# Introduction

The **Switch port configuration** module provides you with the instructions and Cisco hardware to develop your hands on skills in configuring Cisco switchports. This module includes the following exercises:

1) Basic port settings

2) Interface Macro commands

Lab Diagram

During your session you will have access to the following lab configuration. Depending on the exercises you may or may not use all of the devices, but they are shown here in the layout to get an overall understanding of the topology of the lab.
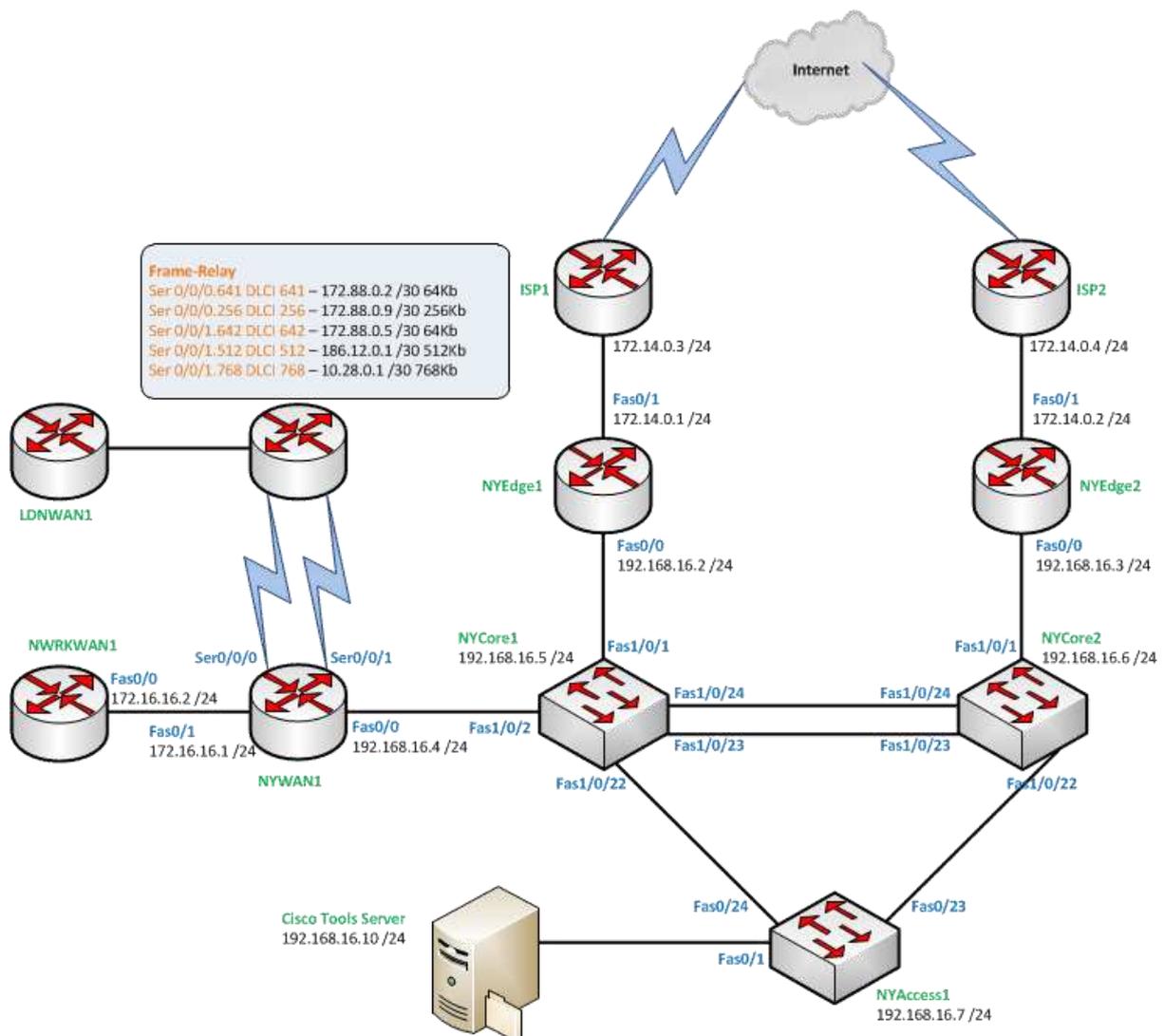
Connecting to your lab

In this module you will be working on the following equipment to carry out the steps defined in each exercise.

