

# SSH 端口转发与 SOCKS 代理



K8sCat LV.5

2022年01月19日 23:37

## SSH port forwarding 端口转发的三种方式

### Local Port Forwarding 本地端口转发

将连接从客户端主机转发到 SSH 服务器主机，然后转发到目标主机端口。

```
ssh -L [LOCAL_IP:]LOCAL_PORT:DESTINATION:DESTINATION_PORT [USER@]SSH_SERVER
```

参数说明：

- [LOCAL\_IP:]LOCAL\_PORT - 本地 IP 和端口号，LOCAL\_IP 默认是 localhost。
- DESTINATION:DESTINATION\_PORT - 目标机器的 IP 地址和端口号。
- [USER@]SERVER\_IP - 远程 SSH 地址和登录用户。

案例：

使用本地地址 `127.0.0.1:3336` 连接远程的数据库 `db001.host:3306`，同时 `127.0.0.1:3337` 连接到 `db002.host:3306`：

```
ssh -L 3336:db001.host:3306 3337:db002.host:3306 user@pub001.host
```

检查 SSH Server 配置：AllowTcpForwarding=yes

### Remote Port Forwarding 远程端口转发

远程端口转发与本地端口转发相反。它允许您将远程（ssh 服务器）机器上的端口转发到本地（ssh 客户端）机器上的端口，然后再转发到目标机器上的端口。

远程端口转发主要用于向外部人员提供对内部服务的访问权限。

```
ssh -R [REMOTE:]REMOTE_PORT:DESTINATION:DESTINATION_PORT [USER@]SSH_SERVER
```

参数说明：

- [REMOTE:]REMOTE\_PORT - 远程服务器地址和端口号。REMOTE 默认会所有地址。
- DESTINATION:DESTINATION\_PORT - 目标机器的 IP 地址和端口号。
- [USER@]SERVER\_IP - 远程 SSH 地址和登录用户。

案例：

```
ssh -R 8080:127.0.0.1:3000 -N -f user@remote.host
```

上面的命令将使 `ssh` 服务器侦听端口 8080，并将所有流量从此端口传输到本地计算机的 3000 端口上。这样就可以在浏览器中输入 `the_ssh_server_ip:8080` 访问应用了。

检查 SSH Server 配置：GatewayPorts=yes

## Dynamic Port Forwarding 动态端口转发

动态端口转发允许您在本地（ssh 客户端）机器上创建一个套接字，它充当 SOCKS 代理服务器。当客户端连接到这个端口时，连接被转发到远程（ssh 服务器）机器，然后被转发到目标机器上的动态端口。

```
ssh -D [LOCAL_IP:]LOCAL_PORT [USER@]SSH_SERVER
```

参数说明：

- [LOCAL\_IP:]LOCAL\_PORT - 本地 IP 地址和端口号。LOCAL\_IP 默认是 localhost。
- [USER@]SERVER\_IP - 远程 SSH 地址和登录用户。

案例：

```
ssh -D 8080 -N -f -C -q user@remote.host
```

- -D 8080 启动一个 SOCKS 服务并监听本地的 9090 端口
- -f 后台运行
- -C 压缩请求数据
- -q 使用静默模式
- -N 不执行远程命令

## SOCKS 代理使用场景

---

### cURL 代理

```
curl -x socks5://127.0.0.1:8080 https://google.com
```

### 终端代理

```
export http_proxy=socks5://127.0.0.1:8080 \  
https_proxy=socks5://127.0.0.1:8080 \  
all_proxy=socks5://127.0.0.1:8080
```

### Git 代理

```
git config --global http.proxy socks5://127.0.0.1:8080  
git config --global https.proxy socks5://127.0.0.1:8080
```

### SSH 代理

编辑 `~/.ssh/config` :

```
Host github.com  
  HostName github.com  
  User git  
  IdentityFile ~/.ssh/id_rsa  
  ProxyCommand nc -v -x 127.0.0.1:8080 %h %p
```

config