

# Lab 3: Access Control Lists; Network Address Translation

## Contents

Introduction to LAB 3 .....	2
Exercise 1: Implement security with ACLs .....	2
Task 1: Prepare the topology .....	2
Task 2: Enable DNS and WEB services.....	4
Task 3: Create and apply ACL to restrict client access .....	8
Exercise 2: Configure NAT .....	10
Task 1: Prepare the topology .....	11
Task 2: Configure NAT between the local network and Internet.....	13
Task 3: Test the connectivity and verify the address translations .....	15
Useful commands for checking your configurations and troubleshooting.....	17

## Introduction to LAB 3

In the first exercise of this lab, you will practice with Access Control List to understand how they can be used to protect the network. In the second exercise, you will configure Network Address Translation (more specifically Port Address Translation) to translate private IP addresses into a public one, which is routable in Internet.

### Exercise 1: Implement security with ACLs

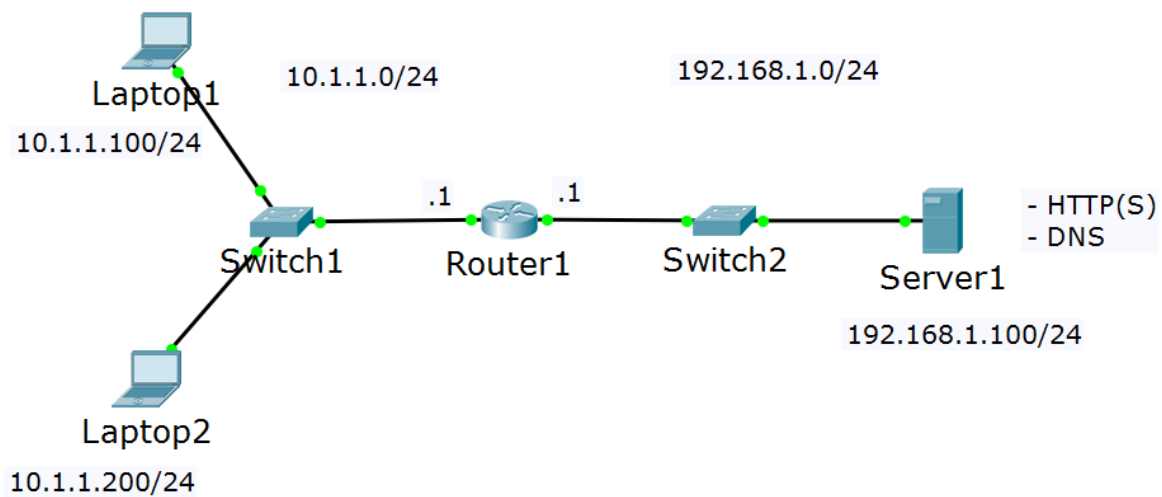
#### Task 1: Prepare the topology

##### 1. Create the physical topology

In the packet tracer, move the following devices to the workspace:

- Two end devices (use Laptop-PT, the second in the list)
- Two switches (2960)
- One router (2911)
- One server (Server-PT, the third one from the End Devices list)

Rename and connect the devices as per the picture below (IP addressing will be discussed in a second)



## 2. Assign the IP addresses.

This is a simple topology with two IP subnets: 10.1.1.0/24, which will be the client network and 192.168.1.0/24, which will be the server network. Refer to the table for the exact IP address assignments:

Device/Port	IP Address	Belongs to network (informational only)
<b>Laptop1</b>	10.1.1.100	10.1.1.0
<b>Laptop2</b>	10.1.1.200	10.1.1.0
<b>Router1/port-to-Switch1</b>	10.1.1.1	10.1.1.0
<b>Router1/port-to-Switch2</b>	192.168.1.1	192.168.1.0
<b>Server1</b>	192.168.1.100	192.168.1.0

Note: All masks are /24

## 3. Configure the connectivity

Both laptops and the server should be able to communicate. The “client” and the “server” networks are separated by a single router, which knows for both of these networks/subnets. As you have learned before, this is known as direct routing and no additional routing configuration on Router1 is required.

Still, you will need to set up default gateway addresses on the clients and on the server. Configure them as following:

- Both laptops should have default gateway of **10.1.1.1**
- The server should have default gateway of **192.168.1.1**

#### 4. Test the connectivity

Now you should be able to ping between all the devices. For example, open the CLI of Laptop1 and ping 10.1.1.200 (Laptop2) and 192.168.1.100 (Server1). If you do not have a successful ping, go back and check your topology, IP addresses and the default gateways.