- NYCORE1
- NYCORE2
- NYACCESS1
- PLABCSCO01

Each exercise will detail which terminal you are required to work on to carry out the steps.

During the boot up process an activity indicator will be displayed in the device name tab:

- Black - Powered Off
- Orange - Working on your request
- Green - Ready to access

If the remote terminal is not displayed automatically in the main window (or popup) click the **Connect** icon located in the tools bar to start your session.

Copyright Notice

Exercise 1 – Basic port settings

In this exercise you will learn how to configure the basic port settings of a switch interface. You will also learn some troubleshooting techniques to find problems and rectify them using show commands. Please refer to your course material or use your preferred search engine to gain an understanding of these tasks.

Lab Diagram

This diagram focuses more on the devices in used in this module.
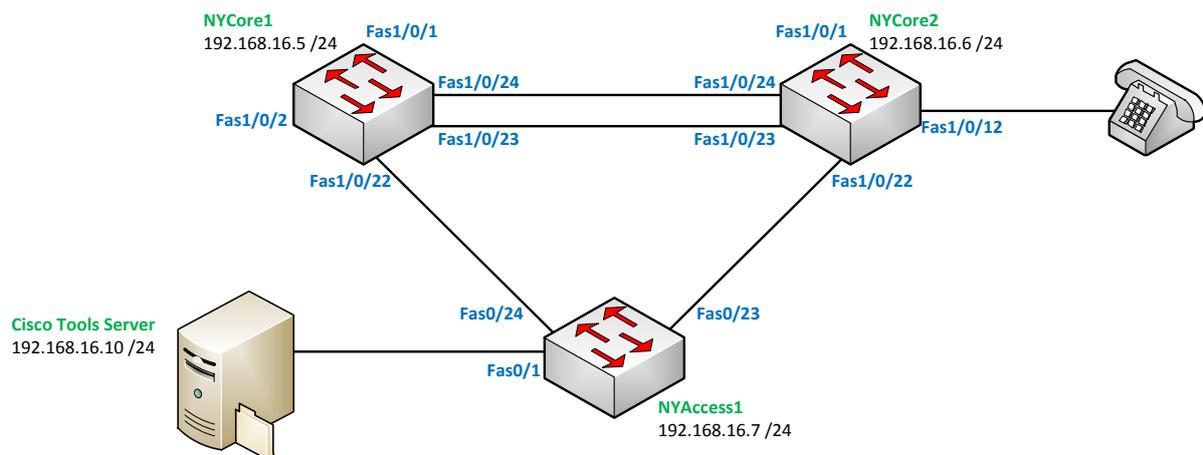
Configuring basic settings

Configuring switch interfaces is a daily task for most network engineers, whether it is new requirements from your business or general house-keeping and maintenance.

The commands used in this exercise will become second nature once you have used them a number of times.

In this exercise we are going to manually configure some switch port settings, see what happens when things are incorrectly configured and use some commands to help troubleshoot issues.

Ensure you have powered on PLABCSCO01 (this will need to manually be done) and the switches in the lab.

Configuring IP address

The first thing we need to do is configure some IP addresses on our Switches so that we can do some connectivity testing. If you look at the diagram above, you will see that the Windows Server has an IP address of **192.168.16.10**, we are going to assign our switches an IP address in the same range and use the server to test.

