Sharing System Preference

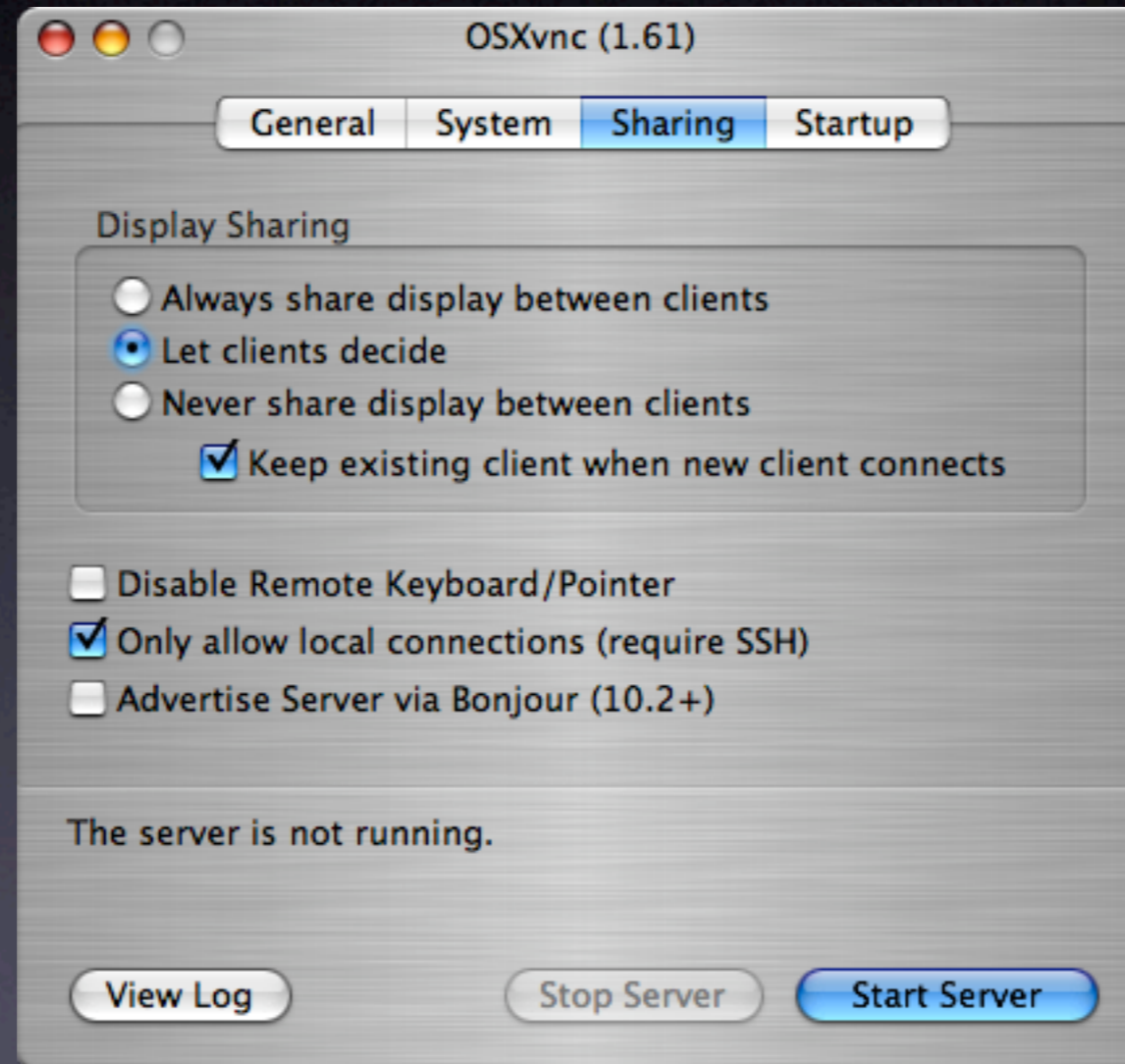
ARD Install and Setup



OSXvnc

- Simple
- Standalone application or startup item
- Secure connection option

OSXvnc options



Client Applications

Microsoft Remote Desktop Connection

- Use to connect to Terminal Services / Remote Desktop
- Good interface and options
- Connection profiles

