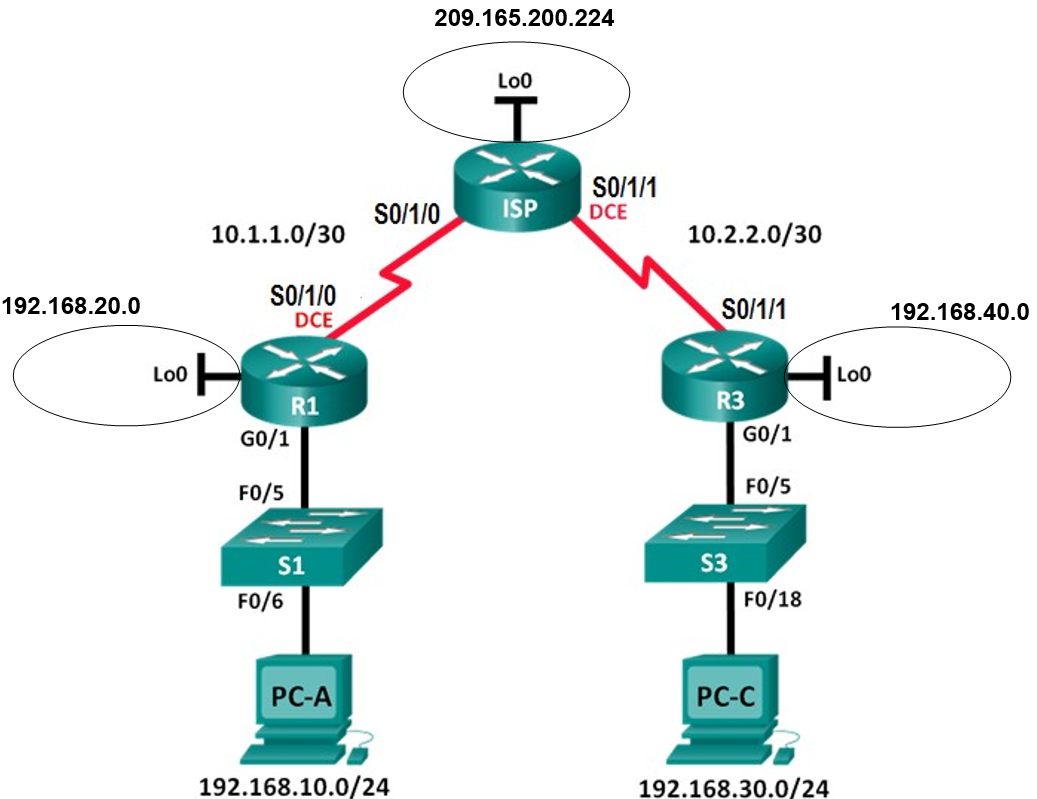


**Lab 7.2.2.6– Configuring and Verifying Standard IPv4 ACLs**

**Topology**

****

**Addressing Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Device** | **Interface** | **IP Address** | **Subnet Mask** | **Default Gateway** |
| R1 | G0/1 | 192.168.10.1 | 255.255.255.0 | N/A |
|  | Lo0 | 192.168.20.1 | 255.255.255.0 | N/A |
|  | S0/1/0 (DCE) | 10.1.1.1 | 255.255.255.252 | N/A |
| ISP | S0/1/0 | 10.1.1.2 | 255.255.255.252 | N/A |
|  | S0/1/1 (DCE) | 10.2.2.2 | 255.255.255.252 | N/A |
|  | Lo0 | 209.165.200.225 | 255.255.255.224 | N/A |
| R3 | G0/1 | 192.168.30.1 | 255.255.255.0 | N/A |
|  | Lo0 | 192.168.40.1 | 255.255.255.0 | N/A |
|  | S0/1/1 | 10.2.2.1 | 255.255.255.252 | N/A |
| S1 | VLAN 1 | 192.168.10.11 | 255.255.255.0 | 192.168.10.1 |
| S3 | VLAN 1 | 192.168.30.11 | 255.255.255.0 | 192.168.30.1 |
| PC-A | NIC | 192.168.10.3 | 255.255.255.0 | 192.168.10.1 |
| PC-C | NIC | 192.168.30.3 | 255.255.255.0 | 192.168.30.1 |

**Objectives**

### Part 1: Configure Devices and Verify Connectivity

* Assign a static IP address to PCs.
* Configure basic settings on routers.
* Configure basic settings on switches.
* Configure RIP v2 routing on R1, ISP, and R3.
* Verify connectivity between devices.

### Part 2: Configure and Verify Standard Numbered and Named ACLs

* Configure, apply, and verify a numbered standard ACL.
* Configure, apply, and verify a named ACL.