Step 1

On NYCORE1, assign an IP address on VLAN 1 by using the following commands (it's not recommended to use VLAN1 for management, but as this is a lab and to keep things simple, we'll use VLAN 1):

```
NYCORE1#configure terminal

NYCORE1(config)#interface VLAN 1

NYCORE1(config-if)#ip address 192.168.16.5 255.255.255.0

no shutdown
```

Confirm your configuration by using the show command:

```
show ip interface brief
```

The output from this command on NYCORE1:

```
NYCORE1#show ip interface brief

Interface            IP-Address      OK? Method Status            Protocol

VLAN1                192.168.16.5    YES manual up                up

FastEthernet1/0/1    unassigned      YES unset  down              down

FastEthernet1/0/2    unassigned      YES unset  up                up

FastEthernet1/0/3    unassigned      YES unset  down              down

FastEthernet1/0/4    unassigned      YES unset  down              down

FastEthernet1/0/5    unassigned      YES unset  down              down
```

(some output has been omitted)

Step 2

Configure the other switches as follows:

NYCORE2

```
VLAN1: 192.168.16.6 /24
```

NYACCESS1

```
VLAN1: 192.168.16.7 /24
```

Ensure all the interfaces are up using the show command above.

Step 3

Make sure you can ping each switch using the ping command. Test each switch in turn and ensure all 3 devices are up, for example from **NYACCESS1** you will need to ping 192.168.16.5 and 192.168.16.6:

```
NYACCESS1#ping 192.168.16.5

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.16.5, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/203/1000 ms

NYACCESS1#

NYACCESS1#ping 192.168.16.6

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.16.6, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/204/1004 ms

NYACCESS1#
```

Step 4

You will notice in the diagram that there multiple links between NYCORE1 and NYCORE2, and a single link between NYCORE1 and NYACCESS1 and also NYCORE2 and NYACCESS1.

These will have automatically come up as the default configuration settings will enable this to happen.

In this step we'll configure the interface descriptions and set the port speed and duplex settings on **NYCORE2**. These days auto-negotiate works very well with both workstations and servers and other networking devices so you may not need to set all these settings in the real world, but it's important to understand how to configure a port to work at 100mbps and in full duplex for example.

Configure interfaces 1/0/22 to be 100Mbps full duplex and give it a description using the commands (**Note:** You should configure both ends of the link when setting speed and duplex settings – this is very important in the real world when manually configuring these settings):

```
NYCORE2#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE2(config)#interface fastEthernet 1/0/22

NYCORE2(config-if)#description CON-NYACCESS1-Fas0-23

NYCORE2(config-if)#speed 100

NYCORE2(config-if)#duplex full
```

Step 5

Ensure you can communicate from the PLABCSCO01 Server by using **ping** in a command prompt:

