Contents

Cisco DevNet Sandbox Guide

A Comprehensive Resource for Network Lab Exercises

Prepared by: EQ6 **Subject:** Network Essentials - DevNet Integration **Purpose:** Complete guide to accessing and using Cisco DevNet sandboxes for hands-on labs

---

Table of Contents

---

1. What is Cisco DevNet?

Cisco DevNet is Cisco's developer program that provides:

- **Free access** to real Cisco networking equipment
- **Cloud-based labs** (sandboxes) with live routers, switches, and other devices
- **APIs and SDKs** for network automation

- **Learning resources** including tutorials, documentation, and code samples
- **Community support** through forums and developer communities

Why Use DevNet for Labs?

Benefit	Description
Real Equipment	Work with actual Cisco IOS, not just simulators
No Cost	Completely free - no hidden charges
Pre-configured	Devices are already set up and networked
Safe Environment	Break things without consequences
Remote Access	Access from anywhere with internet
Enterprise Experience	Same equipment used in production networks

Official Website: https://developer.cisco.com

---

2. Types of Sandboxes

DevNet offers different sandbox types based on access and availability:

Always-On Sandboxes

- **Availability:** 24/7 instant access
- **Reservation:** Not required
- **Duration:** Unlimited
- **Shared:** Multiple users may be on the same sandbox
- **Best for:** Quick tests, learning basic commands
- **Limitation:** May be reset or changed by other users

Reservation-Based Sandboxes

- **Availability:** Must reserve in advance
- **Reservation:** Required (typically 1-4 hours to provision)
- **Duration:** Usually 1-7 days
- **Dedicated:** Private environment just for you
- **Best for:** Serious labs, course exercises, testing
- **Limitation:** Limited time, must plan ahead

Private Sandboxes (Advanced)

- **Availability:** Custom arrangements
- **Access:** Requires special permission or partnership
- **Duration:** Negotiable
- **Best for:** Development projects, research

---

3. Getting Started

Step 1: Create a Free DevNet Account

1. Navigate to https://developer.cisco.com
2. Click **"Log In"** or **"Register"** in the top-right corner
3. Fill out the registration form:
    - Email address
    - Password
    - First/Last name
    - Country
4. Verify your email address
5. Complete your profile (optional but recommended)

Note: You do NOT need a Cisco CCO account, but if you have one, you can use it to log in.

Step 2: Navigate to Sandbox Catalog

1. Log in to DevNet
2. Hover over **"Sandbox"** in the top menu
3. Click **"Browse Catalog"**
4. Or go directly to: https://devnetsandbox.cisco.com/

Step 3: Browse Available Sandboxes

You'll see categories: - **Networking** - Routers, switches, SD-WAN, wireless - **Collaboration** - Webex, voice, video - **Data Center** - UCS, ACI, storage - **Security** - Firewalls, Stealthwatch, ISE - **IoT** - Edge computing, industrial networking

---

4. Accessing a Sandbox

For Always-On Sandboxes:

1. Find an always-on sandbox (marked with green "Always On" badge)
2. Click **"View Details"**
3. Scroll to **"Access Information"** section
4. Note the connection details:
    - VPN credentials (if required)
    - Device IP addresses
    - SSH/HTTPS access URLs
    - Usernames and passwords
5. Connect immediately - no reservation needed

For Reservation-Based Sandboxes:

1. Find the sandbox you want (marked with clock icon)
2. Click **"Reserve"**
3. Select:
    - **Start Date/Time** (when sandbox will become available)

   - **Duration** (how long you need it - 1 hour to 7 days)
4. Provide **reason for use** (optional but helpful)
5. Click **"Reserve"**
6. Wait for provisioning (you'll receive email notifications):
   - **Provisioning Started** - Setup has begun
   - **Sandbox Ready** - Your sandbox is ready to use
7. Check your email for:
   - VPN connection file (.ovpn)
   - Access credentials
   - Device IP addresses

Provisioning Time: Typically 15 minutes to 2 hours depending on complexity.

