big

Instruction Note:

Connect GNS3 Virtual Router with Local Machine

Requirements

- **GNS3 installed on your PC (https://www.gns3.com/)**
- **A virtual router image (Cisco IOS, CSR, or similar) loaded in GNS3**
- **GNS3 VM installed (recommended)**
- **Your PC network adapter (Wi-Fi or Ethernet)**

Step 1 — Add a Cloud in GNS3

1. **Open your GNS3 project.**
2. **On the left toolbar, drag Cloud into the workspace.**
3. **Double-click the Cloud → Configure → select your PC network adapter:**
    - **For Wi-Fi: Wi-Fi (or Intel(R) Wi-Fi)**
    - **For Ethernet: Ethernet**
4. **Click OK.**

This connects GNS3 to your host PC network.

big

Before doing this, we need to arrange the network adapter settings on your Windows PC (Windows 11).

Step 1: Open Device Manager

1. Press **Windows + R** → type devmgmt.msc → Enter.
2. In Device Manager, click **Action → Add legacy hardware**.

Step 2: Start the Add Hardware Wizard

1. Click **Next** on the wizard.
2. Select **Install the hardware that I manually select from a list (Advanced)** → Next.

Step 3: Select Network Adapter

1. Scroll down and select **Network adapters** → Next.
2. On the left: **Microsoft**
3. On the right: Select **KM-TEST Loopback Adapter** → Next.
4. Click **Next** to install it.

If you don't see it, make sure you chose **Microsoft** as manufacturer and **Network adapters** as device type.

Step 4: Finish Installation

1. The wizard installs the adapter.
2. Click **Finish**.

Step 5: Verify in Network Connections

1. Press **Windows + R** → ncpa.cpl → Enter.
2. You should see **Ethernet X – KM-TEST Loopback Adapter**.

big

Why we need a loopback adapter

A. Bridge virtual devices to your Windows PC

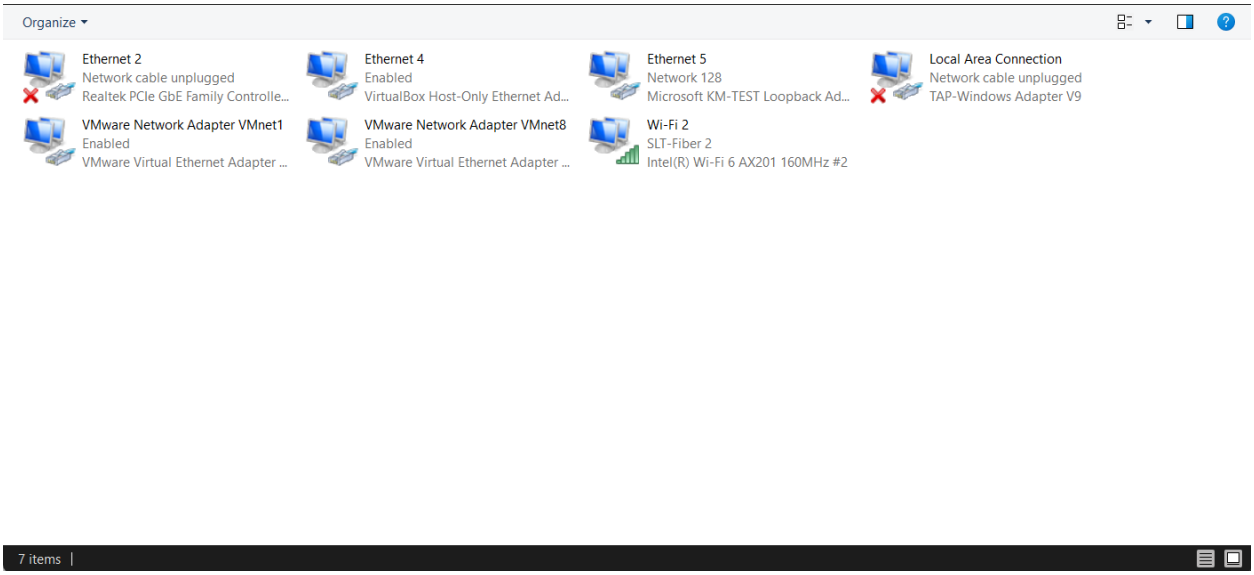
- *GNS3 routers live inside a virtual environment.*
- *If you want **your Windows PC to communicate with the virtual routers** (for SSH, Netmiko scripts, ping, etc.), there must be a **network interface on Windows** that GNS3 can connect to.*
- *The KM-TEST adapter acts like a **fake network card**, giving GNS3 a virtual "host network" to bridge to.*

