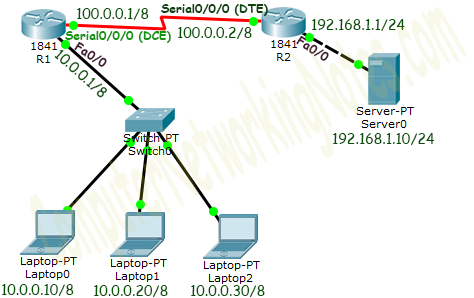
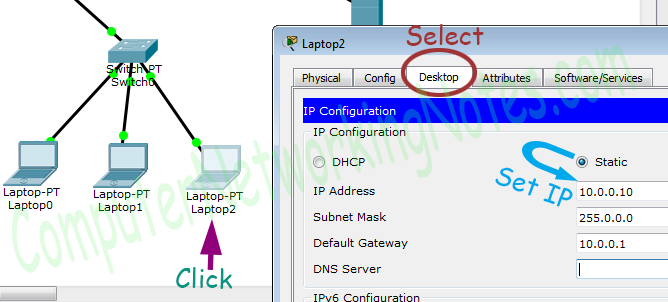
Create a lab as illustrates in following figure.



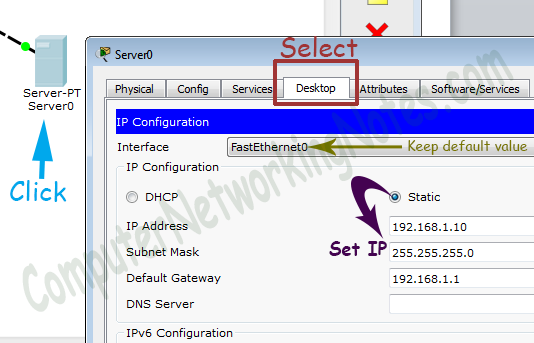
Initial IP Configuration

|  |  |  |
| --- | --- | --- |
| **Device / Interface** | **IP Address** | **Connected With** |
| Laotop0 | 10.0.0.10/8 | Fa0/0 of R0 |
| Laptop1 | 10.0.0.20/8 | Fa0/0 of R0 |
| Laptop2 | 10.0.0.30/8 | Fa0/0 of R0 |
| Server0 | 192.168.1.10/24 | Fa0/0 of R1 |
| Serial 0/0/0 of R1 | 100.0.0.1/8 | Serial 0/0/0 of R2 |
| Serial 0/0/0 of R2 | 100.0.0.2/8 | Serial 0/0/0 of R2 |

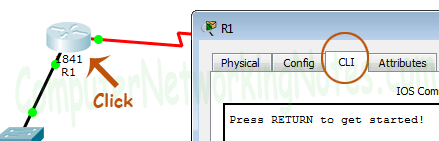
To assign IP address in Laptop click **Laptop** and click **Desktop** and click **IP configuration** and Select **Static** and set **IP address** as given in above table.



Following same way configure IP address in Server.



To configure IP address in Router1 click **Router1** and select **CLI** and press **Enter key**.



Run following commands to set IP address and hostname.

Router>enable

Router# configure terminal

Router(config)#

Router(config)#hostname R1

R1(config)#interface FastEthernet0/0

R1(config-if)#ip address 10.0.0.1 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface Serial0/0/0

R1(config-if)#ip address 100.0.0.1 255.0.0.0

R1(config-if)#clock rate 64000

R1(config-if)#bandwidth 64

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#

Same way access the command prompt of R2 and run following commands to set IP address and hostname.

Router>enable

Router#configure terminal

Router(config)#hostname R2

R2(config)#interface FastEthernet0/0

R2(config-if)#ip address 192.168.1.1 255.255.255.0

R2(config-if)#no shutdown

R2(config-if)#exit

R2(config)#interface Serial0/0/0

R2(config-if)#ip address 100.0.0.2 255.0.0.0

R2(config-if)#no shutdown

R2(config-if)#exit

R2(config)#

That’s all initial IP configuration we need. Now this topology is ready for the practice of pat.