---

5. Recommended Sandboxes for Network Labs

For Routing & Switching Labs:

IOS XE on CSR - Always On

- **Type:** Always-On
- **Devices:** 1 Cisco CSR 1000v router
- **Best for:** Basic routing, CLI practice
- **Access:** SSH and NETCONF/RESTCONF
- **URL:** Search "IOS XE on CSR" in sandbox catalog

Multi-IOS Cisco Test Network

- **Type:** Reservation
- **Devices:** Multiple routers and switches
- **Best for:** Complex routing labs, OSPF, EIGRP
- **Access:** Console, SSH, Telnet
- **Duration:** Up to 7 days

Cisco Modeling Labs (CML) - Previously VIRL

- **Type:** Reservation
- **Devices:** Build your own topology
- **Best for:** Custom network designs, full labs
- **Access:** Web GUI + CLI
- **Duration:** Up to 5 days
- **Note:** Most flexible option - create any topology

For Security & Access Control:

Cisco ISE Sandbox

- **Type:** Always-On or Reservation
- **Best for:** Network access control, 802.1X
- **Access:** Web GUI + CLI

For SD-WAN:

SD-WAN Sandbox

- **Type:** Reservation
- **Best for:** Modern WAN architecture
- **Access:** vManage GUI + device CLI

---

6. Connection Methods

Method 1: Direct SSH Access (No VPN Required)

Some always-on sandboxes allow direct SSH to public IPs.

Example:

```
ssh developer@sandbox-iosxe-latest-1.cisco.com
Password: C1sco12345
```

From Windows PowerShell/CMD:

```
ssh developer@sandbox-iosxe-latest-1.cisco.com
```

From Linux/Mac Terminal:

```
ssh developer@sandbox-iosxe-latest-1.cisco.com
```

Method 2: VPN Connection (Most Common)

Most sandboxes require VPN for security.

Windows VPN Setup:

1. **Install OpenVPN Client:**
   - Download from: https://openvpn.net/community-downloads/
   - Install "OpenVPN Connect" or "OpenVPN GUI"
2. **Import VPN Configuration:**
   - Download `.ovpn` file from sandbox email
   - Open OpenVPN Connect
   - Click **"Import"** → **"File"**
   - Select the downloaded `.ovpn` file
   - Click **"Connect"**
3. **Verify Connection:**
   - You should see "Connected" status
   - You're now on the sandbox network

Mac VPN Setup:

1. **Install Tunnelblick:**
   - Download from: https://tunnelblick.net/
   - Install and open
2. **Import Configuration:**

- Double-click the `.ovpn` file
- Follow prompts to add configuration
- Click **“Connect”**

Linux VPN Setup:

1. **Install OpenVPN:**

```
sudo apt update
sudo apt install openvpn
```

2. **Connect:**

```
sudo openvpn --config /path/to/sandbox.ovpn
```

Leave the terminal open - VPN stays connected while running.

Method 3: Web-Based SSH (Browser Access)

Some sandboxes provide web-based terminals:

1. Click **“Access”** link from sandbox details page
2. Opens web browser with embedded terminal
3. No VPN or SSH client needed
4. May have limited features compared to real SSH

Method 4: Console Access

For devices that support console connections:

1. Connect via VPN
2. Use telnet to console server
3. Access device console directly

Example:

```
telnet 10.10.20.50 17001
```

Where `17001` is the console port number for specific device.

---

7. Working with IOS CLI

Initial Connection

Once connected to a device via SSH:

```
Router>
```

You’re in **User EXEC mode** (limited commands).

Enter Privileged EXEC Mode

```
Router> enable
Password: [enter enable password from sandbox docs]
Router#
```

Now you're in **Privileged EXEC mode** (more commands available).

Common Commands for Lab Setup

```
Router# show running-config
```

View Current Configuration:

```
Router# show ip interface brief
```

View IP Interfaces:

```
Router# show ip route
```

View Routing Table:

```
Router# configure terminal
Router(config)#
```

Enter Configuration Mode:

```
Router(config)# interface GigabitEthernet1
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)#
```

Configure an Interface:

```
Router(config)# ip route 192.168.2.0 255.255.255.0 10.0.0.2
```

Add a Static Route:

```
Router(config)# end
Router# copy running-config startup-config
```

Save Configuration: Or shorter version:

```
Router# write memory
```

```

Router(config)# end
Router#
```

Exit Configuration Mode:

Adapting Lab Exercises to Sandbox

When working with sandbox environments, you need to adapt lab instructions:

1. **Check existing configuration first:**

   ```
   show running-config
   show ip interface brief
   ```

2. **Identify available interfaces:**

   - Sandbox may have different interface names
   - Use `GigabitEthernet1`, `GigabitEthernet2`, etc.
   - NOT the same as Packet Tracer (GigabitEthernet0/0/0)