B. No dependency on real network

- *If your Wi-Fi or Ethernet is busy, restricted, or on a different subnet, you can still run your lab.*
- *Loopback adapter ensures **connectivity between Windows host and GNS3 routers** without touching the real LAN.*

C. Static, predictable IPs

- *You can assign any IP to the loopback adapter (e.g., 192.168.100.1/24) for labs.*
- *This prevents **IP conflicts** with your home network.*
- *Useful for automation scripts that always connect to a fixed IP.*



big

After that you need to Assign an IP to the loopback adapter

Open Network Connections:

Press Windows + R → type ncpa.cpl → Enter.

Right-click the KM-TEST Loopback Adapter → Properties → select Internet Protocol Version 4 (TCP/IPv4) → Properties.

Assign a static IP (for example):

IP address: 192.168.1.10

Subnet mask: 255.255.255.0

Default gateway: 192.168.1.1

Now we need to adjust the GNS3 Virtual Machine settings.

Step 014 Open GNS3 Preferences

1. Open GNS3.
2. Go to **Edit → Preferences → GNS3 VM**.

Step 02 Enable the GNS3 VM

1. Make sure **Enable the GNS3 VM** is **checked**.
2. Set the **VM name** according to your virtualization software (VMware Workstation, VirtualBox).
    - Example: GNS3 VM
3. Select the correct **VM type** (VMware Workstation/Player or VirtualBox).

big

Step 03 Adjust Network Adapters for the VM

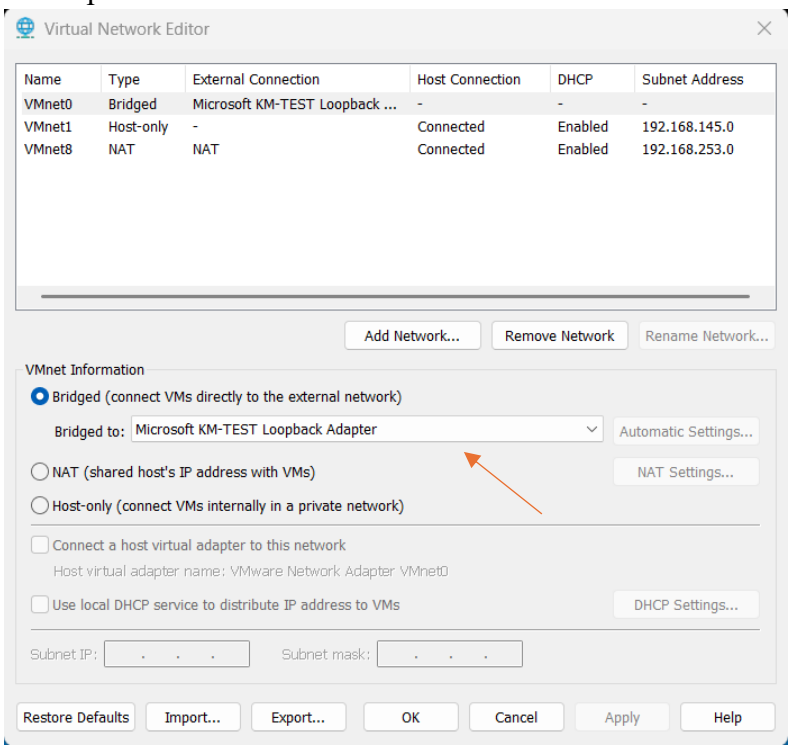
1. Open your **VMware Workstation/Player** (or VirtualBox).
2. Go to **GNS3 VM settings → Network Adapter**.
3. Set **Adapters**
    - Network adapter – host only
    - Network adapter 2 – NAT
    - network adapter 3 – Bridged (Automatic)

The next step is to configure the **Virtual Network Editor (VNE)** in VMware (if you're using VMware Workstation/Player) to make sure your **host (Windows)** and **GNS3 VM** can communicate properly.