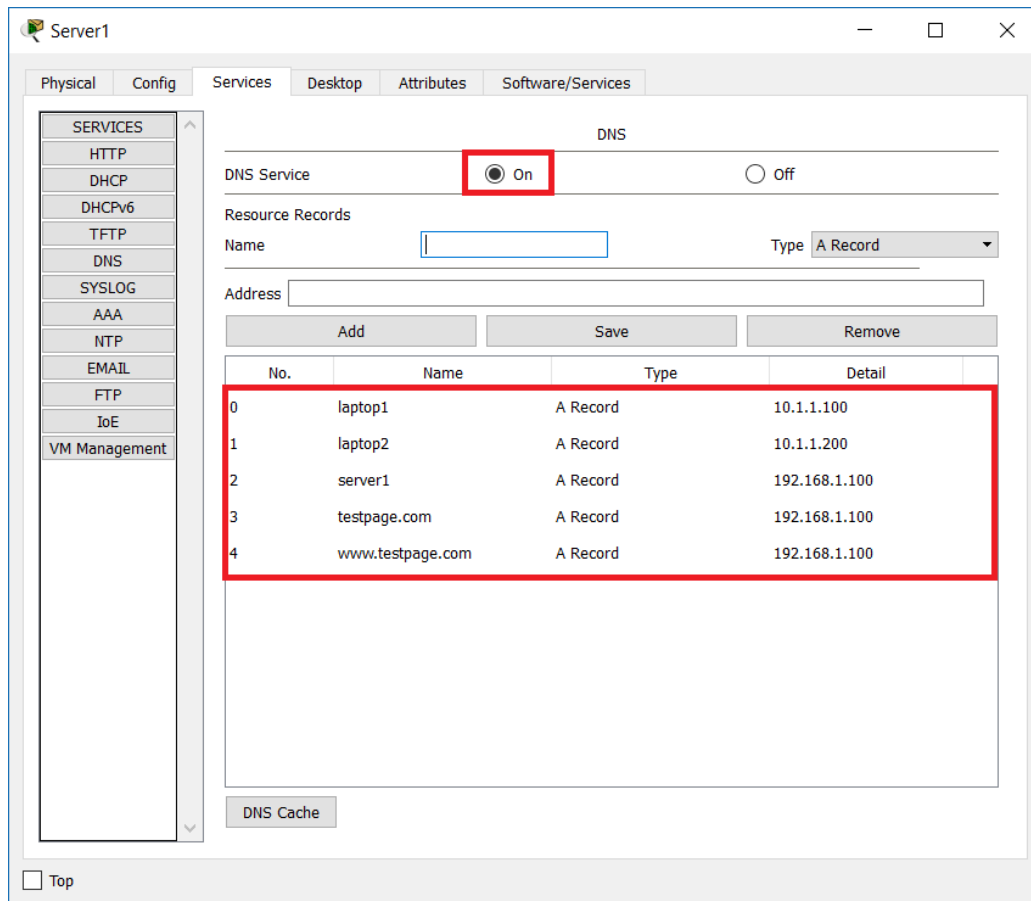
### Task 2: Enable DNS and WEB services

Server1 will act as DNS and WEB server. Each of these services exist in the Services tab on Server1 but need to be enabled and configured.

#### 1. Enable and configure the DNS server

Open the Services tab on Server1, go to DNS section and enable the service. Then, create the following “A” records:

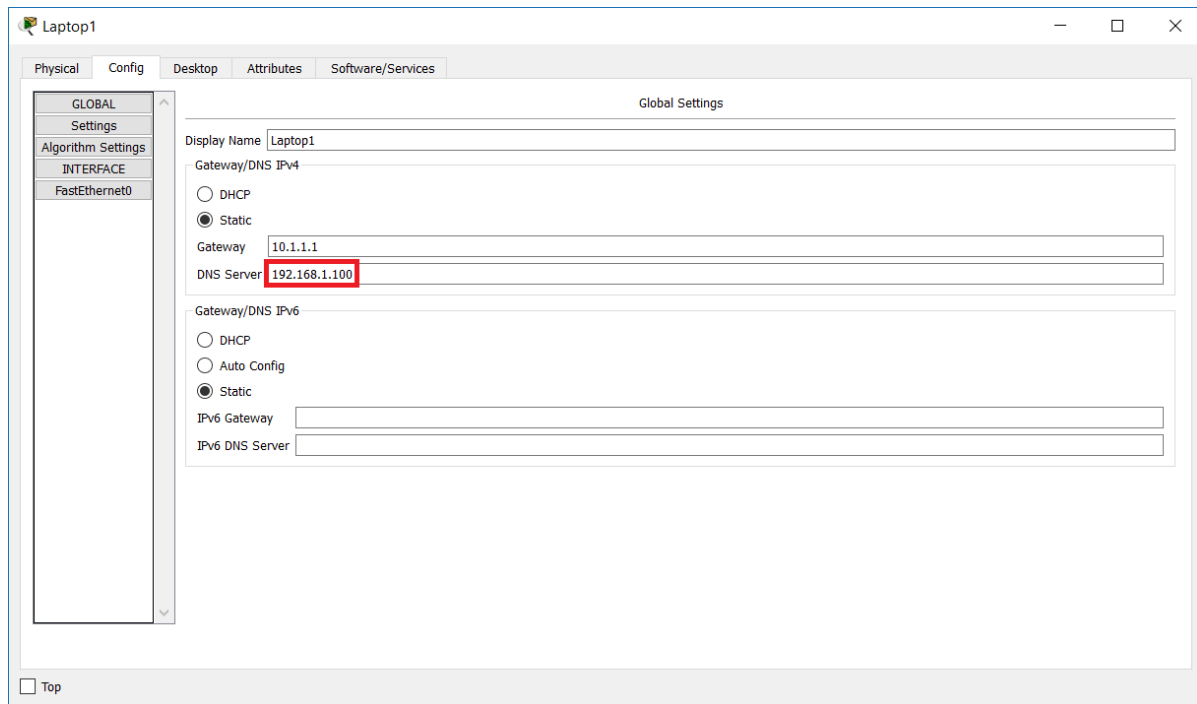
- laptop1 -> 10.1.1.100
- laptop2 -> 10.1.1.200
- server1 -> 192.168.1.100
- testpage.com -> 192.168.1.100
- www.testpage.com -> 192.168.1.100



Note that there are three records pointing to the server's IP address – server1, testpage.com and www.testpage.com.

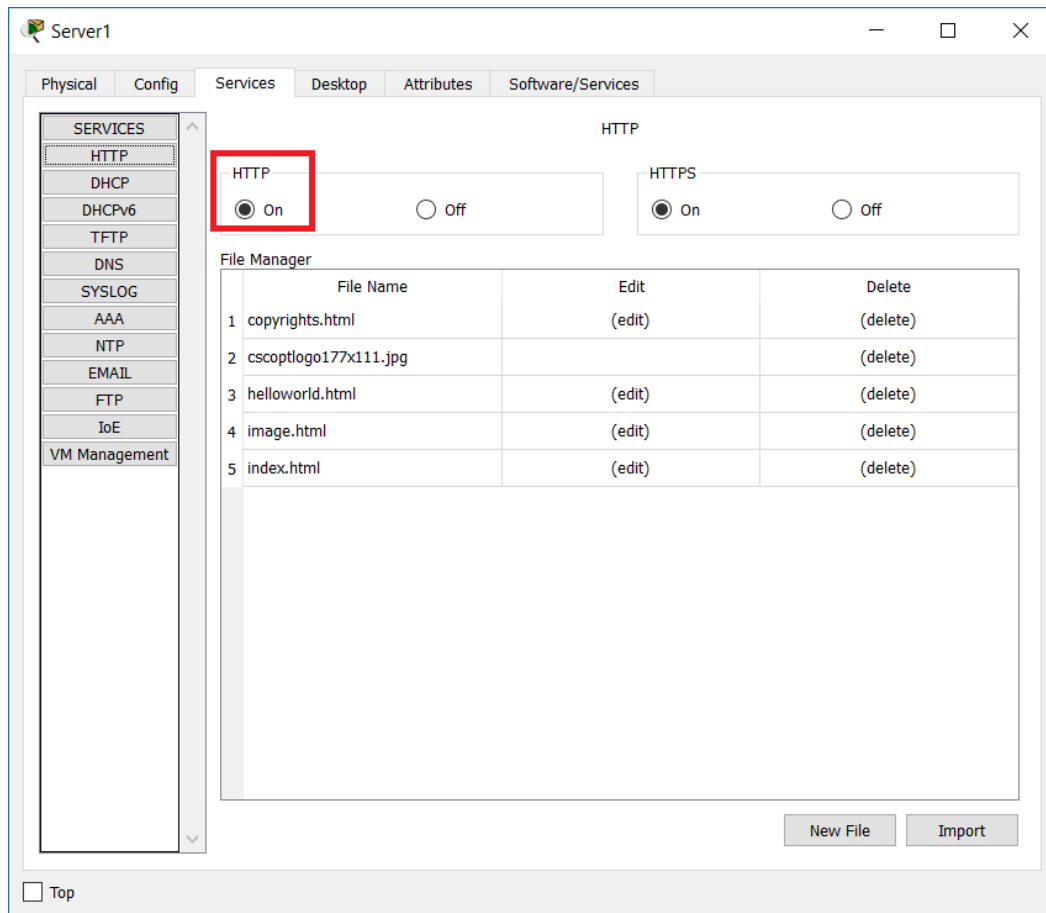
## 2. Configure the DNS clients.