- ping 192.168.16.5
- ping 192.168.16.6
- ping 192.168.16.7
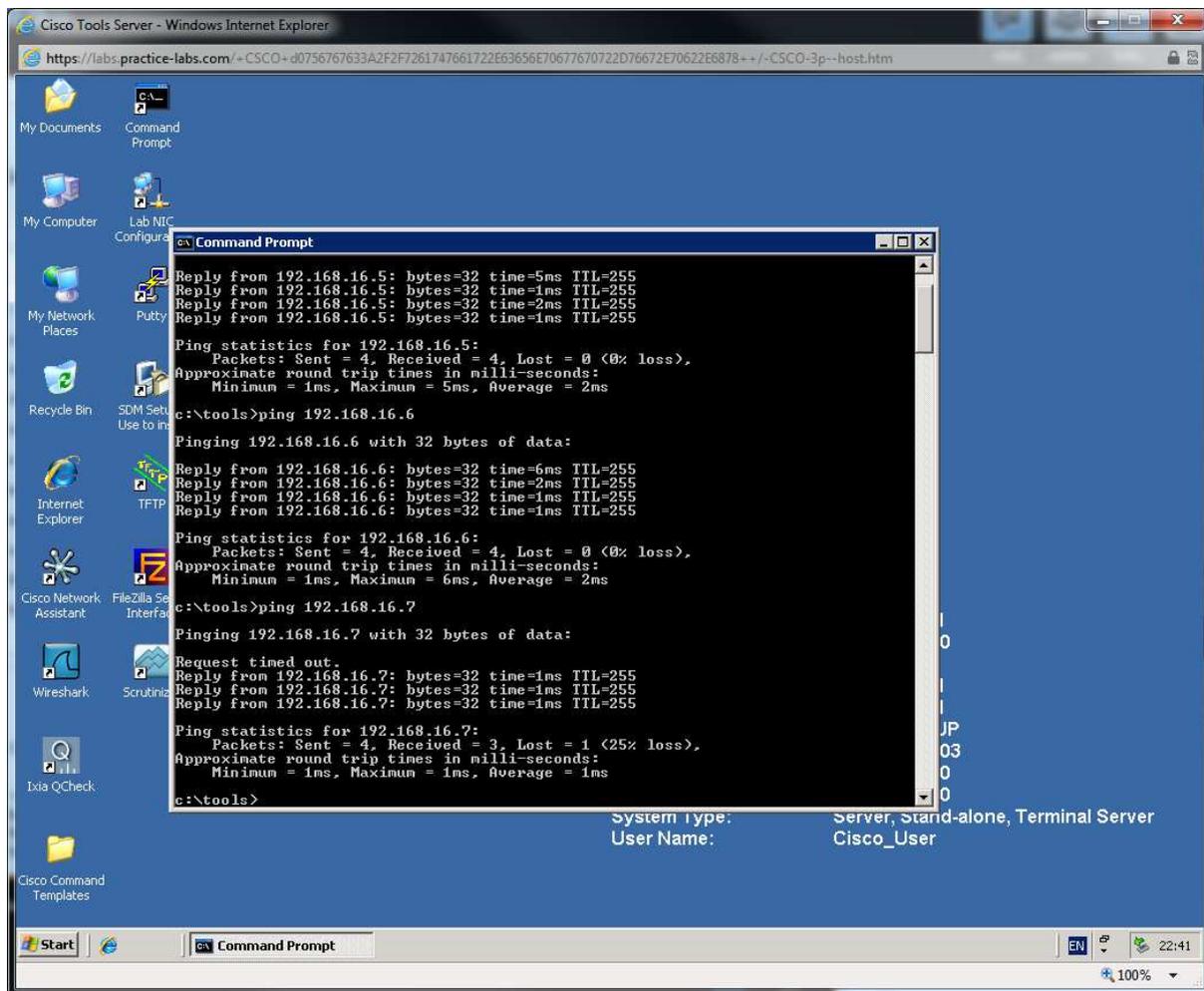
Using the Range Command

NYCORE1 has 2 interfaces that we want to configure with the same settings therefore this is a good candidate to use the **interface range** command.

Configure interfaces fastethernet 1/0/23 and 1/0/24 to be 100Mbps full duplex with a description:

```
NYCORE1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE1(config)#interface range fastEthernet 1/0/23-24

NYCORE1(config-if-range)#description CON-NYCORE2

NYCORE1(config-if-range)#speed 100

NYCORE1(config-if-range)#duplex full

NYCORE1(config-if-range)#

*Mar  1 00:09:40.307: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet1/0/23, changed state to down
```

```
*Mar  1 00:09:40.332: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet1/0/24, changed state to down

*Mar  1 00:09:41.322: %LINK-3-UPDOWN: Interface FastEthernet1/0/23, changed state
to down
```

