

Computing Networks

Laboratory No. 7

**Basic Infrastructure
and Network Layer**

Students:

Andrea Camila Torres González
Jorge Andrés Gamboa Sierra

Presented to:

Fabian Eduardo Sierra Sánchez
Semester 2024-2

Content

Objective	3
Abstract.....	7
Tools to be Used.....	7
Introduction	7
Theoretical framework	8
Linux	9
PowerShell	11
Base Software Installation.....	12
1. Other Useful Commands	12
a. Network Information Shell	12
b. Open and running ports	18
2. Network Management	19
2.1. Solaris	19
2.2. Linux Slackware	40
3. Network Administration – Azure	59
3.1. Configuration of the web application service	59
3.2. Questionnaire	67
Experiments.....	69
1. Use of ICMP Messages.....	69
2. Some Questions About Router Commands.....	80
3. Setup: Access and Basic Configuration of Routers	82
4. Setup - Serial Interconnection.....	92
5. Static Routing.....	96
6. Closure	96
Conclusions	97
References.....	98

Figure 1. First part of the shell script for network information	13
Figure 2. Second part of the shell script for network information.....	14
Figure 3. Testing the first option of the script.....	15
Figure 4. Testing second option of the script	15
Figure 5. Testing third option of the script.....	16
Figure 6. Testing fourth option of the script	16
Figure 7. Testing fifth option with TCP	17
Figure 8. Testing fifth option with UDP	17
Figure 9. shell script for open and running ports	18
Figure 10. Testing port 80 in the second shell script.....	18
Figure 11. Testing unknown ports in the second shell script	18
Figure 12. Installing gd on Solaris	19
Figure 13. Installing gcc on Solaris.....	19
Figure 14. Downloading nagios.tar.gz file	20
Figure 15. Decompressing .tar.gz file	20
Figure 16. Moving nagios-4.5.7 into Nagios directory	20
Figure 17. Configuring Nagios	21
Figure 18. Opening worker-ping.c file	21
Figure 19. Changing worker-ping.c file.....	22
Figure 20. Opening utils.c file	22
Figure 21. Changing utils.c file.....	23
Figure 22. Opening /cgi/Makefile file.....	23
Figure 23. Changing /cgi/Makefile file.....	24
Figure 24. Compile Nagios command.....	24
Figure 25. Compiling Nagios	25
Figure 26. Creating Nagios user.....	25
Figure 27. Installing Nagios using 'gmake install' command.....	26
Figure 28. Finishing the Nagios configuration with the command 'gmake install-commandmode'	26
Figure 29. Finishing the Nagios configuration with the command 'gmake install-config'	27
Figure 30. Opening cgi.cfg file	27
Figure 31. Configuring cgi.cfg file	28
Figure 32. Downloading nagios plugins.....	28
Figure 33. Decompressing nagios-plugins	29
Figure 34. Configuring nagios plugins	29
Figure 35. Using 'gmake install' to install nagios plugins	29
Figure 36. Installing nagios plugins.....	30
Figure 37. Hosts monitoring configuration	31
Figure 38. Nagios services configuration.....	32
Figure 39. Command to verify the Nagios configuration	32
Figure 40. Verifying Nagios configuration	33
Figure 41. Creating a password for the user nagiosadmin.....	33
Figure 42. Opening snmpd.conf	33
Figure 43. Configuring SNMP.....	34
Figure 44. Adding additional information on snmpd.conf	35
Figure 45. Testing SNMP service on Solaris.....	36
Figure 46. Starting SNMP service on Solaris.....	36