Open Virtual Network Editor

1. Open **VMware Workstation/Player**.
2. Go to **Edit → Virtual Network Editor**.
3. Click **Change Settings** (you may need Administrator privileges).

Change the VMnet information on the GNS3 VM to use the Bridged connection mode, and bridge it to the Microsoft KM-TEST Loopback Adapter

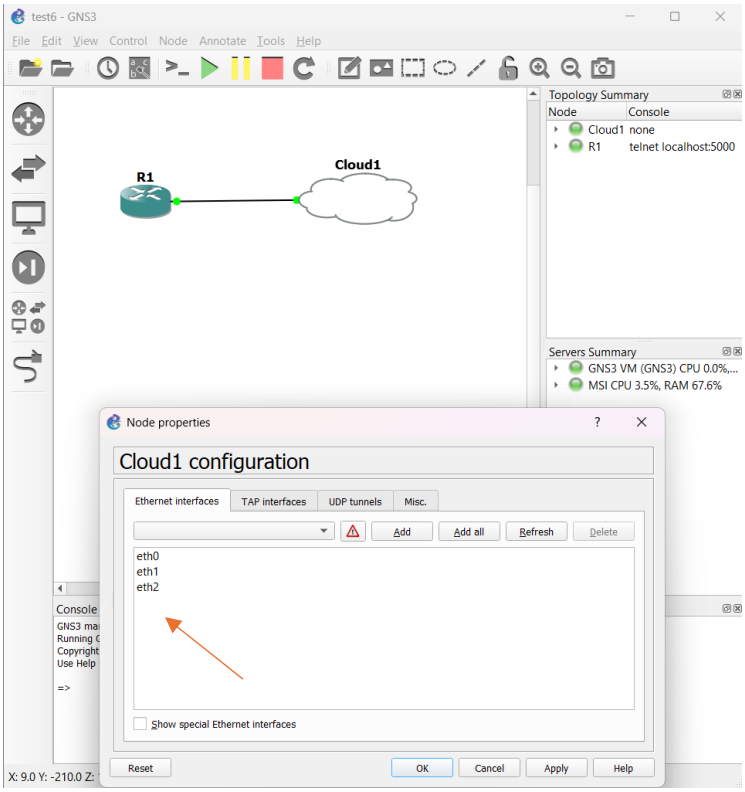


big

Now you can make topology

According to the GNS3 Virtual Machine settings, the third network adapter (Adapter 3) is typically the one configured for the Bridged connection mode.

The router's **FastEthernet 0/0** interface will then need to be connected to the **eth2** Ethernet interface on the Cloud component.



big

Now that your Windows PC, KM-TEST adapter, GNS3 VM, Network Topology and Virtual Network Editor are all set up, we can configure the routers in GNS3

Step 01 Add the router to GNS3

1. Drag a **Cisco IOSv or IOSvL2 router** into the workspace.
2. Connect the router to the **Cloud node** that is linked to your **KM-TEST adapter / VMnet1**.
3. Start the router.