Use the following command to confirm your settings:

```
show interfaces  status
```

```
NYCORE1#show interfaces status
Port        Name            Status        VLAN      Duplex  Speed Type
Fa1/0/1                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/2                     connected     1         a-full  a-100 10/100BaseTX
Fa1/0/3                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/4                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/5                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/6                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/7                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/8                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/9                     notconnect    1          auto    auto 10/100BaseTX
Fa1/0/10                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/11                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/12                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/13                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/14                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/15                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/15                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/17                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/18                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/19                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/20                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/21                    notconnect    1          auto    auto 10/100BaseTX
Fa1/0/22                    connected     trunk     a-full  a-100 10/100BaseTX
Fa1/0/23    CON-NYCORE2     connected     trunk      full    100 10/100BaseTX
```

```
Fa1/0/24  CON-NYCORE2      connected   trunk       full   100 10/100BaseTX

Gi1/0/1                    notconnect  1           auto   auto Not Present

Gi1/0/2                    notconnect  1           auto   auto Not Present
```

The interface status command conveniently shows you the description, speed, duplex and VLAN ids of the interfaces on your switch.

Compare this command on each NYCORE1, 2 and NYACCESS1 and you will notice some differences:

```
NYACCESS1#show interfaces status

Port       Name            Status      Vlan      Duplex Speed Type

Fa0/1                      connected   1         a-full a-100 10/100BaseTX

Fa0/2                      notconnect  1           auto   auto 10/100BaseTX

Fa0/3                      notconnect  1           auto   auto 10/100BaseTX

Fa0/4                      notconnect  1           auto   auto 10/100BaseTX

Fa0/5                      notconnect  1           auto   auto 10/100BaseTX

Fa0/6                      notconnect  1           auto   auto 10/100BaseTX

Fa0/7                      notconnect  1           auto   auto 10/100BaseTX

Fa0/8                      notconnect  1           auto   auto 10/100BaseTX

Fa0/9                      notconnect  1           auto   auto 10/100BaseTX

Fa0/10                     notconnect  1           auto   auto 10/100BaseTX

Fa0/11                     notconnect  1           auto   auto 10/100BaseTX

Fa0/12                     notconnect  1           auto   auto 10/100BaseTX

Fa0/13                     notconnect  1           auto   auto 10/100BaseTX

Fa0/14                     notconnect  1           auto   auto 10/100BaseTX

Fa0/15                     notconnect  1           auto   auto 10/100BaseTX

Fa0/16                     notconnect  1           auto   auto 10/100BaseTX

Fa0/17                     notconnect  1           auto   auto 10/100BaseTX

Fa0/18                     notconnect  1           auto   auto 10/100BaseTX

Fa0/19                     notconnect  1           auto   auto 10/100BaseTX

Fa0/20                     notconnect  1           auto   auto 10/100BaseTX

Fa0/21                     notconnect  1           auto   auto 10/100BaseTX

Fa0/22                     notconnect  1           auto   auto 10/100BaseTX
```

```
Fa0/23                           connected   trunk      full    100 10/100BaseTX

Fa0/24                           connected   trunk      a-full  a-100 10/100BaseTX
```

NYACCESS1 has different interface naming convention, the speed and duplex settings set to **a-full a-100** on a number of interfaces that are up, contrast this to **NYCORE1** where some of the interfaces are set to **full 100** this means these settings have been **manually assigned**.

NYCORE2 has the description setting that you configured previously, this can help a lot in troubleshooting, however don't always rely on it. This is a manual setting, and we all get lazy from time to time and forget to update our settings right.

Errdisable

Before we continue to the macro commands in the next exercise, let's do some testing, we need to cover off errdisable which is a condition that occurs when a switch notices **problems** on an interface and essentially shuts the port down because of these errors. A port can also go in to errdisable because of a security violation also.

In this example, as it's a reliable mechanism to get an interface to go in to an errdisable state, we are going to use port security.