We want to avoid additional complications and that is why you will setup the clients manually (instead of adding DHCP service). For each of the clients (Laptop1 and Laptop2), go to Config -> Settings and for DNS Server, put **192.168.1.100**



### 3. Enable and configure the WEB server

The WEB service should be enabled by default. To check this, go to Server1 -> Services -> HTTP and look if HTTP is set to On. If it is not, enable it



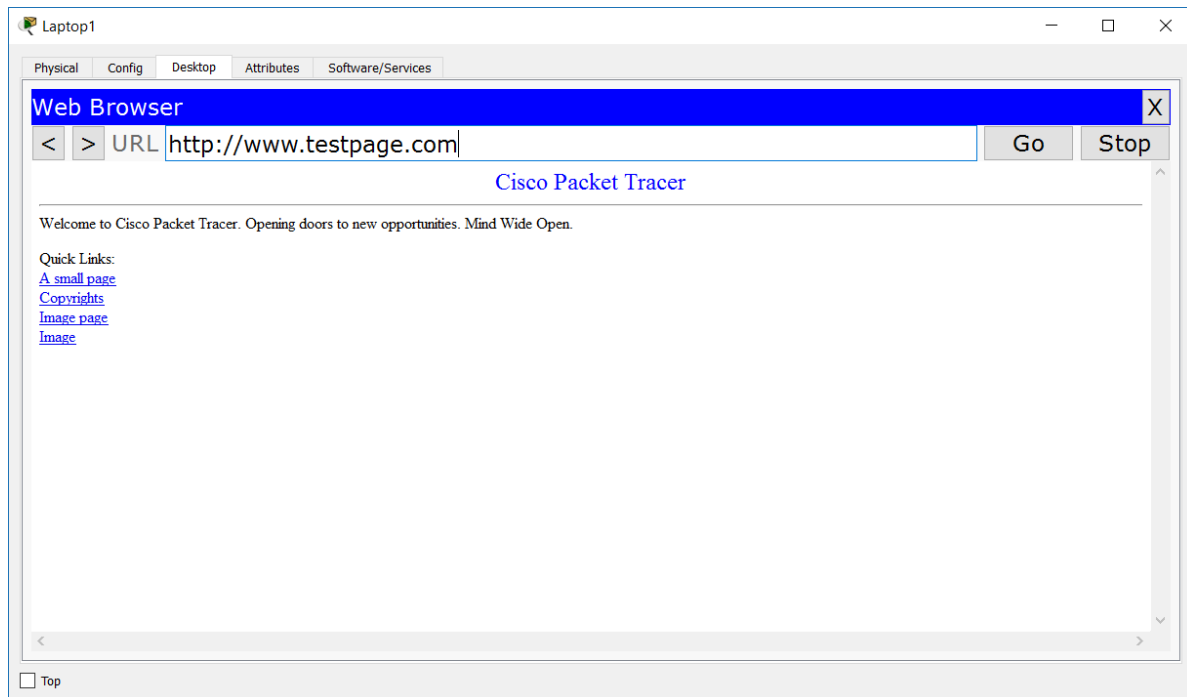
Optionally, you can create your own index.html file and upload it if you want to see a custom web page.

