

Tinc-VPN LAN party notes

Build tinc from git

Install packages (Ubuntu 20.04):

```
sudo apt install git build-essential autoconf texinfo \
    zlib1g-dev liblz2-dev libssl-dev libncurses-dev \
    libreadline-dev libminiupnpc-dev
```

Build and install:

```
git clone https://github.com/gsliepen/tinc.git
cd tinc
autoreconf -fsi
./configure --disable-legacy-protocol --enable-miniupnpc
make
sudo make install-strip
```

Set needed capabilities

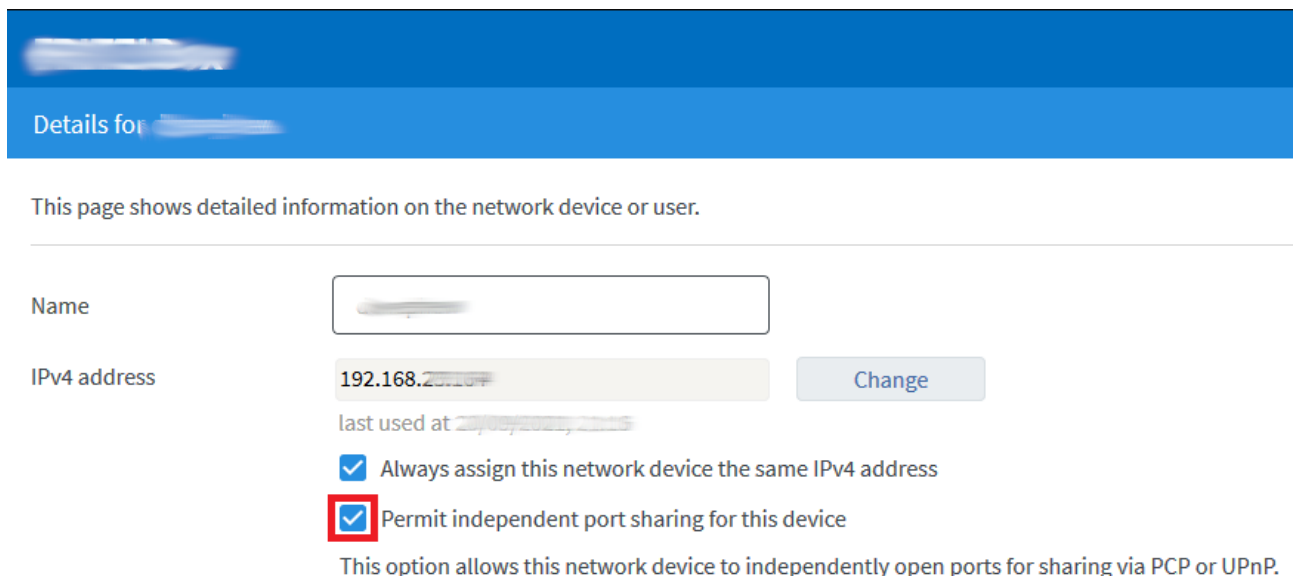
If you used port numbers bigger than 1024 then you don't need `CAP_NET_BIND_SERVICE`. `CAP_SYS_NICE` is good for lower latency but isn't needed if `ProcessPriority = high` is disabled from "tinc.conf".

```
sudo setcap cap_net_bind_service,cap_sys_nice+ep /usr/local/sbin/tincd
```

Firewall

If you want to host game sessions it is advised to allow tinc to forward its UDP/TCP port at your internet access. You'll have to use the same port number (forwarding from another port won't work). The more nodes reachable externally via tinc port the better.

The easiest way to do this is allow your PC to configure the firewall on your internet router via UPnP. For example:



Details for [redacted]

This page shows detailed information on the network device or user.