Configure PAT (NAT Overload)

PAT configuration requires four steps: -

* Create an access list of IP addresses which need translation
* Create a pool of all IP address which are available for translation
* Map access list with pool
* Define inside and outside interfaces

In first step we will create a standard access list which defines which inside local addresses are permitted to map with inside global address.

To create a standard numbered ACL following global configuration mode command is used:-

Router(config)# access-list ACL\_Identifier\_number permit/deny matching-parameters

Let’s understand this command and its options in detail.

**Router(config)#**

This command prompt indicates that we are in global configuration mode.

**access-list**

Through this parameter we tell router that we are creating or accessing an access list.

**ACL\_Identifier\_number**

With this parameter we specify the type of access list. We have two types of access list; standard and extended. Both lists have their own unique identifier numbers. Standard ACL uses numbers range 1 to 99 and 1300 to 1999. We can pick any number from this range to tell the router that we are working with standard ACL. This number is used in groping the conditions under a single ACL. This number is also a unique identifier for this ACL in router.

**permit/deny**

An ACL condition has two actions; permit and deny. If we use permit keyword, ACL will allow all packets from the source address specified in next parameter. If we use deny keyword, ACL will drop all packets from the source address specified in next parameter.

**matching-parameters**

This parameter allows us to specify the contents of packet that we want to match. In a standard ACL condition it could be a single source address or a range of addresses. We have three options to specify the source address.

* Any
* host
* A.B.C.D

Any

Any keyword is used to match all sources. Every packet compared against this condition would be matched.

Host

Host keyword is used to match a specific host. To match a particular host, type the keyword host and then the IP address of host.

A.B.C.D

Through this option we can match a single address or a range of addresses. To match a single address, simply type its address. To match a range of addresses, we need to use wildcard mask.

Wildcard mask

Just like subnet mask, wildcard mask is also used to draw a boundary in IP address. Where subnet mask is used to separate network address from host address, wildcard mask is used to distinguish the matching portion from the rest. Wildcard mask is the invert of Subnet mask. Wildcard can be calculated in decimal or in binary from subnet mask.

We have three hosts in lab. Let’s create a standard access list which allows two hosts and denies one host.

R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0

R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0

R1(config)#access-list 1 deny any

To learn standard ACL in detail you can use following tutorial.

[Standard ACL Explained with Examples](http://www.computernetworkingnotes.com/ccna-study-guide/standard-acl-configuration-commands-explained.html)

In second step we define a pool of inside global addresses which are available for translation.

Following command is used to define the NAT pool.

Router(config)#ip nat pool [Pool Name] [Start IP address] [End IP address] netmask [Subnet mask]

This command accepts four options pool name, start IP address, end IP address and Subnet mask.

**Pool Name**: - This is the name of pool. We can choose any descriptive name here.

**Start IP Address**: - First IP address from the IP range which is available for translation.

**End IP Address**: - Last IP address from the IP range which is available for translation. There is no minimum or maximum criteria for IP range for example we can have a range of single IP address or we can have a range of all IP address from a subnet.

**Subnet Mask**: - Subnet mask of IP range.

Let’s create a pool named ccna with a single IP address.

R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.1 netmask 255.0.0.0

In third step we map access list with pool. Following command will map the access list with pool and configure the PAT.

Router(config)#ip nat inside source list [access list name or number] pool [pool name]overload

This command accepts two options.

**Access list name or number**: - Name or number the access list which we created in first step.

**Pool Name**: - Name of pool which we created in second step.

In first step we created a standard access list with number 1 and in second step we created a pool named ccna. To configure a PAT with these options we will use following command.

R1(config)#ip nat inside source list 1 pool ccna overload

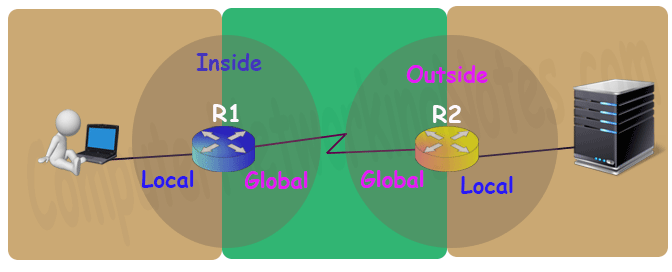
Finally we have to define which interface is connected with local network and which interface is connected with global network.