#### 4. Test DNS and WEB

From each of the clients (Laptop1 and Laptop2), do the following tests

- **ping laptop1**
- **ping laptop2**
- **ping server1**
- **ping testpage.com**
- **ping www.testpage.com**
- open a web browser and navigate to
  - **testpage.com**
  - **www.testpage.com**

All the pings should resolve to an IP addresses and should be successful. Also, the clients should be able to open the test page which looks like this



## Task 3: Create and apply ACL to restrict client access

### 1. Create an ACL

In this task, you will create an extended ACL which will have the following characteristics:

- restrict ping (ICMP) from both laptops to the server
- allow the laptops to use DNS
- restrict HTTP access from laptop1
- allow HTTP access from laptop2
- everything else should be denied



To do this, make the following configuration from global config mode on Router1:

```
ip access-list extended acl_1
permit udp host 10.1.1.100 host 192.168.1.100 eq domain
permit udp host 10.1.1.200 host 192.168.1.100 eq domain
deny icmp host 10.1.1.100 host 192.168.1.100 echo
deny icmp host 10.1.1.200 host 192.168.1.100 echo
permit tcp host 10.1.1.200 host 192.168.1.100 eq www
```

Explanations:

- the first and the second rules will allow DNS requests from both laptops to the server. Note that in this case we need port UDP 53. The keyword at the end **domain** is predefined and means exactly “DNS”, or “port 53”
- rules number 3 and 4 will deny ping from both laptops to the server (and vice-versa)
- rule number 5 (the last written one) will allow HTTP from laptop2 to the server. Note that this is tcp port 80. The keyword “www” is predefined and means “port 80”
- finally, remember that we have implicit “**deny ip any any**” at the end of each ACL. It is not written but it is always there. This will ensure that all other traffic will be denied

TIP: if you make a mistake or want to change something after you have created the ACL, the easiest way is to copy it to a notepad and edit it there. Then, delete the old ACL (with **no ip access-list extended acl\_1**), recreate it again and copy/paste from the notepad.

## 2. Apply the ACL

At this moment, the ACL will be in the configuration but will not be effective. The reason is that it is not applied to an interface. You can apply it

to either the interface which points to the laptops (inbound) or to the interfaces which points to the server (outbound). As a best practice, extended ACLs is better to be applied to the closest to the source interface.

To apply the ACL, go to the interface which points to the clients (in this example it is G0/0) and type:

**ip access-group acl\_1 in**

Now, the ACL is applied and should be effective.

### 3. Test the ACL

To test if you have achieved the desired results, do the following:

- From laptop1 and laptop2: **ping server1**  
The result should be that the IP address is resolved but the ping is not successful. You should see a message saying "Reply from 10.1.1.1: Destination host unreachable."
- From laptop1 open a web browser and type www.testpage.com  
The result should be Request Timeout
- From laptop2 open a web browser and type www.testpage.com  
The result should be that you see the web page
- From laptop1: **ping laptop2**  
The result should be that the IP address is resolved and also you will have successful ping. Why this happens? Because this traffic is local and does not go to the ACL inspection at all.

## Exercise 2: Configure NAT

In this exercise, you will have two simulated networks – a local network, which will have private IP addresses and public network (Internet) which will have public IP addresses. There will be a WEB server on the Internet which has to be reached

from the clients in the local network and their traffic should be translated to a public IP address in order to go through Internet

