

# Tips and Tricks: SSH Tunnels

Chris Charles (@ChrisUnfiltered)

Guelph Coffee and Code

March 24, 2011

# SSH tunnels



# SSH tunnels



**CLIENT**

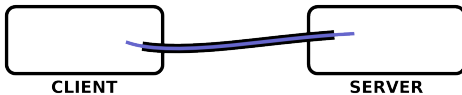


**SERVER**

# SSH tunnels



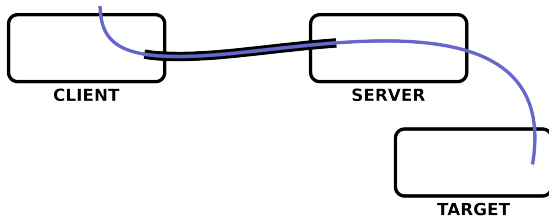
# SSH tunnels



# SSH tunnels



# SSH tunnels



# Use case 1: Add encryption to an insecure protocol

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

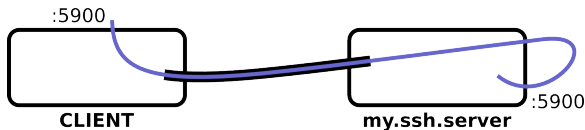


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

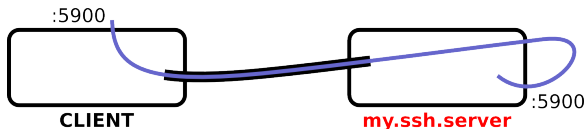


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

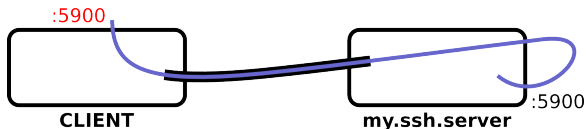


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

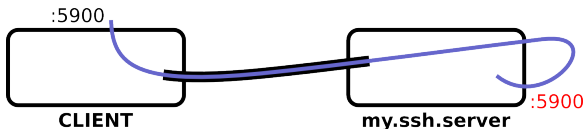


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

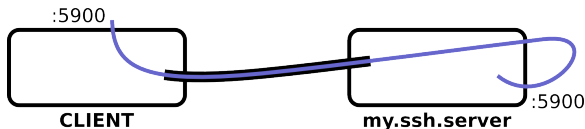


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

- Point your RDP program on the client machine at `localhost:5900`

# Use case 1: Add encryption to an insecure protocol

- **Situation:** You need to connect to `my.ssh.server` over an untrusted network using an insecure protocol like VNC.

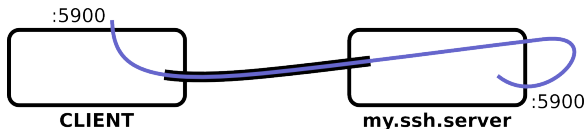


Figure: `ssh -L 5900:localhost:5900 my.ssh.server`

- Point your RDP program on the client machine at `localhost:5900`
- Use case 2: Also use this to access `localhost-only` administrative tools

# Use case 3: The poor person's VPN

## Use case 3: The poor person's VPN

- **Situation:** You need to access firewalled resources on `my.target.server` that are not open to the outside world. There is a machine inside the firewall, `my.ssh.server` to which you can connect via SSH.

## Use case 3: The poor person's VPN

- **Situation:** You need to access firewalled resources on `my.target.server` that are not open to the outside world. There is a machine inside the firewall, `my.ssh.server` to which you can connect via SSH.

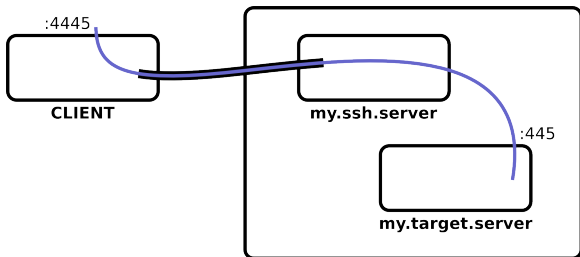


Figure: `ssh -L 4445:my.target.server:445 my.ssh.server`

## Use case 3: The poor person's VPN

- **Situation:** You need to access firewalled resources on `my.target.server` that are not open to the outside world. There is a machine inside the firewall, `my.ssh.server` to which you can connect via SSH.

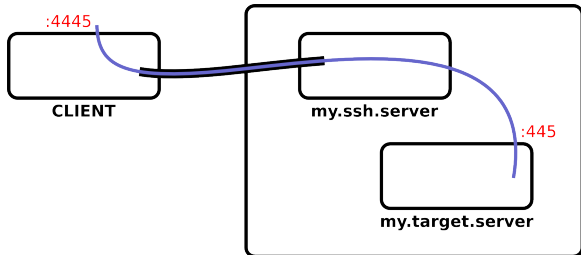


Figure: `ssh -L 4445:my.target.server:445 my.ssh.server`

## Use case 3: The poor person's VPN

- **Situation:** You need to access firewalled resources on `my.target.server` that are not open to the outside world. There is a machine inside the firewall, `my.ssh.server` to which you can connect via SSH.

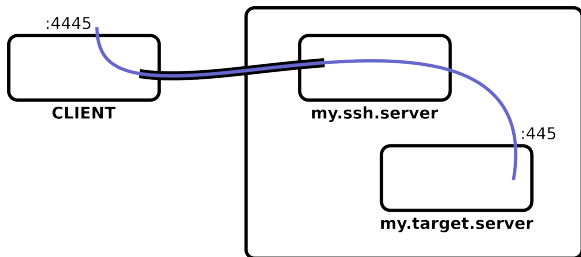


Figure: `ssh -L 4445:my.target.server:445 my.ssh.server`

- Point your CIFS client at `localhost:4445`

# Use case 4: Running a temporary server

## Use case 4: Running a temporary server

- **Situation:** You want to grant temporary access to a friend outside your firewall to a server running inside your firewall. There is a server that you have SSH access to outside your firewall.

