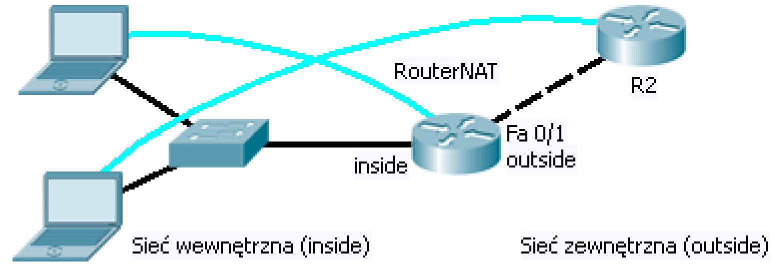
COMPUTER NETWORKS - LABORATORY 073

Subject:

Cisco IOS – configuring NAT. Overloading, static translations and DMZ

Task A: NAT - overloading.

1. Prepare installation consisting of one Cisco router connected to two networks with different addresses - through Fast Ethernet interfaces.
   For the duration of experiments specify one of these networks as the "outside" (public network), and second as “inside” (private, internal).
   Place two computers in “inside” network (using Ethernet switch). Place another router in “outside” network.



   In this configuration, the external router (R2) will simulate a service in the Internet. The NAT translation will affect all datagrams coming to that server from inside and back.
   Outside interface address of the NAT router will be used as a source address of all IP datagrams, leaving the internal network. Subsequently hosts from internal network will not be visible to the external network users (NAT will cause the translation of computer addresses into one official address. Computer will therefore be masked by NAT in public network.

2. Enable proper IP addressing on NAT router interfaces, for example:

```
RouterNAT (config) # ip routing
RouterNAT (config) # int fa 0 / 0
RouterNAT (config-if) # ip address 10.0.0.1 255.255.255.0
RouterNAT (config-if) # exit
RouterNAT (config) # int fa 0 / 1
RouterNAT (config-if) # ip address 200.200.200.1 255.255.255.0
```

   Also define an IP address of R2:

```
R2 (config) # int fa 0 / 0
R2 (config-if) #ip address 200.200.200.2, 255.255.255.0
```

3. NAT traffic is controlled by access control lists (ACL). Define the access list for NAT. The list must permit outbound traffic from the inside network with a source address pool os inside network:

   ```
   RouterNAT(config)#access-list 5 permit 10.0.0.0 0.0.0.255
   ```

   Then assign the ACL to the NAT, defining the translation rule:

   ```
   RouterNAT (config) #ip nat inside source list 5 interface fa 0/1 overload
   ```

   where the overload starts aggregating addresses to be translated back when packet returns in one-to-many.

4. Specify the side of NAT interface to an internal network (inside):

   ```
   RouterNAT(config)#int fa 0 / 0
   RouterNAT (config-if) #ip nat inside
   ```

5. Specify the side of NAT interface to an external network (outside):

   ```
   RouterNAT(config) #int fa 0 / 1
   RouterNAT (config-if) #ip nat outside
   ```

6. ICMP test:
   Test the NAT using internal network (inside) with capture-analyzer running (e.g. Wireshark).
   Before performing tests enable debug mode for the NAT in NAT router:

   ```
   RouterNAT # debug ip nat
   ```

   and ICMP datagrams tracking on the router R2 in outside network:

   ```
   Router # debug ip icmp
   ```

   The test can be performed using ping command – from internal computer to router R2 in outside network.

   Compare IP addresses of both sender and recipient in captured communication – observing it in the computer (Wireshark captured traffic) and a router R2 in outside network (debug mode messages).

7. TCP test:
   Start any service that utilizes TCP protocol (e.g. Telnet and HTTP server) on router R2 in outside network:

   ```
   R2 (config) #ip http server
   R2 (config) #line vty 0 4
   R2 (config-line) #password sieci
   R2 (config-line) #login
   ```

   In addition, enable TCP tracing events:

   ```
   R2 # debug ip tcp transactions
   ```

   Then connect to the TCP server from computer inside using respectively a web browser (HTTP) and network client such as Putty (TELNET).