Figure 47. Apache file configuration	37
Figure 48. Initializing Nagios, SNMP and Apache	37
Figure 49. Accessing the Nagios web interface	38
Figure 50. Nagios web interface	38
Figure 51. Solaris Host log information into Nagios	39
Figure 52. Solaris Host Graphic	39
Figure 53. Solaris Host information into Nagios	40
Figure 54. Command to install SNMP	40
Figure 55. Selecting SNMP package	41
Figure 56. SNMP installation	42
Figure 57. Opening snmpd.conf file	42
Figure 58. Configuring SNMP	43
Figure 59. SNMP starting failed	43
Figure 60. Installing libnsl	44
Figure 61. Installing libnl	44
Figure 62. Installing libsensors	45
Figure 63. Installing pciutils	45
Figure 64. Starting SNMP on Slackware	45
Figure 65. Testing SNMP on slackware	46
Figure 66. Verifying port 161 is open	46
Figure 67. Downloading zabbix.tar.gz file	47
Figure 68. Command to decompress the .tar.gz file	47
Figure 69. Decompressing zabbix.tar.gz file	47
Figure 70. Configuring user for Zabbix	48
Figure 71. Creating database for Zabbix	48
Figure 72. Configure Zabbix command	48
Figure 73. Configuring Zabbix	49
Figure 74. Compiling Zabbix	50
Figure 75. Installing Zabbix	51
Figure 76. Zabbix SQL files	51
Figure 77. Moving the schema.sql file into the database	51
Figure 78. Modification of the Zabbix database after inserting schema.sql	52
Figure 79. Moving the images.sql file into the database	52
Figure 80. Modification of the Zabbix database after inserting images.sql	53
Figure 81. Moving the data.sql file into the database	53
Figure 82. Modification of the Zabbix database after inserting data.sql	54
Figure 83. Opening zabbix_server.conf	54
Figure 84. Configuring database and username into zabbix_server.conf	55
Figure 85. Assigning a password for the Zabbix database	56
Figure 86. Opening zabbix_agentd.conf	56
Figure 87. configuring Slackware machine that Zabbix will monitor	57
Figure 88. Executing a query on the Zabbix database	58
Figure 89. Searching "Education" Section	59
Figure 90. Navigating to "Templates" Section	59
Figure 91. Selecting "Web App Deployment from GitHub" template	60
Figure 92. Deploying web service	60

Figure 93. Verifying deployment	61
Figure 94. Finalizing deployment.....	62
Figure 95. Navigating to Web App resource.....	62
Figure 96. Exploring the web application	62
Figure 97. Verifying that Web Service is working.....	63
Figure 98. Navigating to Application Insights	63
Figure 99. Activating the "Application Insights" service.....	64
Figure 100. Enabling the "Application Insights" service	64
Figure 101. Verifying configuration of "Application Insights"	65
Figure 102. Navigating to Application Insights data.....	66
Figure 103. Overview of Application Insights	66
Figure 104. Real-time view of the performance of the web service application	67
Figure 105. Metrics after refreshing the website repeatedly	68
Figure 106 traceroute-online page.....	70
Figure 107 Search for the laboratory page.....	70
Figure 108 Open Visual Traceroute page	74
Figure 109 Completion of the program installation	75
Figure 110 Initial interface of Open Visual Traceroute	75
Figure 111 Route from Bogotá to Australia.....	76
Figure 112 Packet route to Australia	76
Figure 113 Route from Bogotá to Costa Rica	77
Figure 114 Packet route to Costa Rica.....	77
Figure 115 Route from Bogotá to the United States.....	78
Figure 116 Packet route to United States	78
Figure 117 Route from Bogotá to Hong Kong	79
Figure 118 Packet route to Honk Kong	79
Figure 119 Route from Bogotá to Hong Kong	80
Figure 120 Packet route to Finland	80
Figure 121 RS-232 serial cable extension	83
Figure 122 PuTTY configuration	83
Figure 123 initial router configuration	84
Figure 124 basic configuration	85
Figure 125 subnetting configuration	86
Figure 126 connection of devices.....	87
Figure 127 connection between devices and router.....	88
Figure 128 connection between devices and router.....	88
Figure 129 connection to console cable.....	89
Figure 130 configuration interface Fa0/	89
Figure 131 configuration interface fa0/1	89
Figure 132 Verification of the configuration of the interfaces.....	90
Figure 133 Configuration of IP and gateway on one of the devices.....	90
Figure 134 Configuration of IP and gateway on one of the devices.....	91
Figure 135 Disabling the firewall	91
Figure 136 test to interface 88.0.4.1	92
Figure 137 test to device with Ip 88.0.4.2	92
Figure 138 test to device with Ip 88.0.16.0.2	92

Figure 139 connection between devices	94
Figure 140 connection between devices an router.....	94
Figure 141 serial connection	95
Figure 142 configuration interface s0/0/1	95
Figure 143 verification connection interfaces.....	96
Figure 144 test connection between routers.....	96
Figure 145 static routes router 8.....	96
Figure 146 static routes router 9.....	¡Error! Marcador no definido.
Figure 147 trace route between devices.....	96

Objective

Continue learning about the operation of operating systems and network services.

Install network management tools.

Configure routers and static routing.

Abstract

In this lab, we will deepen our understanding of operating systems and network services. We will install network management tools, configure routers, and set up static routes. We will create shell scripts for network diagnostics and configure monitoring servers on different operating systems. Using Azure, we will deploy and monitor web applications. Practical activities include tracing routes with ICMP, configuring routers, and setting up serial interconnections, gaining skills in network management and configuration. Finally, we will demonstrate our configurations to the professor to ensure everything is functioning correctly.