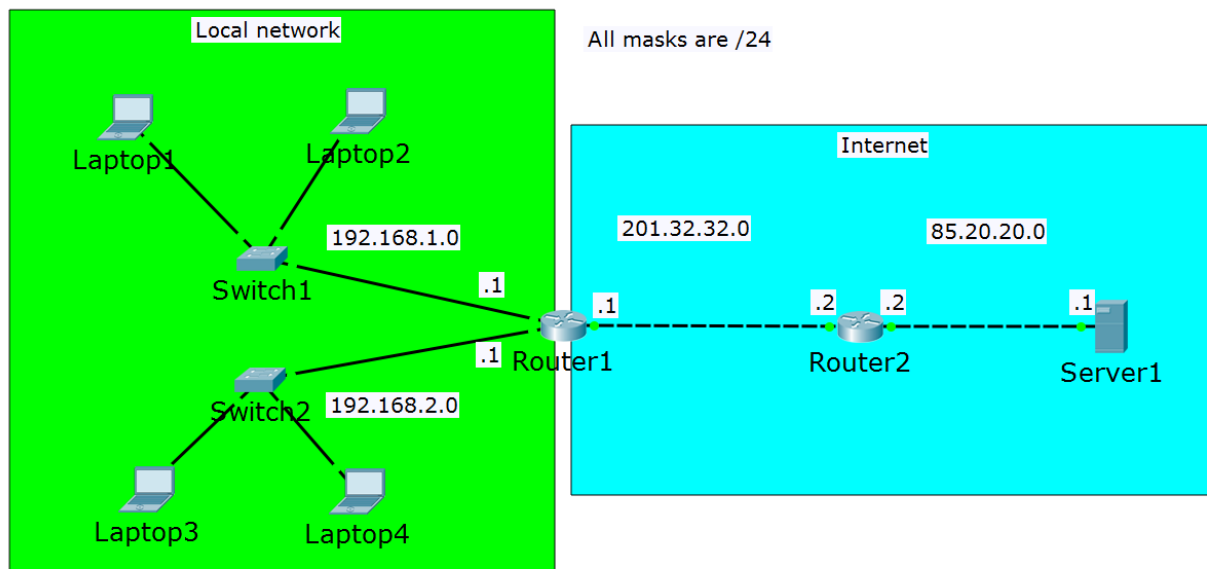
### Task 1: Prepare the topology

#### 1. Create the physical topology.

For the purposes of this exercise, you will need the following devices:

- Four end devices (use Laptop-PT, the second in the list)
- Two switches (2960)
- Two routers (2911)
- One server (Server-PT, the third one from the End Devices list)

Rename the devices to start from 1 and connect them as per the picture below.



#### 2. Setup IP addresses.

There are four subnets in this topology. Two of them will represent the local network (192.168.1.0/24 and 192.168.2.0/24) and two of them will represent networks in the Internet (201.32.32.0/24 and 85.20.20.0/24). You

can use the picture to help understand what IP address to put and you can consult the table below for the exact IP addresses that you need to configure for each interface. All masks are /24.

Device/Port	IP Address	Belongs to network (informational only)
Laptop1	192.168.1.100	192.168.1.0
Laptop2	192.168.1.200	192.168.1.0
Router1/port-to-Switch1	192.168.1.1	192.168.1.0
Laptop3	192.168.2.100	192.168.2.0
Laptop4	192.168.2.200	192.168.2.0
Router1/port-to-Switch2	192.168.2.1	192.168.2.0
Router1/port-to-Router2	201.32.32.1	201.32.32.0
Router2/port-to-Router1	201.32.32.2	201.32.32.0
Router2/port-to-Server1	85.20.20.2	85.20.20.0
Server1	85.20.20.1	85.20.20.0

### 3. Configure connectivity

- a. Imagine that you are only responsible for the local network configuration. What do you need to do? Well, one thing is to configure Router1 to be default gateway for all the clients.  
Setup **192.168.1.1** to be default gateway for laptop1 and laptop2 and **192.168.2.1** to be default gateway for laptop3 and laptop4
- b. Now imagine there is another guy who is responsible for the external network and the server. There are two things that need to be configured there:
  - Default gateway for Server1. Configure it to be **85.20.20.2**
  - Route from Router1 to network 85.20.20.0/24. To do this, type the following command from global config mode of Router1:

**ip route 85.20.20.0 255.255.255.0 201.32.32.2**

What is the result of these configurations? First, the two local subnets can reach each-other (test it) and second, all the laptops have gateways so they are ready to exit the local network.

On the other side, there is now connectivity between Router1 and Server1 (Router1 has a route to the remote network and Server1 has a default gateway).

But what about the routing between the local network and Server1? You do not have it. If you initiate a ping from one of the laptops to Server1, you can monitor (with the simulation mode) that the packet will reach the server - the laptops have default gateway and Router1 has a route to the Server's network. But it cannot return – Server1 will send it to Router2 (because it has a default gateway), but Router2 does not have a route to the local subnets. And if you think about it, this is normal – the devices in internet usually does not have route to your private home network (that is why it is private).