### Part 3: Modify a Standard ACL

* Modify and verify a named standard ACL.
* Test the ACL.

## Background / Scenario

Network security is an important issue when designing and managing IP networks. The ability to configure proper rules to filter packets, based on established security policies, is a valuable skill.

In this lab, you will set up filtering rules for two offices represented by R1 and R3. Management has established some access policies between the LANs located at R1 and R3, which you must implement. The ISP router sitting between R1 and R3 will not have any ACLs placed on it. You would not be allowed any administrative access to an ISP router because you can only control and manage your own equipment.

1. Configure Devices and Verify Connectivity
   1. Configure IP addresses on PC-A and PC-C.
   2. Configure basic settings for the routers.
      1. Console into the router and enter global configuration mode.
      2. Copy the following basic configuration and paste it to the running-configuration on the router.

no ip domain-lookup

hostname R1

service password-encryption

enable secret class

banner motd #

Unauthorized access is strictly prohibited. #

Line con 0

password cisco

login

logging synchronous

line vty 0 4

password cisco

login

* + 1. Configure the device name as shown in the topology.
    2. Create loopback interfaces on each router as shown in the Addressing Table.
    3. Configure interface IP addresses as shown in the Topology and Addressing Table.
    4. Assign a clock rate of **128000** to the DCE serial interfaces.
  1. Configure basic settings on the switches.
     1. Console into the switch and enter global configuration mode.
     2. Copy the following basic configuration and paste it to the running-configuration on the switch.

no ip domain-lookup

service password-encryption

enable secret class

banner motd #

Unauthorized access is strictly prohibited. #

Line con 0

password cisco

login

logging synchronous

line vty 0 15

password cisco

login

exit

* + 1. Configure the device name as shown in the topology.
    2. Configure the management interface IP address as shown in the Topology and Addressing Table.
    3. Configure a default gateway.
  1. Configure Rip routing on R1, ISP, and R3.
     1. Configure **RIP version** 2 and advertise all networks on **R1**, **ISP**, and **R3**. The RIP configuration for R1 and ISP is included for reference.

R1(config)# **router rip**

R1(config-router)# **version 2**

R1(config-router)# **network 192.168.10.0**

R1(config-router)# **network 192.168.20.0**

R1(config-router)# **network 10.1.1.0**

ISP(config)# **router rip**

ISP(config-router)# **version 2**

ISP(config-router)# **network 209.165.200.224**

ISP(config-router)# **network 10.1.1.0**

ISP(config-router)# **network 10.2.2.0**

R3(config)#

R1(config-router)#

R3(config-router)#

R3(config-router)#

R3(config-router)#

* + 1. After configuring Rip on R1, ISP, and R3, verify that all routers have complete routing tables, listing all networks. Troubleshoot if this is not the case.
  1. Verify connectivity between devices.

**Note**: It is very important to test whether connectivity is working **before** you configure and apply access lists! You want to ensure that your network is properly functioning before you start to filter traffic.

* + 1. From PC-A, ping PC-C and the loopback interface on R3. Were your pings successful? \_\_\_\_\_\_\_
    2. From R1, ping PC-C and the loopback interface on R3. Were your pings successful? \_\_\_\_\_\_\_
    3. From PC-C, ping PC-A and the loopback interface on R1. Were your pings successful? \_\_\_\_\_\_\_
    4. From R3, ping PC-A and the loopback interface on R1. Were your pings successful? \_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| **From** | **To** | **IP Address (To)** | **Ping results**  (Fail / Success) |
| **PC-A** | **PC-C** | **192.168.30.3** |  |
| **PC-A** | **R3 Lo0** | **192.168.40.1** |  |
| **R1** | **PC-C** | **192.168.30.3** |  |
| **R1** | **R3 Lo0** | **192.168.40.1** |  |
| **PC-C** | **PC-A** | **192.168.10.3** |  |
| **PC-C** | **R1 Lo0** | **192.168.20.1** |  |
| **R3** | **PC-A** | **192.168.10.3** |  |
| **R3** | **R1 Lo0** | **192.168.20.1** |  |

1. Configure and Verify Standard Numbered and Named ACLs
   1. Configure a numbered standard ACL.

Standard ACLs filter traffic based on the source IP address only. A typical best practice for standard ACLs is to configure and apply it as close to the destination as possible. For the first access list, create a standard numbered ACL that allows traffic from all hosts on the **192.168.10.0/24** network and all hosts on the **192.168.20.0/24** network to access all hosts on the **192.168.30.0/24** network. The security policy also states that a **deny any** access control entry (ACE), also referred to as an ACL statement, should be present at the end of all ACLs.