MSRDC Negatives

- Cannot connect to more than one server at a time
- Failed connect requires you to re-open the application

RDC Launcher

- Applescript
- Calls up LaunchCFMApp to start an additional copy of MSRDC

RDC Menu

- Lives in the menu bar
- Pull-down menu to start additional copies of MSRDC

RDesktop

- Command-line application
- Runs in X11 window system
- Download and compile from source


```
./configure
```

```
make
```

```
sudo make install
```

RDesktop

- Installs to `/usr/local/bin`
- Add to path (`PATH=$PATH:/usr/local/bin`)
in `.profile`
- Write shell script to start


```
#!/bin/bash
```

```
/usr/local/bin/rdesktop \
```

```
-u you \
```

```
-d Domain \
```

```
-g 1024x768 \
```

```
-x 1 \
```

```
-z \
```

```
-a 24 \
```

```
-T "Windows XP" \
```

```
winxp
```

RDesktop

- Associate shell script with X11.app
- Place folder in Dock

Windows VNC Clients

- RealVNC, TightVNC, UltraVNC, usually include client piece as well as server
- Can control Mac or Windows target machines

Chicken of the VNC

- Connection profiles
- “Listen for Server” connects to VNC servers behind a firewall

Security

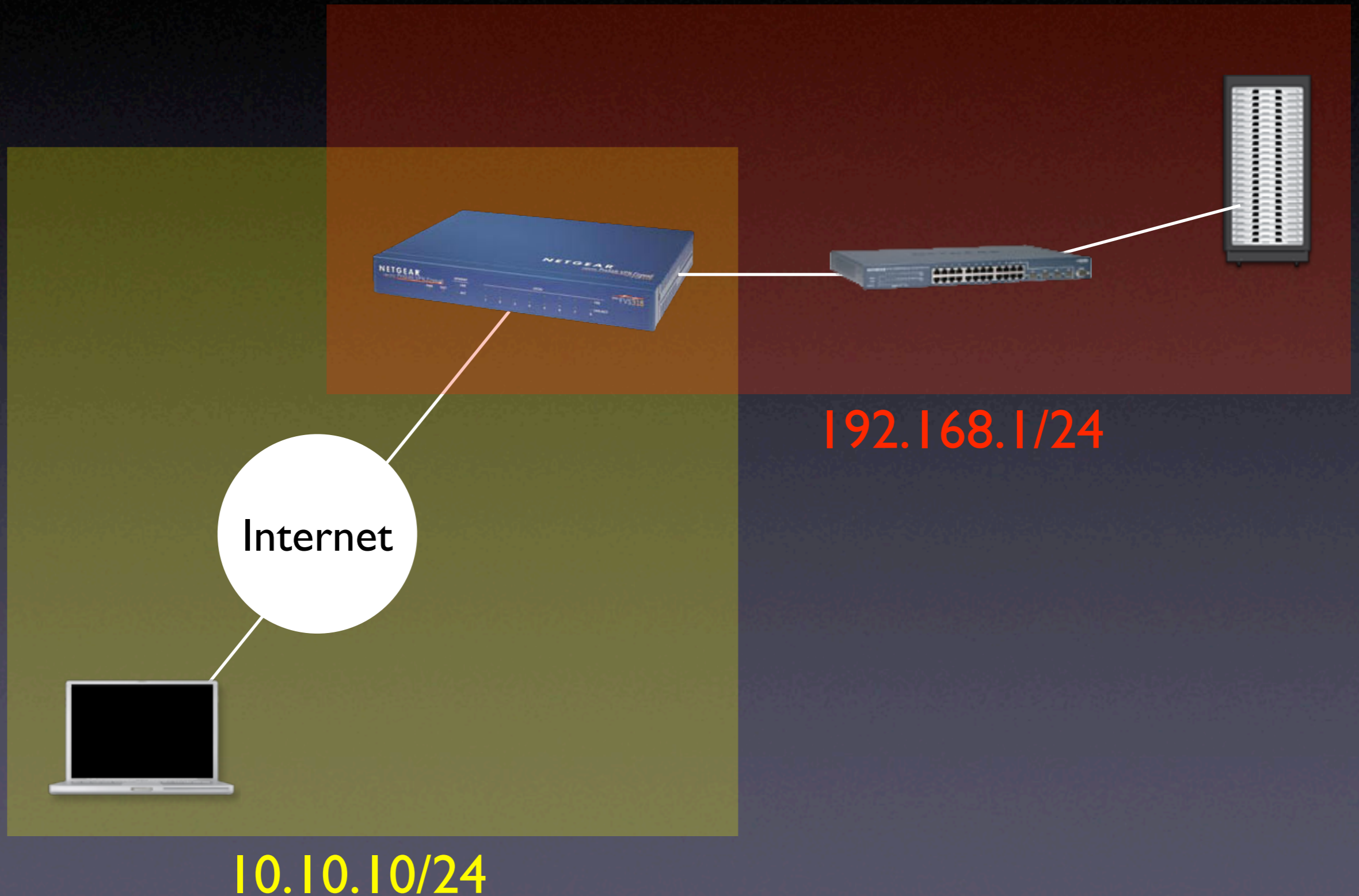
Protocols and Ports

- A port is a logical data stream designated by a number
- Microsoft Terminal Services / Remote Desktop listens on port TCP 3389
- VNC listens on port TCP 5900-5903