In the next task, you will configure Network Address Translation (or more specifically Port Address Translation), so the private IP addresses will be translated to a public one and vice-versa and because of this, the laptops will be able to reach the server.

## Task 2: Configure NAT between the local network and Internet

1. Select the local (private) addresses with a standard ACL

In the first exercise, you used ACLs for security. This is not the only usage of ACLs. They can be used just to select traffic which can be used later for something else, like NAT. This means that if there is a match in the ACL (permit), this traffic will be selected to be translated.

Note: Remember that if you want to use ACL to filter traffic (for security) you need also to apply the ACL to an interface – we will not do this step here, we just need to match a particular traffic.

To match all the private addresses from both subnets in the local network, enter the following configuration in global config mode of Router1:

**access-list 1 permit 192.168.1.0 0.0.0.255**

**access-list 1 permit 192.168.2.0 0.0.0.255**

Now, access-list 1 matches both networks 192.168.1.0/24 and 192.168.2.0/24 – these are the private networks, which will be used for inside local

2. Enter the “**ip nat inside**” command

Now that you have ACL that matches all internal networks that need to be translated, you need to define the address translations with the **ip nat inside** command on Router1:

**ip nat inside source list 1 interface gigabitEthernet 0/2 overload**

Note: In this example gigabitEthernet 0/2 is the interface pointing to Router2, in your topology it may be a different one.

Explanations:

- The **ip nat inside source** part of the command shows that:
  - The router will translate the source of IP packets that are traveling inside to outside (the outgoing packets)
  - The router will translate the destination of the IP packets that are traveling outside to inside (the returning packets)
- The **list 1** part of the command shows that the inside packets are those which match the ACL 1 (remember that access-list 1 matches both networks 192.168.1.0/24 and 192.168.2.0/24)
- The **interface gigabitEthernet 0/2** part of the command shows that the IP address configured there (201.32.32.1) will be used instead of any internal source IP address
- The **overload** part of the command shows that multiple internal/local IP addresses can be mapped to a single global (internet routable) IP address, which in fact is PAT (port address translation). Either if you

do not type **overload** during the configuration, it will automatically append in this case and you will see it in the **show run** command

3. Specify the inside and outside interfaces

Router1 interfaces pointing to Switch1 and Switch2 (in this example g0/0 and g0/1) should be configured with **ip nat inside** and the interface pointing to Router2 (going to Internet) should be configured as **ip nat outside**

Configure the following from global config mode in Router1:

```
int g0/0
ip nat inside
```

```
int g0/1
ip nat inside
```

```
int g0/2
ip nat outside
```

Note: Again, please note that in your topology you may have different interfaces pointing to Switch1, Switch2 and Router2.

### Task 3: Test the connectivity and verify the address translations

1. Test the connectivity

From all laptops, try to **ping 85.20.20.1** (Server1). It should now succeed. Why the pings are now successful?

You still do not have routing between the private subnets (192.168.1.0/24 and 192.168.2.0/24) and the server network (85.20.20.0/24). But the traffic from these subnets is now translated to 201.32.32.1 and you already have a route between 201.32.32.0/24 and 85.20.20.0/24 – that is the reason for the successful connectivity.

Note: You can also enable and test other services on the server, for example HTTP and DNS.