To define an inside local we use following command

Router(config-if)#ip nat inside

Following command defines inside global

Router(config-if)#ip nat outside



Let’s implement all these commands together and configure the PAT.

R1 PAT (NAT Overload) Configuration

R1>enable

R1#configure terminal

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

R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0

R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0

R1(config)#access-list 1 deny any

R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.1 netmask 255.0.0.0

R1(config)#ip nat inside source list 1 pool ccna overload

R1(config)#interface FastEthernet 0/0

R1(config-if)#ip nat inside

R1(config-if)#exit

R1(config)#interface Serial 0/0/0

R1(config-if)#ip nat outside

R1(config-if)#exit

R1(config)#

For testing purpose I configured pat translations for two addresses only.

On R2 we can keep standard configuration or can configure dynamic NAT or can configure static NAT as we learnt in pervious parts of this article.

Let’s do a quick recap of what we learnt in previous part and configure static NAT on R2.

R2>enable

R2#configure terminal

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

R2(config)#ip nat inside source static 192.168.1.10 200.0.0.10

R2(config)#interface Serial 0/0/0

R2(config-if)#ip nat outside

R2(config-if)#exit

R2(config)#interface FastEthernet 0/0

R2(config-if)#ip nat inside

R2(config-if)#exit

R2(config)#

To understand above commands in detail please see the second part of this tutorial.

Before we test this lab we need to configure the IP routing. IP routing is the process which allows router to route the packet between different networks. Following tutorial explain routing in detail with examples

[Routing Protocol Explained](http://www.computernetworkingnotes.com/ccna-study-guide/basic-routing-concepts-and-protocols-explained.html)

Configure static routing in R1

R1(config)#ip route 200.0.0.0 255.255.255.0 100.0.0.2

Configure static routing in R2

R2(config)#ip route 50.0.0.0 255.0.0.0 100.0.0.1

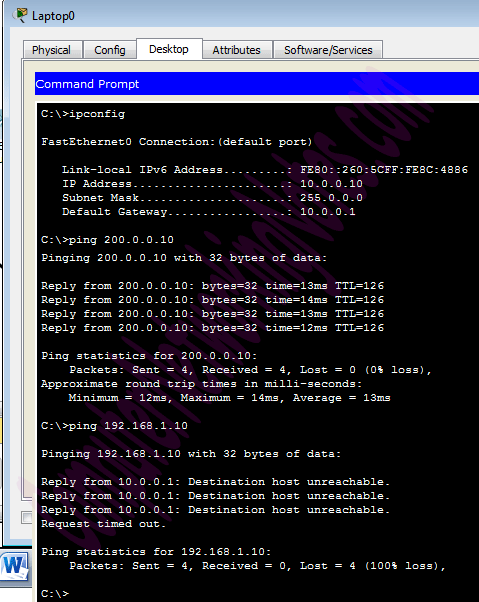
Testing PAT Configuration

In this lab we configured PAT on R1for 10.0.0.10 and 10.0.0.20 and static NAT on R2 for 192.168.1.10.

|  |  |  |
| --- | --- | --- |
| **Device** | **Inside Local IP Address** | **Inside Global IP Address** |
| Laptop0 | 10.0.0.10 | 50.0.0.1 |
| Laptop1 | 10.0.0.20 | 50.0.0.2 |
| Server | 192.168.1.10 | 200.0.0.10 |

To test this setup click **Laptop0** and **Desktop** and click **Command Prompt**.

* Run **ipconfig** command.
* Run **ping 200.0.0.10** command.
* Run **ping 192.168.1.10** command.