IPSec VPN

- No need to open firewall ports
- Low overhead
- Very good security

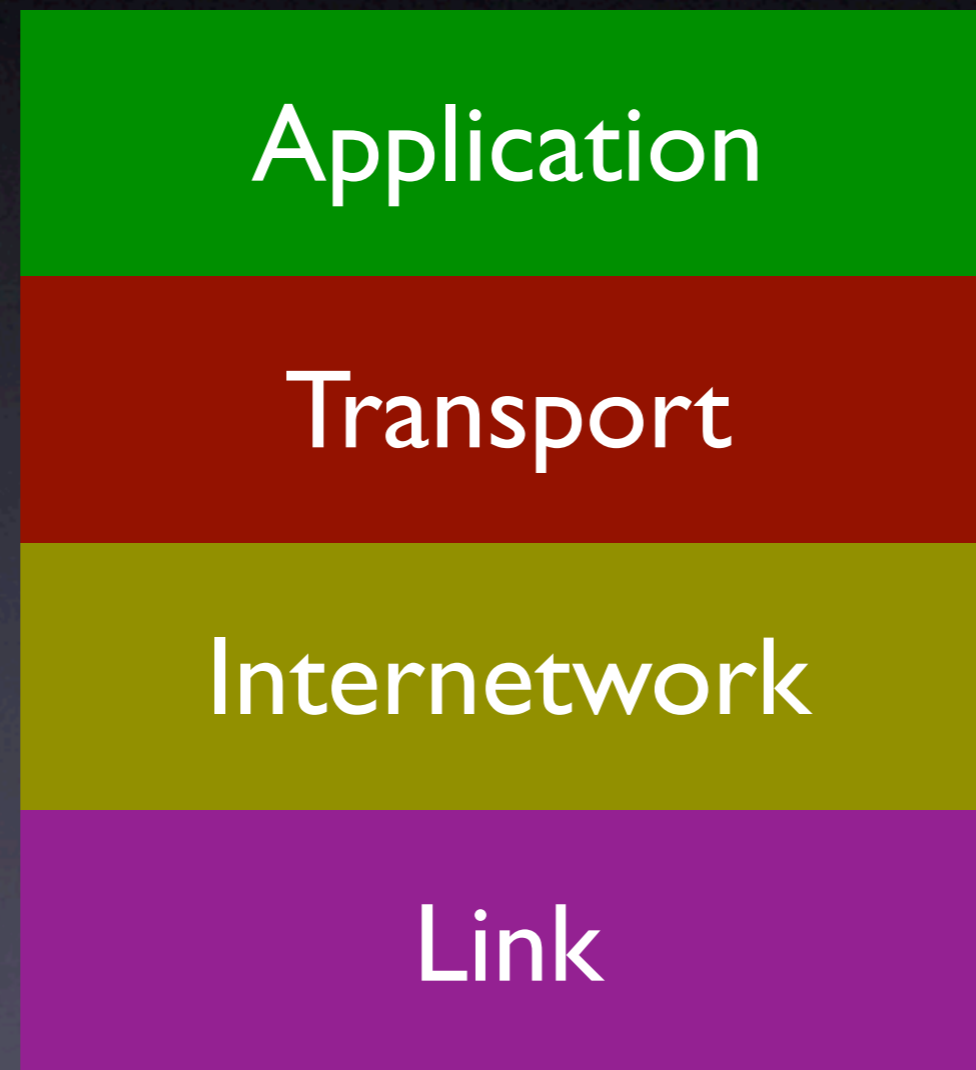
IPSec VPN Diagram



Secure Shell (SSH) Tunnels

- SSH comes with every Mac
- GUIs available
- Good security
- Requires opening only one firewall port for multiple services
- You must have an account on remote machine