## Use case 4: Running a temporary server

- **Situation:** You want to grant temporary access to a friend outside your firewall to a server running inside your firewall. There is a server that you have SSH access to outside your firewall.

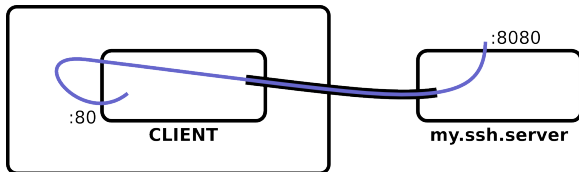


Figure: `ssh -R :8080:localhost:80 my.ssh.server`

## Use case 4: Running a temporary server

- **Situation:** You want to grant temporary access to a friend outside your firewall to a server running inside your firewall. There is a server that you have SSH access to outside your firewall.

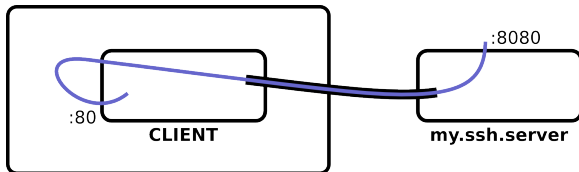


Figure: `ssh -R :8080:localhost:80 my.ssh.server`

## Use case 4: Running a temporary server

- **Situation:** You want to grant temporary access to a friend outside your firewall to a server running inside your firewall. There is a server that you have SSH access to outside your firewall.

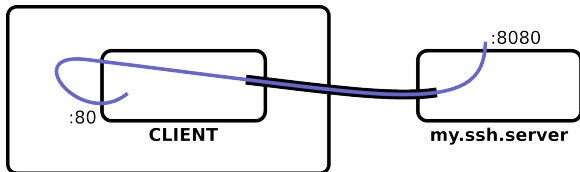


Figure: `ssh -R :8080:localhost:80 my.ssh.server`

- Ask your friend to point their browser at `my.ssh.server:8080`

## Use case 4: Running a temporary server

- **Situation:** You want to grant temporary access to a friend outside your firewall to a server running inside your firewall. There is a server that you have SSH access to outside your firewall.

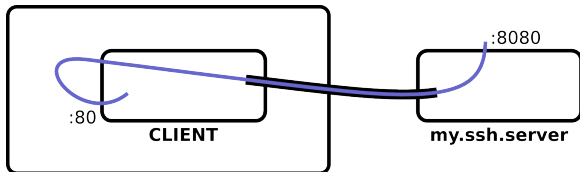


Figure: `ssh -R :8080:localhost:80 my.ssh.server`

- Ask your friend to point their browser at `my.ssh.server:8080`
- Note that this requires the `GatewayPorts` option to be enabled on `my.ssh.server`

# Use case 5: Dynamic web proxy

## Use case 5: Dynamic web proxy

- **Situation:** You want to route all of your web traffic through a server outside your firewall, `my.proxy.server`, that you can access via SSH.

# Use case 5: Dynamic web proxy

- **Situation:** You want to route all of your web traffic through a server outside your firewall, `my.proxy.server`, that you can access via SSH.

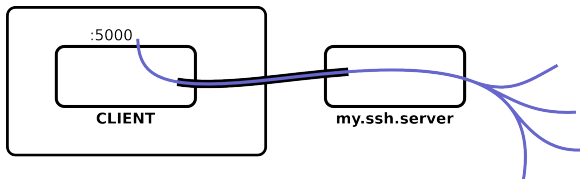


Figure: `ssh -D 5000 my.ssh.server`

## Use case 5: Dynamic web proxy

- **Situation:** You want to route all of your web traffic through a server outside your firewall, `my.proxy.server`, that you can access via SSH.

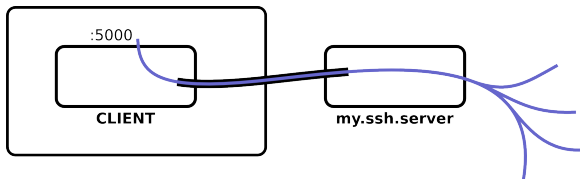


Figure: `ssh -D 5000 my.ssh.server`

- Set your web browser's SOCKS5 proxy to `localhost:5000`

## Use case 5: Dynamic web proxy

- **Situation:** You want to route all of your web traffic through a server outside your firewall, `my.proxy.server`, that you can access via SSH.

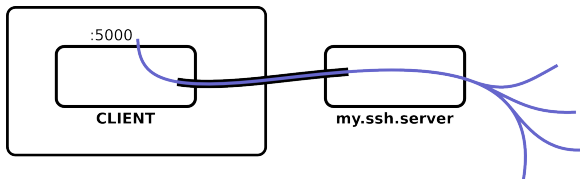


Figure: `ssh -D 5000 my.ssh.server`

- Set your web browser's SOCKS5 proxy to `localhost:5000`
- Use FoxyProxy or Proxy Switchy! to gain flexibility

## Documentation

- OpenSSH CLI suite
  - Available for Linux, Windows, OSX
  - Look at the -L (local), -R (remote) and -D (dynamic) options
- PuTTY
  - Usually on Windows, but also available for Linux
  - Relevant settings found in Connection → SSH → Tunnels