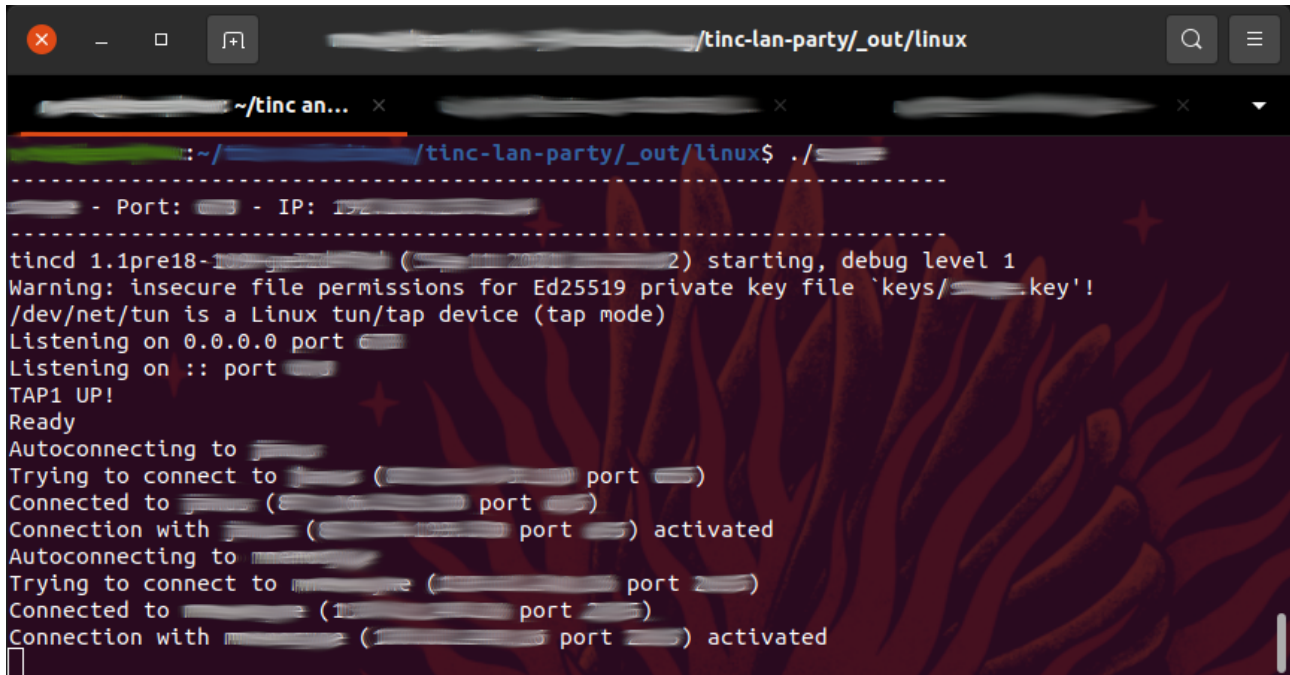
Name	[redacted]
IPv4 address	192.168.254.104 Change
	last used at [redacted]
	<input checked="" type="checkbox"/> Always assign this network device the same IPv4 address
	<input checked="" type="checkbox"/> Permit independent port sharing for this device

This option allows this network device to independently open ports for sharing via PCP or UPnP.

Running the node scripts

At the first run the **TAP1** device will be created for your user (it can be removed by calling `tincvpn/remove-TAP`). Your user needs to be able to execute commands as super user with `sudo` because otherwise the tap adapter can neither be created nor configured (I haven't found a way to simplify this).

run the script with `./nodename`:

A terminal window titled "/tinc-lan-party/_out/linux" showing the execution of a script. The prompt is "~/tinc an...". The script output includes: a separator line, "Port: " and "IP: " fields, another separator line, "tincd 1.1pre18- (2) starting, debug level 1", a warning about insecure file permissions for the Ed25519 private key, "/dev/net/tun is a Linux tun/tap device (tap mode)", "Listening on 0.0.0.0 port ", "Listening on :: port ", "TAP1 UP!", "Ready", "Autoconnecting to ", "Trying to connect to (port)", "Connected to (port)", "Connection with (port) activated", "Autoconnecting to ", "Trying to connect to (port)", "Connected to (port)", and "Connection with (port) activated".

```
~/tinc an... x /tinc-lan-party/_out/linux
~/tinc an... x /tinc-lan-party/_out/linux$ ./
-----
- Port:  - IP: 
-----
tincd 1.1pre18- (2) starting, debug level 1
Warning: insecure file permissions for Ed25519 private key file `keys/.key'!
/dev/net/tun is a Linux tun/tap device (tap mode)
Listening on 0.0.0.0 port 
Listening on :: port 
TAP1 UP!
Ready
Autoconnecting to 
Trying to connect to ( port )
Connected to ( port )
Connection with ( port ) activated
Autoconnecting to 
Trying to connect to ( port )
Connected to ( port )
Connection with ( port ) activated
```