**Step-by-Step Setup of Cisco Trex**

🔽 Step 1: Download Trex

cd ~

wget --no-check-certificate <https://trex-tgn.cisco.com/trex/release/v2.98.tar.gz>

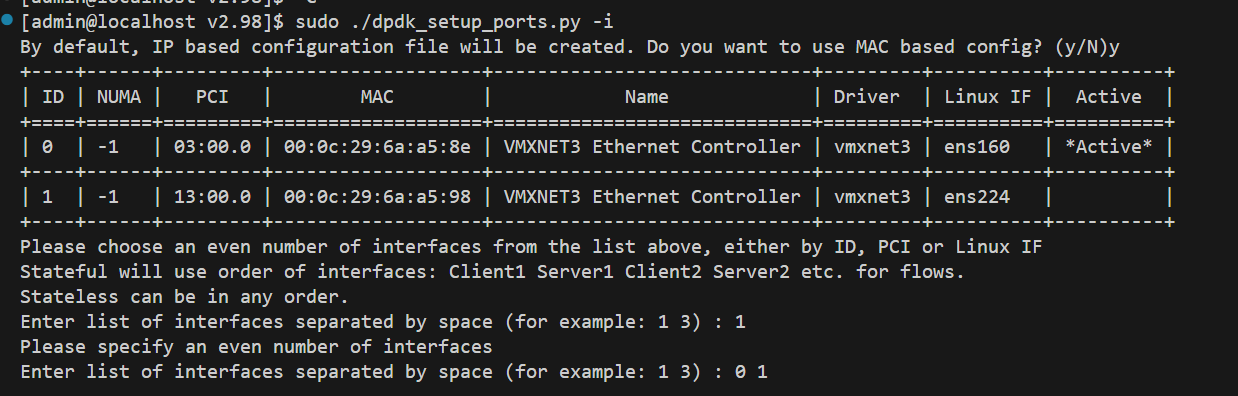
tar -xvzf v2.98.tar.gz

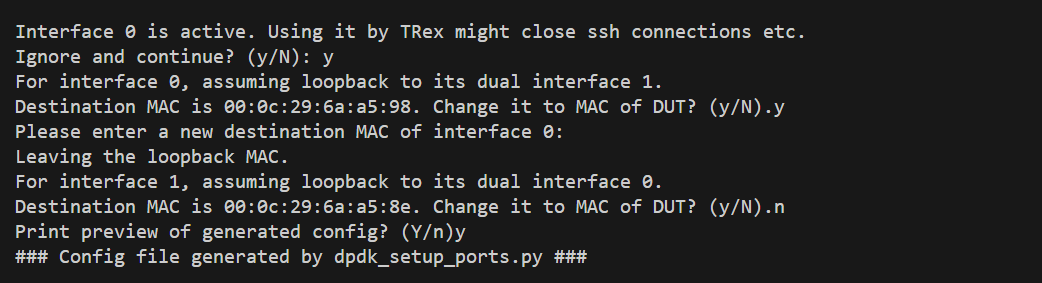
cd v2.98

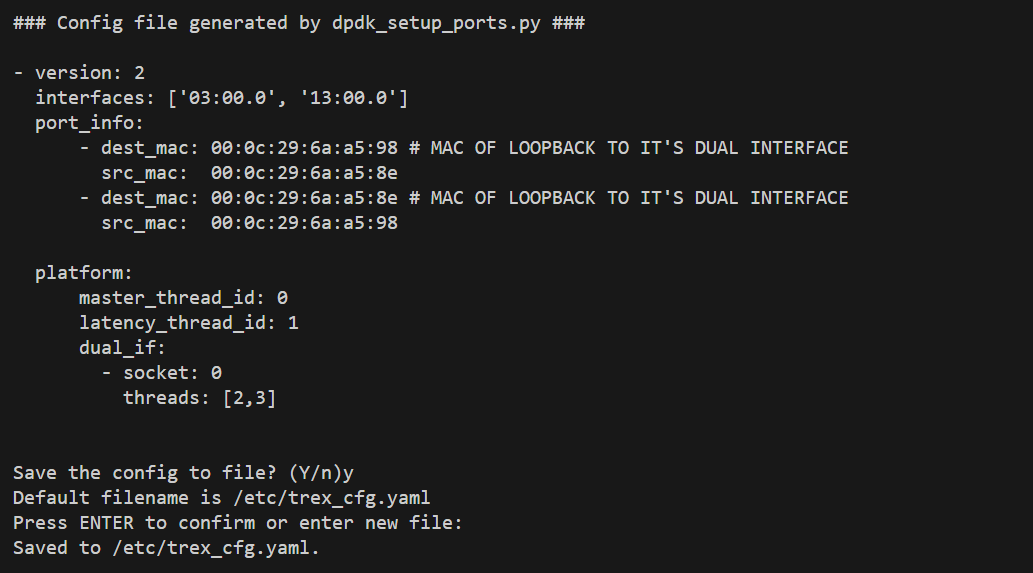
Note: -- Check for the latest release -- [Index of /trex/release](https://trex-tgn.cisco.com/trex/release/)