Step 1

On **NYCORE2**, configure interface fastethernet 1/0/12 to have port-security enabled on it, use the following commands on fastethernet 1/0/12:

```
NYCORE2#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE2(config)#interface fas 1/0/12

NYCORE2(config-if)#switchport mode access

NYCORE2(config-if)#switchport port-security mac-address 0010.0020.0030

NYCORE2(config-if)#switchport port-security violation shutdown

NYCORE2(config-if)#switchport port-security
```

In this example we have configured port security to monitor the interface for only a MAC address of 0010.0020.0030 which (for this example) is a bogus address. Once traffic is generated on this switchport from a different mac address the port will shutdown:

```
*Mar  1 00:05:02.207: %PM-4-ERR_DISABLE: psecure-violation error detected on
Fa1/0/12, putting Fa1/0/12 in err-disable state

*Mar  1 00:05:02.207: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation
occurred, caused by MAC address 0013.c460.50f6 on port FastEthernet1/0/12.

*Mar  1 00:05:03.214: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet1/0/12, changed state to down
```

```
*Mar  1 00:05:03.919: %SYS-5-CONFIG_I: Configured from console by console

*Mar  1 00:05:04.397: %LINK-3-UPDOWN: Interface FastEthernet1/0/12, changed state
to down
```

Using the **show interfaces status** command we can see that indeed the interface is in an errdisable state.

```
NYCORE2#show interfaces status

Port        Name            Status      Vlan      Duplex  Speed Type

Fa1/0/1                     connected   1         a-full  a-100 10/100BaseTX

Fa1/0/2                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/3                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/4                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/5                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/6                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/7                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/8                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/9                     notconnect  1          auto    auto 10/100BaseTX

Fa1/0/10                    notconnect  1          auto    auto 10/100BaseTX

Fa1/0/11                    notconnect  1          auto    auto 10/100BaseTX

Fa1/0/12                    err-disabled 1         auto    auto 10/100BaseTX
```

Above you can see the status of the last interface (the rest has been omitted) is **err-disable**.

Step 2
There are a number of ways we bring an err-disable interface back online. First we can change the errdisable timer to automatically bring the interface back up. We can also manually bring the interface back online, using the **shutdown/no shutdown** interface commands.

```
shutdown

no shutdown
```

Alternatively we can automatically achieve this using the global configuration errdisable timer:

```
errdisable recovery interval 30
```

This would set the interface recovery to 30 seconds, this is helpful in the real world when an interface has entered the err-disable state because of errors and you want the interface to come back up automatically, image a key servers interface does this in the middle of the night, you don't

want to be logging in to reset the interface right! But you at least want to know about it, that's where logs and network management tools can be handy. You must rectify this issue if it happens because of errors and not because of a port-security violation.

```
NYCORE2#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE2(config)#errdisable recovery interval 30
```

Errdisable Detection

Tuning the errdisable detection is a straightforward process, I have never had to touch these settings, however you can tune them using the **errdisable detect cause** global configuration command:

```
errdisable detect cause
```

The output of this command on NYCORE2:

```
NYCORE2(config)#errdisable detect cause ?

  all                  Enable error detection on all cases

  arp-inspection       Enable error detection for arp inspection

  bpduguard            Enable error detection on bpdu-guard

  dhcp-rate-limit      Enable error detection on dhcp-rate-limit

  dtp-flap             Enable error detection on dtp-flapping

  gbic-invalid         Enable error detection on gbic-invalid

  inline-power         Enable error detection for inline-power

  l2ptguard            Enable error detection on l2protocol-tunnel

  link-flap            Enable error detection on linkstate-flapping

  loopback             Enable error detection on loopback

  pagp-flap            Enable error detection on pagp-flapping

  pppoe-ia-rate-limit  Enable error detection on PPPoE IA rate-limit

  security-violation   Enable error detection on 802.1x-guard

  sfp-config-mismatch  Enable error detection on SFP config mismatch

  small-frame          Enable error detection on small_frame
```

Using the question mark shows all the available options for this setting.

Leave your devices in their current states and continue to the next exercise.

Exercise 2 – Interface Macro commands

In this exercise you will learn how to define and use interface range macros. Use your course information or use your preferred search engine to gain an understanding of these tasks.

Defining a macro

Interface range macros, if they are maintained well can be a fantastic time.