3. **Work with existing IPs:**

   - Don't change management IPs (you'll lose connection)
   - Use additional interfaces for labs
   - Check which interface is for management

4. **Use available networks:**

   - Sandbox may have pre-configured networks
   - Add your lab networks alongside existing ones

Example Adaptation:

Lab says:

```
interface GigabitEthernet0/0/0
ip address 192.168.1.1 255.255.255.0
```

Sandbox reality:

```
show ip interface brief  # See what's available
interface GigabitEthernet2  # Use available interface
ip address 192.168.1.1 255.255.255.0
```

---

8. Troubleshooting Common Issues

Issue 1: Can't Connect to VPN

Symptoms: OpenVPN shows "Connection failed" or timeout

Solutions: - Check firewall - allow OpenVPN on port 1194 (UDP) - Try different network (corporate networks may block VPN) - Use mobile hotspot as temporary solution - Verify `.ovpn` file is correct and not corrupted - Re-download VPN config from sandbox page

Issue 2: VPN Connected but Can't SSH to Devices

Symptoms: VPN shows connected, but `ssh` times out

Solutions: - Verify you're using correct IP addresses from sandbox docs - Check if sandbox is still active (hasn't expired) - Ping the device first: `ping 10.10.20.48` - Try telnet to test connectivity: `telnet 10.10.20.48 22` - Make sure you're using credentials from sandbox email

Issue 3: Wrong Credentials / Access Denied

Symptoms: "Access denied" or "Authentication failed"

Solutions: - Double-check username and password from sandbox docs - Credentials are case-sensitive - Common defaults: - Username: `developer`, `admin`, or `cisco` - Password: `C1sco12345`, `cisco123`, or listed in sandbox docs - Try enable password: often same as login password

Issue 4: Sandbox Expired or Changed

Symptoms: Nothing works, different IP addresses

Solutions: - Check reservation end time - Always-on sandboxes may have been reset by another user - For reservation sandboxes: extend reservation if time remains - Reserve a new sandbox if expired

Issue 5: Can't Save Configuration

Symptoms: Configuration lost after disconnect

Solutions: - Always run `copy running-config startup-config` - Check if you have write privileges - Some sandboxes reset on timer regardless - Document your configuration externally

Issue 6: Interface Won't Come Up

Symptoms: Interface stays down after `no shutdown`

Solutions: - Check if interface is connected to anything - Sandbox may have limited interfaces available - Try `show interfaces GigabitEthernet1` for details - Some virtual interfaces need additional config

---

9. Best Practices

Before Starting Your Lab:

- ☐ **Read sandbox documentation** - Every sandbox has specific details
- ☐ **Check reservation time** - Know when it expires
- ☐ **Test connectivity** - Verify you can connect before deep work
- ☐ **Document device IPs** - Save them locally for reference

- □ **Understand existing config** - Run `show run` first
- □ **Identify management interfaces** - Don't modify these

During Your Lab:

- □ **Save frequently** - Use `copy run start` after changes
- □ **Keep notes** - Document what you configured
- □ **Use multiple terminals** - Connect to multiple devices simultaneously
- □ **Test incrementally** - Verify each step before moving on
- □ **Keep VPN connected** - Don't disconnect until finished

Configuration Management:

Copy configs to local machine:

```
Router# show running-config
```

Copy the output to a text file on your computer.

Or use SCP (if available):

```
scp developer@10.10.20.48:running-config ./backup-config.txt
```

Time Management:

- **Reservation sandboxes have hard limits** - Work efficiently
- **Set reminders** - 1 hour before expiration
- **Extend early** - Don't wait until last minute
- **Complete critical steps first** - Nice-to-haves come later

Collaboration:

- **Share sandbox access** with lab partners (if allowed)
- **Don't interfere** with always-on sandbox users
- **Use private sandboxes** for group projects
- **Document for team** - Share findings and configs

---

10. Beyond Basic Labs

Network Automation

DevNet sandboxes support automation tools:

Python with Netmiko:

```
from netmiko import ConnectHandler

device = {
    'device_type': 'cisco_ios',
    'host': '10.10.20.48',
    'username': 'developer',
    'password': 'C1sco12345'
}

connection = ConnectHandler(**device)
output = connection.send_command('show ip interface brief')
print(output)
```