🧪 Step 2: Run Hardware Compatibility Check

sudo ./scripts/dpdk\_setup\_ports.py -i



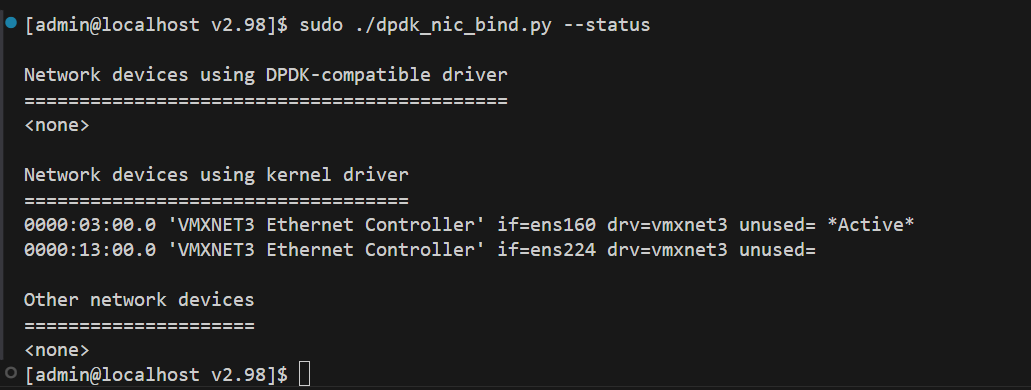




**⚙️ Step 3: Bind NICs to DPDK-compatible drivers**

You'll be prompted during step 2 to bind NICs to DPDK. Or run manually:

sudo ./dpdk\_nic\_bind.py –status



❌ **No NICs are currently bound to a DPDK-compatible driver**, which means TRex cannot use them until they're bound.

**✅ What Needs to Happen**

You need to **bind your interfaces (03:00.0, 13:00.0) to a DPDK-compatible driver** like igb\_uio, vfio-pci, or uio\_pci\_generic.

**🧰 Step-by-Step Fix: Bind NICs for TRex**

**🔧 Step 1: Load the DPDK Drivers (if not done yet)**

**sudo modprobe uio**

**sudo insmod ./dpdk/drivers/uio/igb\_uio.ko**

**( If igb\_uio.ko doesn’t exist in ./dpdk/drivers, use vfio-pci instead: )**

**sudo modprobe vfio-pci**

**🔧 Step 2: Bind the Interfaces**

sudo ./dpdk\_nic\_bind.py --bind=igb\_uio 0000:05:00.0 0000:05:00.1

(Replace PCI IDs with your NICs)

**🚀 Step 4: Start TRex in Stateless Mode**

bash

CopyEdit

sudo ./t-rex-64 -i

* This will launch the TRex interactive stateless CLI.
* You can send traffic, load profiles, and monitor metrics.

**🧪 Sample Traffic Test**

bash

CopyEdit

sudo ./t-rex-64 -f cap2/dns.yaml -m 1000 -d 10 -l 1000

* -f: YAML profile (cap2 folder has examples)
* -m: Packets per second
* -d: Duration in seconds
* -l: Log rate (every N packets)

**📡 GUI (Optional)**

TRex also provides a **web GUI** and **Scapy-like scripting mode**.

**🧹 To Reset (Unbind NICs):**

bash

CopyEdit

sudo ./dpdk\_nic\_bind.py --bind=e1000e 0000:05:00.0