Step 02 Enter router CLI

1. Right-click router → **Console**.
2. Enter enable mode:
3.

Step 03 Basic router settings

configure terminal

hostname R1 ! Set router name

no ip domain-lookup ! Disable DNS lookup to avoid delays

banner motd #Authorized users only!# ! Optional login banner

enable secret cisco ! Set enable password

Step 04 Configure interface

interface GigabitEthernet0/0

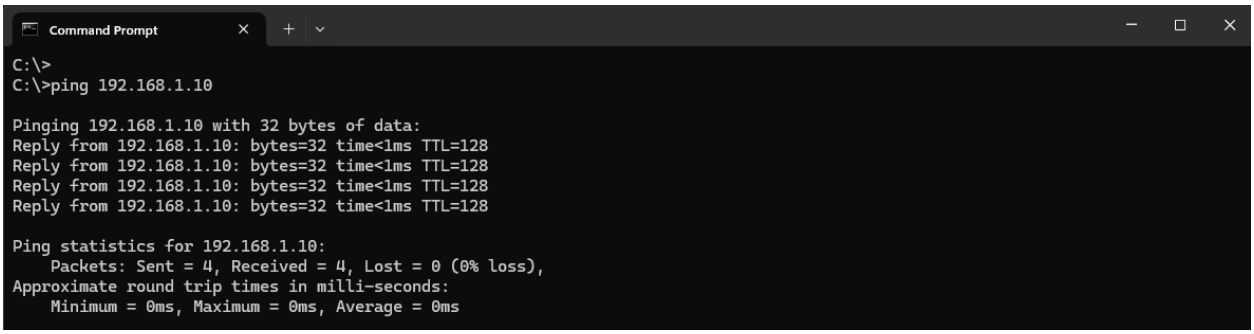
ip address 192.168.1.10 255.255.255.0 ! IP in same subnet as KM-TEST

no shutdown

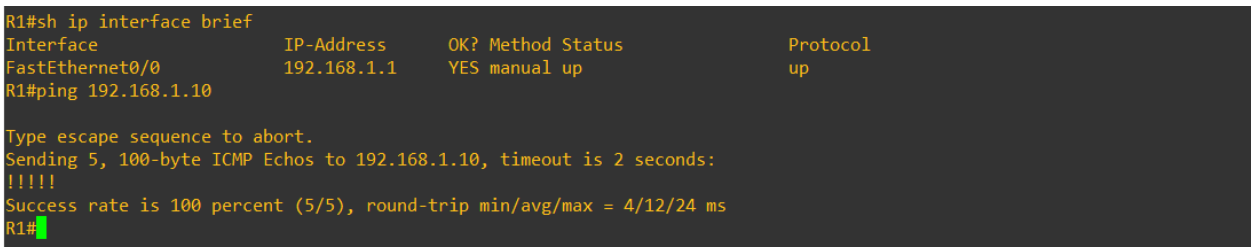
exit

big

Now, you can check the connectivity between the virtual router and your local machine. First, open the Command Prompt (CMD) on your PC and type the command: ping 192.168.1.10



After that, you can ping your PC's network adapter IP address from your virtual router. ping 192.168.1.1



Step 05 Configure SSH for automation scripts

ip domain-name local

username admin privilege 15 secret cisco

crypto key generate rsa modulus 1024

ip ssh version 2

line vty 0 4

login local

transport input ssh

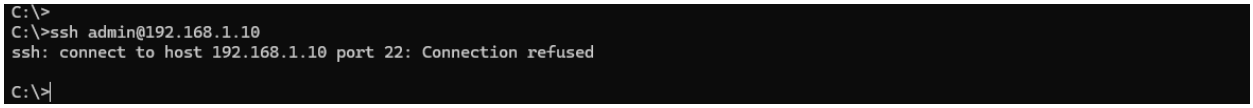
exit

big

Now you can try to connect to the router from your PC using SSH (Secure Shell).

Using- ssh admin@192.168.1.10

But If your router image is outdated The SSH negotiation problem ("no matching host key type found") occurs because you are likely using an older virtual router image that relies on outdated, insecure cryptographic algorithms like ssh-rsa or 3des



Modern Windows OpenSSH clients have tight security rules. They

reject older/legacy SSH algorithms by default.

Your Cisco IOSv router is using:

- Host key: ssh-rsa (old)
- Key exchange: diffie-hellman-group14-sha1 (old)
- Cipher: aes128-cbc (legacy CBC cipher)
- MAC: hmac-sha1 (legacy)

Every time OpenSSH sees something weak or old, it refuses the connection:

-cbc, which your modern PC's SSH client automatically rejects.

ssh -oHostKeyAlgorithms=+ssh-rsa -oKexAlgorithms=+diffie-hellman-group14-sha1 -c aes128-cbc -oMACs=hmac-sha1 admin@192.168.1.1