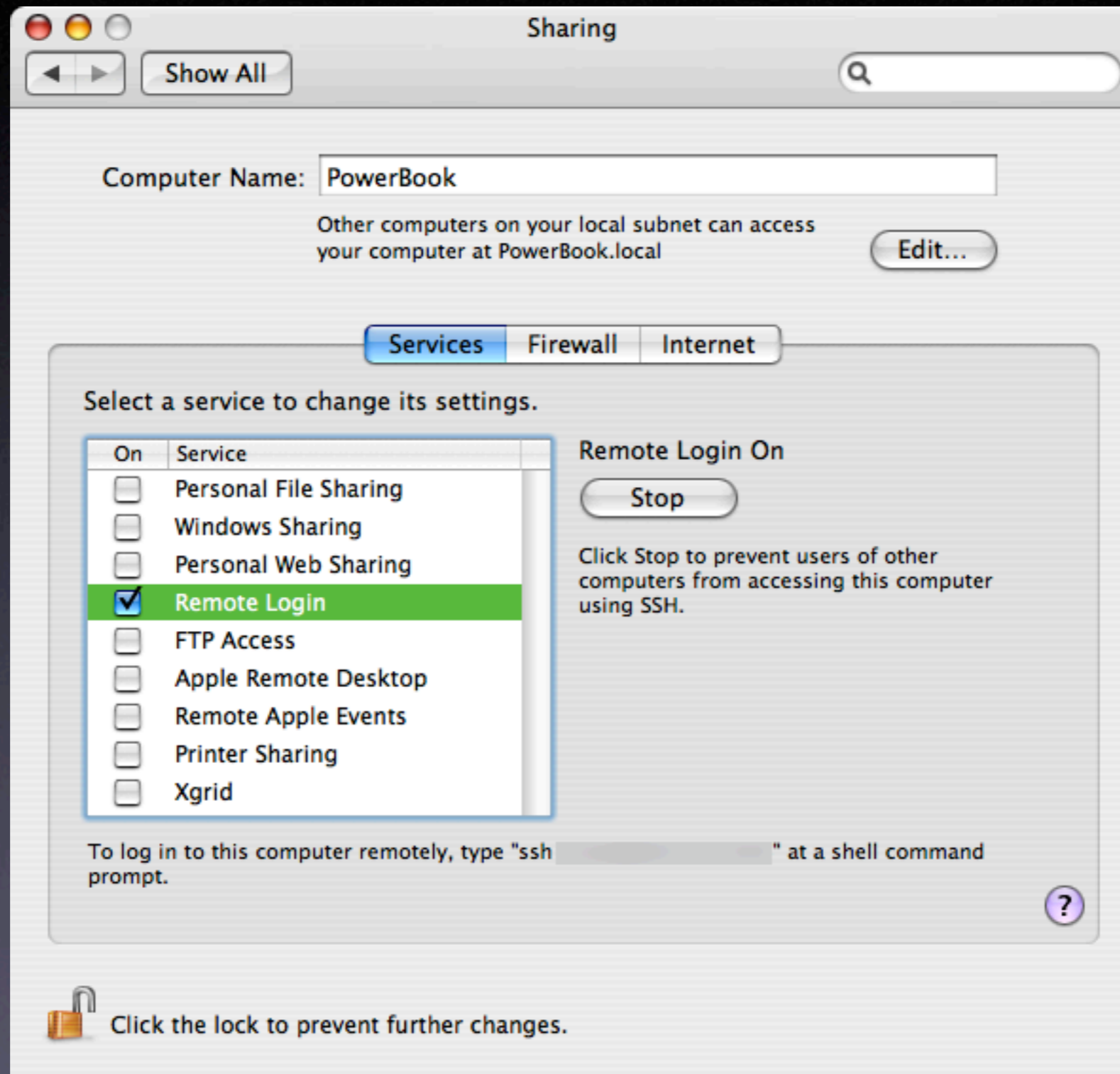
TCP/IP Network Model



Protocol Tunneling



Enable SSH on a Mac



SSH for Windows

- Not included with Windows, must use third party software
- Cygwin is free command line Unix on top of Windows
- Includes SSH client and server

SSH Tunneling Command Line

```
ssh -NfL 5900:localhost.:5900 you@remote.machine.net