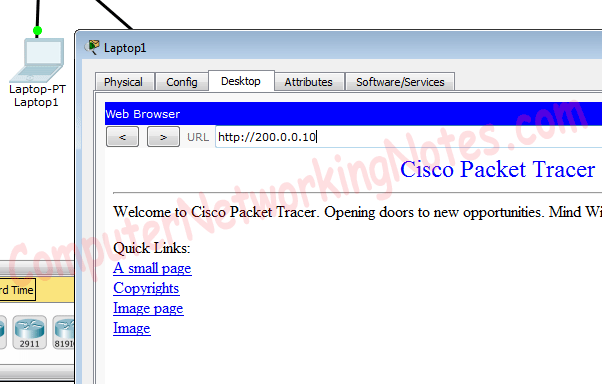


First command verifies that we are testing from correct NAT device.

Second command checks whether we are able to access the remote device or not. A ping reply confirms that we are able to connect with remote device on this IP address.

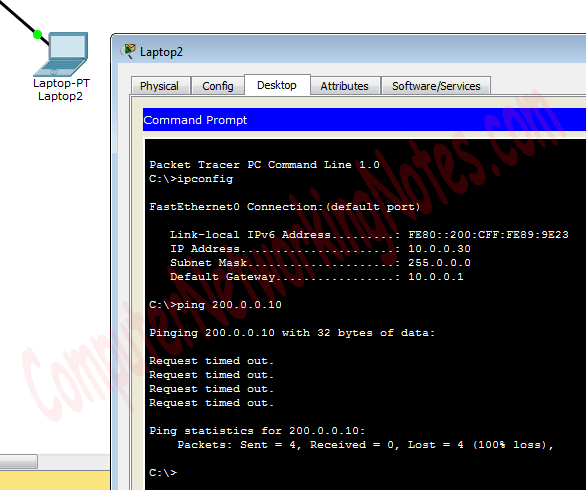
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

Let’s do one more testing. Close the command prompt and click web server and access 200.0.0.10.

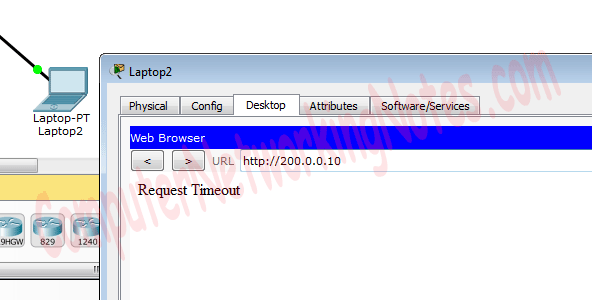


Above figure confirms that host 10.0.0.10 is able to access the 200.0.0.10. You can also do the same testing from Laptop1, result will be same.

Now run **ping 200.0.0.10** command from Laptop2.



Close the command prompt and access web server from this host.



*Why we are not able to connect with the remote device from this host?*

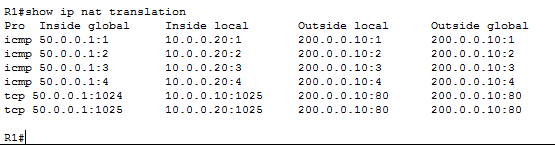
Because we configured PAT only for two hosts (Laptop0 and Laptop1) which IP addresses are 10.0.0.10 and 10.0.0.20. So only the host 10.0.0.10 and 10.0.0.20 will be able to access the remote device.

*If you followed this tutorial step by step, you should get the same output of testing. Although it’s very rare but some time you may get different output. To figure out what went wrong you can use my practice topology with all above configuration. Download my practice topology*

[Download NAT Practice LAB with PAT configuration](https://www.computernetworkingnotes.com/download/cisco/ccna-study-guide/pkt/nat-practice-pat-configured.pkt)

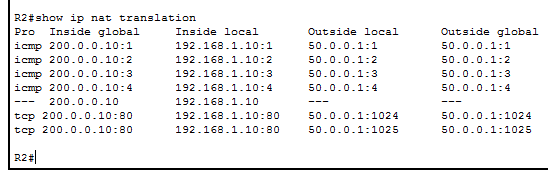
We can also verify this translation on router with ***show ip nat translation*** command.

Following figure illustrate this translation on router R1.



As we can see in above output same inside global IP address is used to translate all the inside local IP addresses. For each inside local IP address a unique port number is used.

Following figure illustrate NAT translation on router R2



In above output the Outside global field also confirms that all packets are coming from single IP address.

That’s all for this article.