Ansible Playbooks:

```
- name: Configure router
  hosts: sandbox_router
  tasks:
    - name: Set interface IP
      ios_config:
        lines:
          - ip address 192.168.1.1 255.255.255.0
        parents: interface GigabitEthernet2
```

APIs and Programmability

Many sandboxes support: - **NETCONF** - XML-based configuration protocol - **RESTCONF** - REST API for network devices - **gRPC** - Modern RPC framework - **Model-driven telemetry** - Streaming network data

Example RESTCONF call:

```
curl -k -u developer:C1sco12345 \
  https://10.10.20.48/restconf/data/ietf-interfaces:interfaces
```

Learning Resources

DevNet Learning Labs: - https://developer.cisco.com/learning/ - Free guided tutorials - Integrated with sandboxes - Topics: Routing, switching, automation, APIs

DevNet GitHub: - https://github.com/CiscoDevNet - Sample code and scripts - Automation examples - Lab guides

DevNet Support: - **Community Forums:** https://community.cisco.com/t5/devnet/ct-p/devnet - **Webex Teams:** DevNet support space - **Stack Overflow:** Tag questions with `cisco-devnet`

Quick Reference Card

Essential URLs

Resource	URL
DevNet Home	https://developer.cisco.com
Sandbox Catalog	https://devnetsandbox.cisco.com

Resource	URL
Learning Labs	https://developer.cisco.com/learning/
Code Samples	https://github.com/CiscoDevNet
API Docs	https://developer.cisco.com/docs/
Community Forums	https://community.cisco.com/t5/devnet/ct-p/devnet

Common Default Credentials

Username	Password	Notes
`developer`	`C1sco12345`	Most common
`admin`	`C1sco12345`	Alternative
`cisco`	`cisco`	Legacy devices
`root`	`D_Vay!_10&`	Linux servers

Always check sandbox documentation - Credentials vary by sandbox.

Essential IOS Commands

Task	Command
Enable privileged mode	`enable`
Enter config mode	`configure terminal`
Show interfaces	`show ip interface brief`
Show routes	`show ip route`
Show config	`show running-config`
Save config	`copy running-config startup-config`
Add static route	`ip route [network] [mask] [next-hop]`
Configure interface	`interface [name]`
Exit config mode	`end` or `exit`

Lab Exercise Integration Checklist

When completing the **Static IP Routes Lab** using DevNet:

- □ Create DevNet account
- □ Find and reserve appropriate sandbox (IOS XE or Multi-IOS)
- □ Wait for provisioning email
- □ Install VPN client
- □ Connect to VPN using `.ovpn` file
- □ SSH to first router

- □ Run `show ip interface brief` to identify available interfaces
- □ Adapt lab IP addresses to available interfaces
- □ Configure first router interfaces
- □ SSH to second router (or reserve multi-device sandbox)
- □ Configure second router interfaces
- □ Configure static routes on both routers
- □ Test connectivity with `ping`
- □ Verify routes with `show ip route`
- □ Test path with `traceroute`
- □ Save all configurations
- □ Document results
- □ Disconnect VPN when complete

Appendix: Example Sandbox Session

Complete Workflow Example

1. Reserve Sandbox: - Logged into DevNet - Found "Multi-IOS Cisco Test Network" sandbox - Reserved for 4 hours starting immediately - Received email after 30 minutes with access details

2. Connect to VPN:

```
# Downloaded sandbox_vpn.ovpn from email
sudo openvpn --config sandbox_vpn.ovpn
# Left terminal open - VPN connected
```

3. SSH to Router-1:

```
ssh developer@10.10.20.48
Password: C1sco12345

Router1> enable
Password: C1sco12345
Router1#
```

4. Check Existing Configuration:

```
Router1# show ip interface brief

Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       10.10.20.48     YES DHCP   up                    up
GigabitEthernet2       unassigned      YES unset  administratively down down
GigabitEthernet3       unassigned      YES unset  administratively down down
```

5. Configure Lab Interface:

```
Router1# configure terminal
Router1(config)# interface GigabitEthernet2
Router1(config-if)# ip address 192.168.1.1 255.255.255.0
Router1(config-if)# no shutdown
```

```
Router1(config-if)# end
Router1# copy running-config startup-config
```

6. Repeat for Router-2 and complete lab…

Document Version: 1.0 **Last Updated:** 2025-12-01 **Prepared by:** EQ6 **For Questions:** Contact your instructor or visit DevNet community forums