```

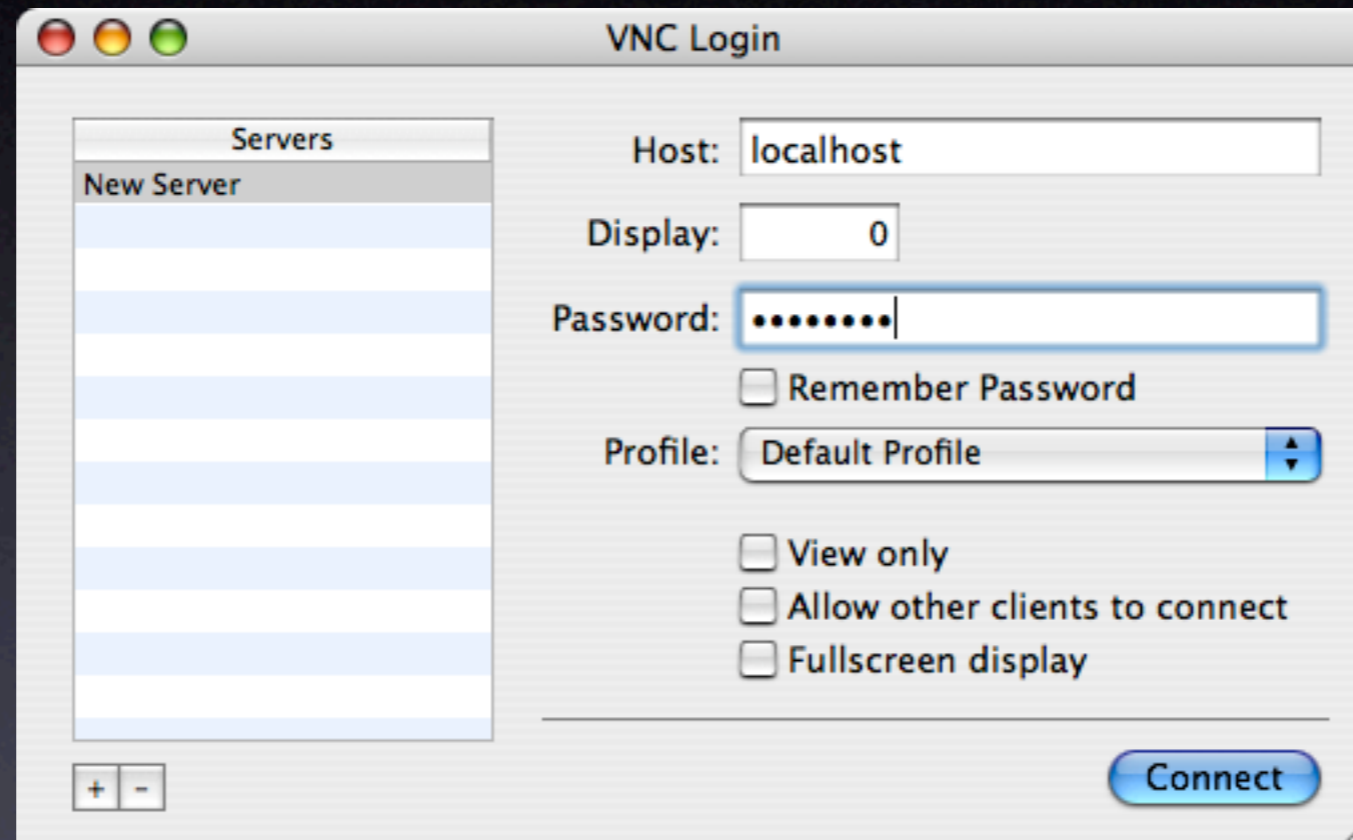
Command	What it does
ssh	SSH command
-N	Do not return a command prompt
f	Run this command in the background
L	Perform the following port redirection
5900:	Local TCP port
localhost.	Remote service host IP / name
:5900	Remote service TCP port
you	SSH login name
@remote.machine.net	IP / name of SSH server