What wildcard mask would you use to allow all hosts on the **192.168.10.0/24** network to access the **192.168.30.0/24** network? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Following Cisco’s recommended best practices, on which router would you place this ACL? \_\_\_\_\_\_\_\_\_\_\_

On which interface would you place this ACL? In what direction would you apply it?\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + 1. Configure the ACL on R3. Use 1 for the access list number.

R3(config)# **access-list 1 remark Allow R1 LANs Access**

R3(config)# **access-list 1 permit 192.168.10.0 0.0.0.255**

R3(config)# **access-list 1 permit 192.168.20.0 0.0.0.255**

R3(config)# **access-list 1 deny any**

* + 1. Apply the ACL to the appropriate interface in the proper direction.

R3(config)# **interface g0/1**

R3(config-if)# **ip access-group 1 out**

* + 1. Verify a numbered ACL.

The use of various **show** commands can aid you in verifying both the syntax and placement of your ACLs in your router.

To see **access list 1** in its entirety with all ACEs, which command would you use?\_\_\_ **show** **access-list 1**

What command would you use to see where the access list was applied and in what direction?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **show ip interface g0/1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + - 1. On R3, issue the **show access-lists 1** command.

R3# **show** **access-list 1**

Standard IP access list 1

10 permit 192.168.10.0, wildcard bits 0.0.0.255

20 permit 192.168.20.0, wildcard bits 0.0.0.255

30 deny any

* + - 1. On R3, issue the **show ip interface g0/1** command.

R3# **show ip interface g0/1**

GigabitEthernet0/1 is up, line protocol is up

Internet address is 192.168.30.1/24

Broadcast address is 255.255.255.255

Address determined by non-volatile memory

MTU is 1500 bytes

Helper address is not set

Directed broadcast forwarding is disabled

Multicast reserved groups joined: 224.0.0.10

Outgoing access list is 1

Inbound access list is not set

Output omitted

* + - 1. Test the ACL to see if it allows traffic from the **192.168.10.0/24** network access to the **192.168.30.0/24** network. From the **PC-A** command prompt, ping the **PC-C** IP address. Were the pings successful? \_\_\_\_\_\_\_
      2. Test the ACL to see if it allows traffic from the 192.168.20.0/24 network access to the **192.168.30.0/24** network. You must do an extended ping and use the loopback 0 address on R1 as your source. Ping PC-C’s IP address. Were the pings successful? \_\_\_\_\_\_\_

R1# ping

Protocol [ip]:

Target IP address: **192.168.30.3**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **192.168.20.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.30.3, timeout is 2 seconds:

Packet sent with a source address of 192.168.20.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/29/32 ms

* + 1. From the **R1** prompt, ping **PC-C’s** IP address again.

R1# **ping 192.168.30.3**

Was the ping successful? Why or why not?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Configure a named standard ACL.

Create a named standard ACL that conforms to the following policy: **allow** traffic from all hosts on the **192.168.40.0/24** network access to all hosts on the **192.168.10.0/24** network. Also, only **allow** host **PC-C** access to the **192.168.10.0/24** network. The name of this access list should be called **BRANCH-OFFICE-POLICY**.

Following Cisco’s recommended best practices, on which router would you place this ACL? \_\_\_\_\_\_\_\_R1\_\_\_

On which interface would you place this ACL? In what direction would you apply it?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_g0/1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + 1. Create the standard named ACL **BRANCH-OFFICE-POLICY** on R1.

R1(config)# **ip access-list standard BRANCH-OFFICE-POLICY**

R1(config-std-nacl)# **permit host 192.168.30.3**

**192.168.30.3 0.0.0.0**

R1(config-std-nacl)# **permit 192.168.40.0 0.0.0.255**

R1(config-std-nacl)# **end**

R1#

\*Feb 15 15:56:55.707: %SYS-5-CONFIG\_I: Configured from console by console

Looking at the first permit ACE in the access list, what is another way to write this?

R1(config-std-nacl)# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + 1. Apply the ACL to the appropriate interface in the proper direction.

R1# **config t**

R1(config)# **interface g0/1**

R1(config-if)# **ip access-group BRANCH-OFFICE-POLICY out**

* + 1. Verify a named ACL.
       1. On R1, issue the **show access-lists** command.

R1# **show** **access-lists**

Standard IP access list BRANCH-OFFICE-POLICY

10 permit 192.168.30.3

20 permit 192.168.40.0, wildcard bits 0.0.0.255

Is there any difference between this ACL on R1 with the ACL on R3? If so, what is it?

\_\_\_\_\_\_\_\_\_\_\_La lista de acceso nombrada no tiene el deny any al final. Es implícito. Pero una Buena práctica es configurarlo.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + - 1. On R1, issue the **show ip interface g0/1** command.