A range macro defines a set of interfaces that you can use in the interface range command to shortcut entering in all the interfaces that you want to configure.

This can be incredibly useful, for example your Microsoft Exchange administrator installs a new set of servers and may require you to change settings over a period of time (let's be honest, those server guys never get it right first time). So, being smart you configure a macro called EXCHANGE-SERVERS and whenever the admin comes to your desk to need a setting changed, you can smile knowing that you are on top of it already with your new found macro!

Step 1

Using NYCORE1, create a macro called EXCHANGE-SERVERS. Simply use the following command (in this example we will just add a range of random interfaces):

```
NYCORE1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE1(config)#define interface-range EXCHANGE-SERVERS fastethernet 1/0/5-10
```

This would add fastethernets 1/0/5, through to fastethernet 1/0/10 to the range command when we invoke it.

Step 2

Invoking the range command and using it is equally as simple. First to invoke we use:

```
interface range macro EXCHANGE-SERVERS
```

```
NYCORE1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE1(config)#interface range macro EXCHANGE-SERVERS
```

Then we are able to configure the settings:

```
description Exchange server

speed 100
```

```
duplex full

switchport mode access

spanning-tree portfast


NYCORE1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.

NYCORE1(config)#interface range macro EXCHANGE-SERVERS

NYCORE1(config-if-range)#description Exchange Server

NYCORE1(config-if-range)#speed 100

NYCORE1(config-if-range)#duplex full

NYCORE1(config-if-range)#switchport mode access

NYCORE1(config-if-range)#spanning-tree portfast

%Warning: portfast should only be enabled on ports connected to a single

 host. Connecting hubs, concentrators, switches, bridges, etc... to this

  interface  when portfast is enabled, can cause temporary bridging loops.

  Use with CAUTION

%Portfast will be configured in 6 interfaces due to the range command

 but will only have effect when the interfaces are in a non-trunking mode.

NYCORE1(config-if-range)#
```

Or even better, we can simply use:

```
description Exchange server

speed 100

duplex full

switchport host
```

For some additional points, try and find out what the **switchport host** command does.

Step 3
Viewing our configuration:

```
NYCORE1#show interfaces status

Port        Name                Status       Vlan        Duplex  Speed Type

Fa1/0/1                         notconnect   1              auto    auto 10/100BaseTX

Fa1/0/2                         connected    1            a-full  a-100 10/100BaseTX
```

```
Fa1/0/3                         notconnect   1          auto    auto 10/100BaseTX

Fa1/0/4                         notconnect   1          auto    auto 10/100BaseTX

Fa1/0/5   Exchange Server       notconnect   1          full     100 10/100BaseTX

Fa1/0/6   Exchange Server       notconnect   1          full     100 10/100BaseTX

Fa1/0/7   Exchange Server       notconnect   1          full     100 10/100BaseTX

Fa1/0/8   Exchange Server       notconnect   1          full     100 10/100BaseTX

Fa1/0/9   Exchange Server       notconnect   1          full     100 10/100BaseTX

Fa1/0/10  Exchange Server       notconnect   1          full     100 10/100BaseTX
```

Notice that all the interfaces in the macro now have the correct description and they are now set to 100Mbs Full Duplex.

Summary

In this module you achieved the following activities:

- You learnt how to configure the speed and duplex of an interface.
- You learnt how to assign an IP address on VLAN 1.
- You learnt how to configure a range of interfaces using the range command.
- You learnt how to configure a macro to configure a range of interfaces.
- You observed the output of the show interfaces status command confirming your configuration.

Also Try

Using your lab infrastructure you can attempt the following topics at your own pace, these are additional tasks that can be done building on what you have learnt in this module:

- Create a new macro for interfaces 1/0/12 through 18. Call the macro DNS-SERVERS.
- Configuring a range of interfaces that are not sequential, for example fastethernet 1/0/5, 1/0/9 and 1/0/15-18 on NYCORE2