   Compare again IP addresses of the sender and recipient observed in a computer (Wireshark) and router R2 (debug mode).

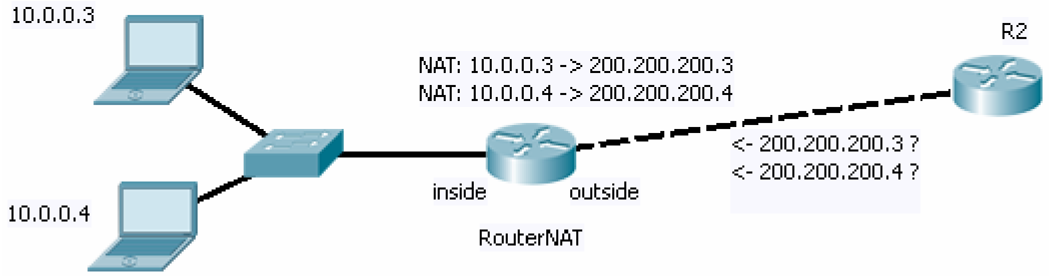
8. During tests check NAT translations on NAT router (translations are quickly outdated):
   RouterNAT # show ip nat translations
   RouterNAT # show ip nat statistics

Task B: NAT static translations, DMZ

1. Static translations are used to denote a specific host on inside network and make it available from outside. Translation will bind one particular host IP inside with public interface of the on the outside network. Static translation (host-to-host) can be introduced at the same time much. Direct translation of a particular IP address on any other gives considerable range of possibilities - led by lot of flexibility instead of IP addresses.
   In the current experiment, the translation will be carried out as a static address substitution of 10.10.10.3 to 200.200.200.3 and 10.10.10.4 to 200.200.200.4 (200.200.200.3 and 200.200.200.4 are free public addresses).
   Consequently, when the packet coming from inside with source IP address of 10.0.0.3 it will be sent to the network outside network with source address of 200.200.200.3. Similarly, for the 10.0.0.4 (conversion to 200.200.200.4). It will also be able to communicate in the opposite direction with translation backwards. So when connection from outside will be made to 200.200.200.3 or 200.200.200.4 it will be redirected respectively to 10.0.0.3 and 10.0.0.4 inside (after NAT conversion). This mechanism is the basis for DMZ zones (hidden behind a NAT but available from outside with a use of special IP address). It could also be just a router public outside address.

2. Define the rules for static NAT conversion for selected hosts, e.g.:

   RouterNAT (config) # ip nat inside source static 10.0.0.3 200.200.200.3
   RouterNAT (config) # ip nat inside source static 10.0.0.4 200.200.200.4



   Addresses 200.200.200.3 and 200.200.200.4 will now became “virtual” s the router will be answering ARP broadcast question for it as well (replaying with vesy same MAC for each of these addresses) In the end, the NAT router public interface will implement multiple IP addresses (200.200.200.1, 200.200.200.3, 200.200.200.4).

   Please note that the static IP address translation is bidirectional (datagrams returning will be translated in reverse).

Configure the IP addresses inside in the computers as 10.0.0.3 or 10.0.0.4. Set its default gateways IP addresses to point 10.0.0.1 (NAT router interface). Then try to connect from these ccomputers (ping) to router R2 (in outside network, 200.200.200.2). Observe in R2 (debug ip icmp) who is a sender of ICMP datagrams received.
Then do a ping from the router R2 to 200.200.200.3 and 200.200.200.4. These datagrams should be redirected to 10.0.0.3 or 10.0.0.4 (in one of the computers verify that using Wireshark).

Note: In a consequence of the DMZ, host 10.0.0.3 or 10.0.0.4 is now in the public network under a fictitious addresses of 200.200.200.3, 200.200.200.4, creating a DMZ (accessible from outside the network). That could cause potential hazardous situations there.