localhost.

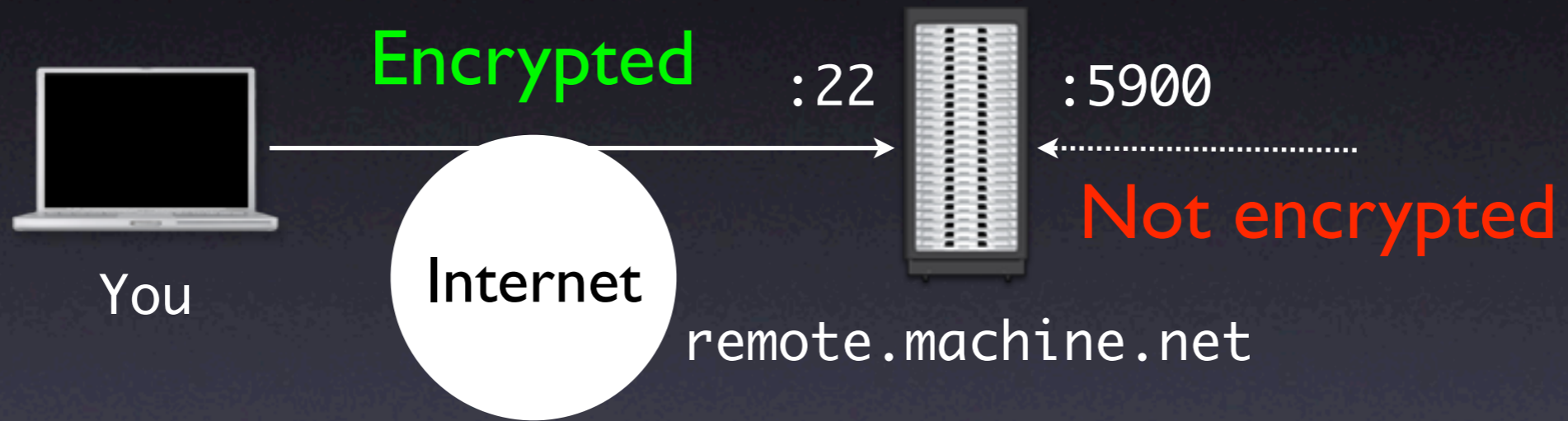


127.0.0.1

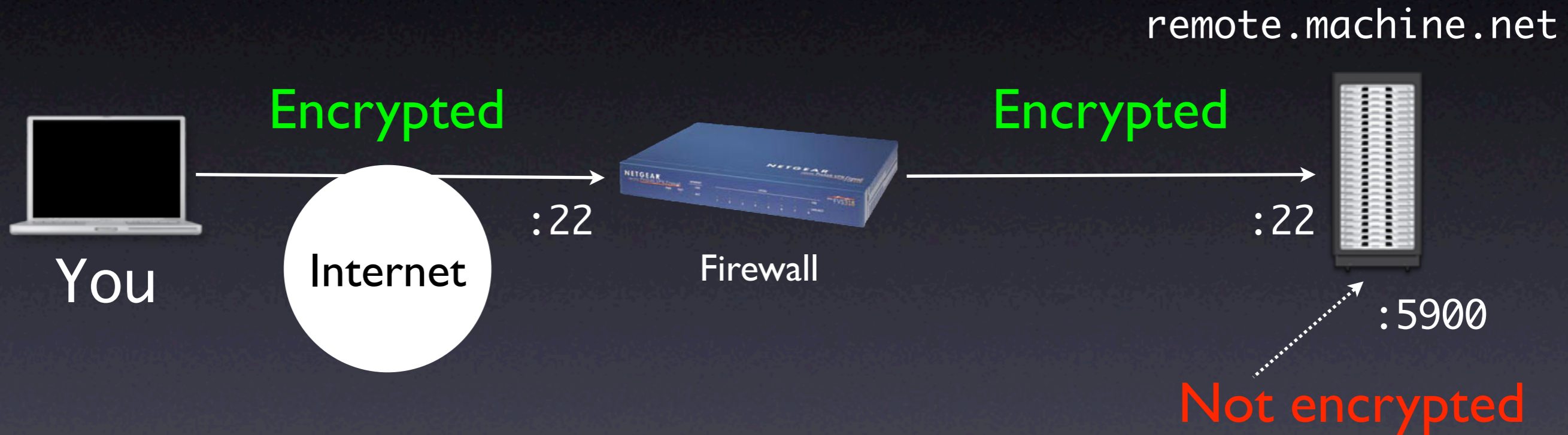
Connecting to the tunnel



A tunnel example graphic



A tunnel example graphic, with firewall



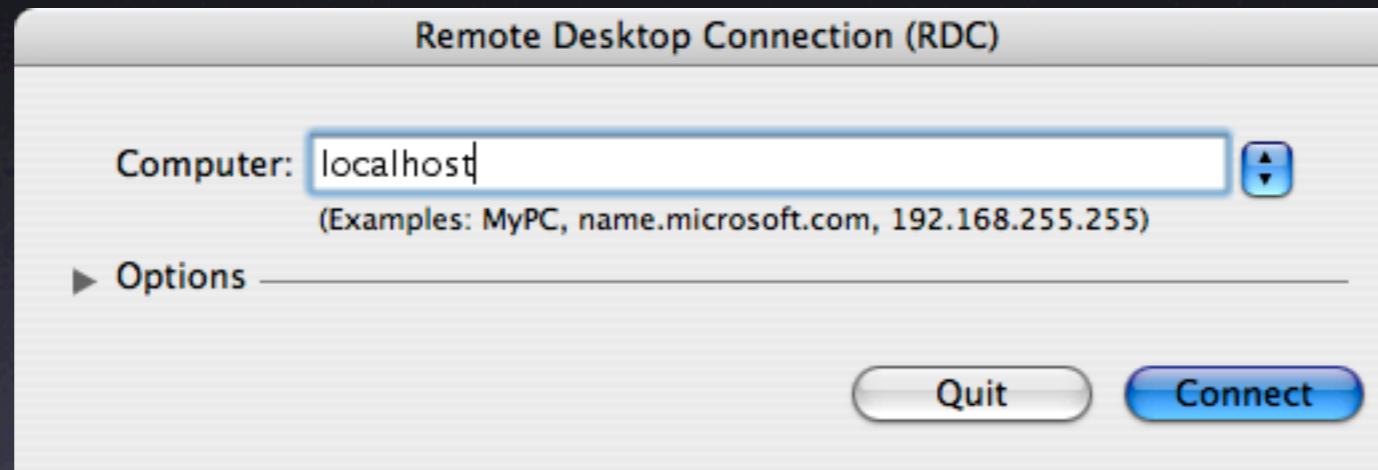
A different tunnel example graphic

```
ssh -NfL 5901:imac.lan.net:5900 you@xserve.lan.net
```



An example RDP tunnel

```
ssh -NfL 3389:localhost.:3389 you@windows.machine.net
```



Optional key-based login

- You will have to enter a password each time you establish a tunnel
- To eliminate the password, create a public / private key pair

```
mkdir ~/.ssh
```

```
cd ~/.ssh
```

```
ssh-keygen -b 1024 -t dsa -f id_dsa -P ''
```

```
touch authorized_keys2
```

```
cat id_dsa.pub >> authorized_keys2
```

```
chmod 400 id_dsa
```

```
scp authorized_keys2 you@remote.machine.net:~/.ssh/
```


In conclusion...