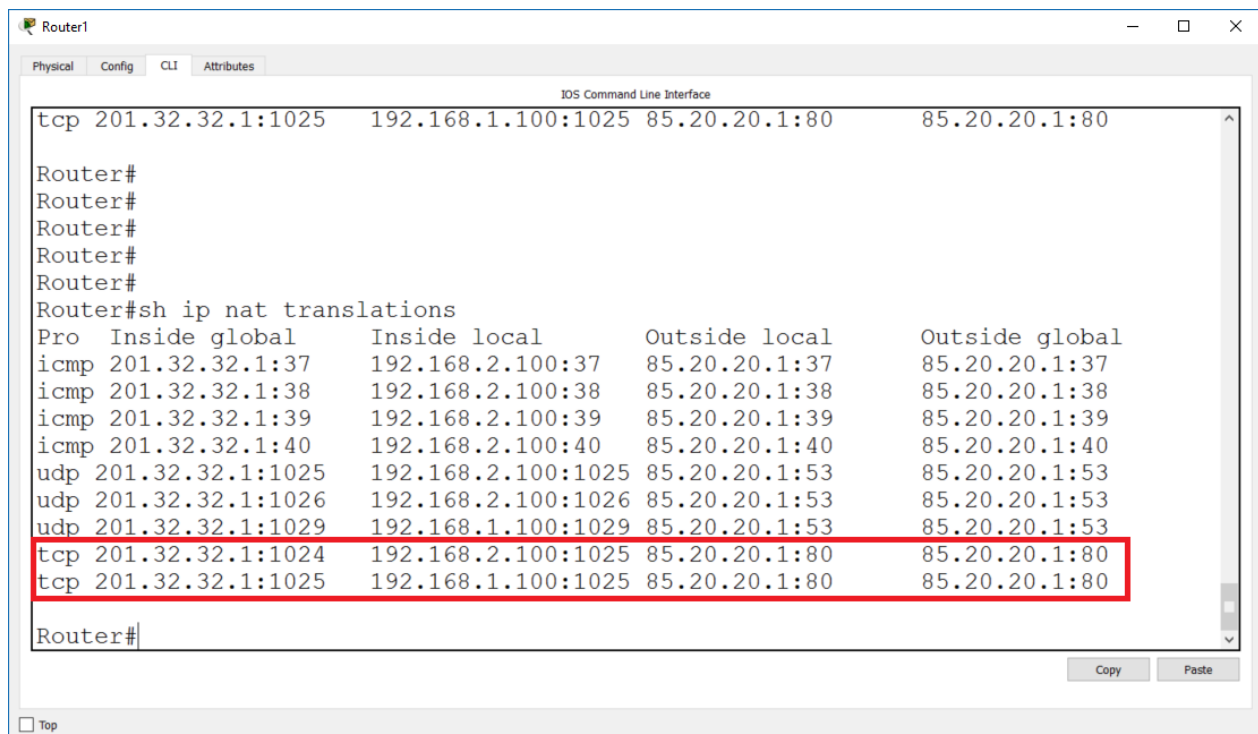
## 2. Verify the address translations

There are two options that you can use to check if your private addresses are translated into a public one.

First, you can use the simulation mode in the packet tracer where in the source IP field of the packet you should see 201.32.32.1 instead of a private address (192.168.1.X or 192.168.2.X).

The second option is to check Router1 translations with the command

**show ip nat translations**



```
Router1
Physical Config CLI Attributes
IOS Command Line Interface
tcp 201.32.32.1:1025 192.168.1.100:1025 85.20.20.1:80 85.20.20.1:80
Router#
Router#
Router#
Router#
Router#
Router#sh ip nat translations
Pro Inside global Inside local Outside local Outside global
icmp 201.32.32.1:37 192.168.2.100:37 85.20.20.1:37 85.20.20.1:37
icmp 201.32.32.1:38 192.168.2.100:38 85.20.20.1:38 85.20.20.1:38
icmp 201.32.32.1:39 192.168.2.100:39 85.20.20.1:39 85.20.20.1:39
icmp 201.32.32.1:40 192.168.2.100:40 85.20.20.1:40 85.20.20.1:40
udp 201.32.32.1:1025 192.168.2.100:1025 85.20.20.1:53 85.20.20.1:53
udp 201.32.32.1:1026 192.168.2.100:1026 85.20.20.1:53 85.20.20.1:53
udp 201.32.32.1:1029 192.168.1.100:1029 85.20.20.1:53 85.20.20.1:53
tcp 201.32.32.1:1024 192.168.2.100:1025 85.20.20.1:80 85.20.20.1:80
tcp 201.32.32.1:1025 192.168.1.100:1025 85.20.20.1:80 85.20.20.1:80
Router#
```

Look at the output in the screenshot above and especially on the last two lines which are highlighted. The first of them shows that a web session with source IP 192.168.2.100 and source port 1025 is translated to source IP 201.32.32.1 and source port 1024. The second one shows that a web



session with source IP 192.168.1.100 and source port 1025 is translated to source IP 20.1.32.32.1 and source port 1025. And this is exactly the idea of PAT (port address translation) – multiple private addresses (and their ports) are translated into a single public address with different source port numbers – that is how the router knows where exactly to send the returning traffic.

You have completed LAB 3.

[Useful commands for checking your configurations and troubleshooting](#)

**show run | begin access**

**show run**

**show ip nat translations**

**show ip nat statistics**