It is even possible to create a static translation of virtual addresses from networks is not directly connected to the NAT router or non-existent, for example:

RouterNAT (config) #no ip nat inside source static 10.0.0.4 200.200.200.4
RouterNAT (config) # ip nat inside source static 10.0.0.4 200.200.201.4

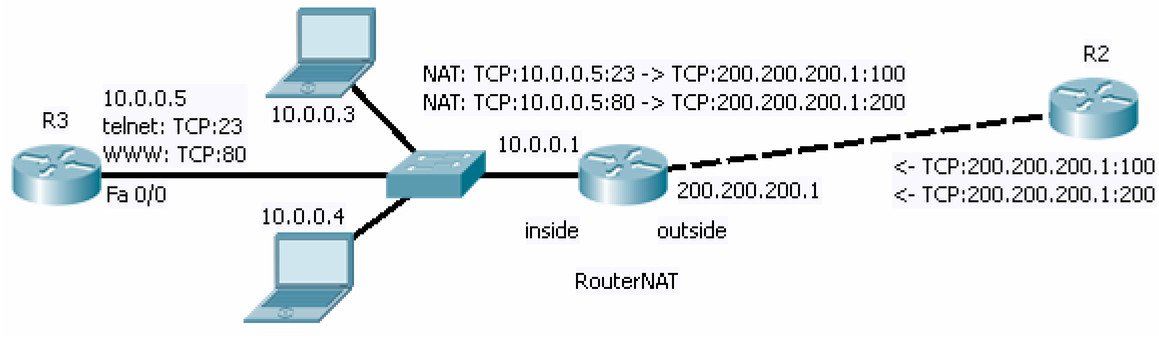
Before moving on to the next exercise remove static routing rules defined:

RouterNAT (config) #no ip nat inside source static 10.0.0.3 200.200.200.3
RouterNAT (config) #no ip nat inside source static 10.0.0.4 200.200.201.4

Task C: NAT translation with static port TCP or UDP forwarding (NAPT)

1. It is possible to access specified services only hosted in private network (inside) from outside. Some hosts in such case, located in the public network, will attempt to communicate with NAT Router through its outside interface using a specific TCP or UDP port. Datagrams sent to NAT router will be converted and send inside with a redirection as before, but additionally a port numver will be taken under consideration during process. Port number can also be converter into some new one.

   Add another router to inside network by placing and define its IP addressing.



Start some test services on that router, such as telnet and HTTP:

```
R3 (config) #int fa 0/0
R3 (config-if) #ip address 10.0.0.5 255.255.255.0
R3 (config-if) #no sh
R3 (config) #ip http server
R3 (config) #line vty 0 4
R3 (config-line) #password pass
R3 (config-line) #login
R3 (config-line) #transport input telnet
```

Due to the fact that the router R3 will act as a host inside with the ability of sending datagrams outside - it is necessary to give him the default routing rules pointing outside infrastructure (the address of the NAT router):

```
R3 (config) #ip route 0.0.0.0 0.0.0.0 10.0.0.1
```

2. Define the rules for static IP address and TCP ports translation in the NAT router:

```
RouterNAT (config) #
                    ip nat inside source static tcp 10.0.0.5 23  200.200.200.1 100
RouterNAT (config) #
                    ip nat inside source static tcp 10.0.0.5 80 200.200.200.1 200
```

Right now TCP traffic from outside sent to outside ports of 100 and 200 will be redirected to host 10.0.0.5 and ports of 23 and 80 respectively.

Note: Due to the bidirectional static IP NAT rules nature, traffic coming in an opposite direction will also be converter back.

3. In the outside router (simulating external client) connect some services on NAT router (ports 100 and 200 TCP) with a build-in TELNET client:

```
R2#telnet 200.200.200.1 100
R2#telnet 200.200.200.1 200
```

Also perform some diagnostics during these test:

```
RouterNAT # debug ip nat
RouterNAT # sh ip nat translatios
R3 # debug ip tcp transactions
```