R1# **show ip interface g0/1**

GigabitEthernet0/1 is up, line protocol is up

Internet address is 192.168.10.1/24

Broadcast address is 255.255.255.255

Address determined by non-volatile memory

MTU is 1500 bytes

Helper address is not set

Directed broadcast forwarding is disabled

Multicast reserved groups joined: 224.0.0.10

Outgoing access list is BRANCH-OFFICE-POLICY

Inbound access list is not set

<Output omitted>

* + - 1. Test the ACL. From the command prompt on **PC-C**, ping **PC-A’s** IP address. Were the pings successful? \_\_\_\_\_\_\_
      2. Test the ACL to ensure that only the **PC-C** host is allowed access to the **192.168.10.0/24** network. You must do an extended ping and use the **G0/1** address on **R3** as your source. Ping **PC-A’s** IP address. Were the pings successful? \_\_\_\_\_\_\_

R3# ping

Protocol [ip]:

Target IP address: **192.168.10.3**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **192.168.30.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.10.3, timeout is 2 seconds:

Packet sent with a source address of 192.168.30.1

U.U.U

* + - 1. Test the ACL to see if it allows traffic from the **192.168.40.0/24** network access to the **192.168.10.0/24** network. You must perform an extended ping and use the **loopback 0** address on **R3** as your source. Ping **PC-A’s** IP address. Were the pings successful? \_\_\_\_\_\_\_

R3# ping

Protocol [ip]:

Target IP address: **192.168.10.3**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **192.168.40.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.10.3, timeout is 2 seconds:

Packet sent with a source address of 192.168.40.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 32/40/60 ms

1. Modify a Standard ACL

It is common in business for security policies to change. For this reason, ACLs may need to be modified. In Part 3, you will change one of the previous ACLs you configured to match a new management policy being put in place.

Management has decided that users from the **209.165.200.224/27** network should be **allowed** full access to the **192.168.10.0/24** network. Management also wants ACLs on all of their routers to follow consistent rules. A **deny any** ACE should be placed at the end of all ACLs. You must modify the **BRANCH-OFFICE-POLICY** ACL.

You will add two additional lines to this ACL. There are two ways you could do this:

**OPTION 1:** Issue a **no ip access-list standard BRANCH-OFFICE-POLICY** command in global configuration mode. This would effectively take the whole ACL out of the router. Depending upon the router IOS, one of the following scenarios would occur: all filtering of packets would be cancelled and all packets would be allowed through the router; or, because you did not take off the **ip access-group** command on the G0/1 interface, filtering is still in place. Regardless, when the ACL is gone, you could retype the whole ACL, or cut and paste it in from a text editor.

**OPTION 2:** You can modify ACLs in place by adding or deleting specific lines within the ACL itself. This can come in handy, especially with ACLs that have many lines of code. The retyping of the whole ACL or cutting and pasting can easily lead to errors. Modifying specific lines within the ACL is easily accomplished.

**Note**: For this lab, use Option 2.

* 1. Modify a named standard ACL.
     1. From R1 privileged EXEC mode, issue a **show access-lists** command.

R1# **show access-lists**

Standard IP access list BRANCH-OFFICE-POLICY

10 permit 192.168.30.3 (8 matches)

20 permit 192.168.40.0, wildcard bits 0.0.0.255 (5 matches)

* + 1. Add two additional lines at the end of the ACL. From global config mode, modify the ACL, BRANCH-OFFICE-POLICY.

R1#(config)# **ip access-list standard BRANCH-OFFICE-POLICY**

R1(config-std-nacl)# **30 permit 209.165.200.224 0.0.0.31**

R1(config-std-nacl)# **40 deny any**

R1(config-std-nacl)# **end**

* + 1. Verify the ACL.
       1. On R1, issue the **show access-lists** command.

R1# **show access-lists**

Standard IP access list BRANCH-OFFICE-POLICY

10 permit 192.168.30.3 (8 matches)

20 permit 192.168.40.0, wildcard bits 0.0.0.255 (5 matches)

30 permit 209.165.200.224, wildcard bits 0.0.0.31

40 deny any

Do you have to apply the **BRANCH-OFFICE-POLICY** to the **G0/1** interface on **R1**?

\_\_\_\_\_\_\_\_\_No hay que aplicarla, ya está aplicada la lista de acceso.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* + - 1. From the ISP command prompt, issue an extended ping. Test the ACL to see if it allows traffic from the **209.165.200.224/27** network access to the **192.168.10.0/24** network. You must do an extended ping and use the **loopback 0** address on **ISP** as your source. Ping PC-A’s IP address. Were the pings successful? \_\_\_\_\_\_\_

Reflection

* 1. As you can see, standard ACLs are very powerful and work quite well. Why would you ever have the need for using extended ACLs?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Typically, more typing is required when using a named ACL as opposed to a numbered ACL. Why would you choose named ACLs over numbered?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_