Key words: monitoring, configuration, networking, router

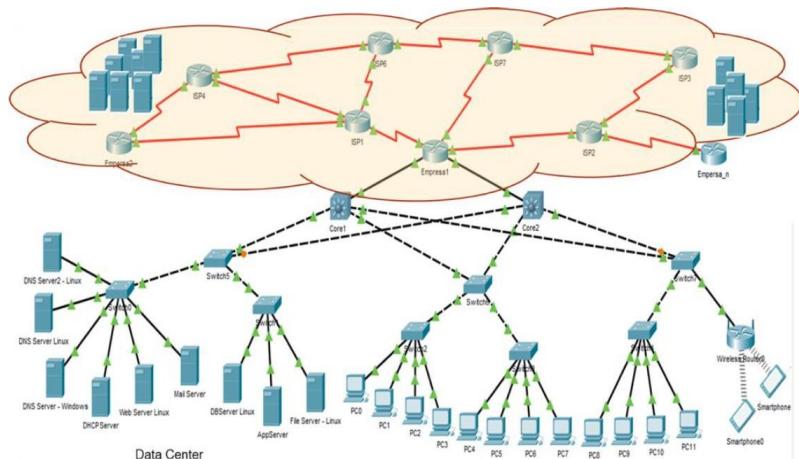
Tools to be Used

- Computers
- Internet access
- Virtualization software
- Packet Tracer and Wireshark
- Routers and switches

Introduction

A company typically has several IT infrastructure services. It includes wired and wireless user stations and servers (both physical and virtualized), all connected through switches (layer 2 and 3), wireless devices, and routers that connect it to the Internet. It is also common to have cloud infrastructures where resources are provisioned according to the organization's needs. Among the servers, there can be web services, DNS, email, database, storage, applications, and others.

The following is a possible configuration:



Theoretical framework

IP Protocol (Internet Protocol) and IP Addressing

The IP Protocol is the backbone of the network layer in the OSI model, responsible for addressing and routing data packets from their origin to their final destination. There are two predominant versions: IPv4, which uses 32-bit addresses, and IPv6, which uses 128-bit addresses to allow for a greater number of unique addresses. IP addressing assigns unique addresses to each device on a network, enabling efficient segmentation and management of the network. Proper use of subnets and subnet masks is essential for organizing and optimizing network communication.

ICMP (Internet Control Message Protocol), Traceroute, and Online Routing Tools

The ICMP Protocol is used to send error messages and operational information about an IP network. It is crucial for network diagnostic tools like ping and traceroute. Traceroute tracks the path that data packets take from the source to the destination, providing information about each intermediate hop, useful for identifying network problems and verifying routing functionality. Online routing tools allow these diagnostics to be performed via web interfaces, facilitating analysis from any device with Internet access. These tools are valuable for monitoring and troubleshooting connectivity and performance issues in both local and remote networks.

Routing and Static Routes

Routing is the process of selecting paths in a network through which to send data. Routers use routing tables to determine the best route for each packet. Routes can be static, configured manually, or dynamic, learned automatically through routing protocols such as OSPF (Open Shortest Path First) or BGP (Border Gateway Protocol). Static routes are manually configured and are ideal for small networks or specific connections where the route does not change frequently.

NAT (Network Address Translation) and VLANs (Virtual Local Area Networks)

NAT is a technique that maps private IP addresses to public IP addresses, allowing multiple devices on a local network to share a single public IP address. This conserves IP addresses and improves security by hiding the internal network structure from external devices. VLANs allow a physical network to be segmented into multiple logical networks, improving traffic management and security. Each VLAN behaves as an independent network, even though devices share the same physical hardware. This facilitates the management of large networks and improves performance by limiting the amount of traffic in each segment.

SNMP (Simple Network Management Protocol)

SNMP is a protocol used for monitoring and managing devices on a network. Network administrators use SNMP to collect information about the performance, availability, and status of network devices such as routers, switches, and servers. SNMP allows communication between devices and a management station, facilitating the automation of network administration tasks. It

uses a hierarchical structure of objects called MIB (Management Information Base) to organize the data that can be queried and controlled.

Nagios and Zabbix: Network Monitoring Tools

Nagios is a network and system monitoring tool that enables IT administrators to identify and resolve infrastructure issues before they affect critical business processes. Nagios monitors the activity of servers, switches, applications, and network services, alerting administrators to any problems detected. Nagios' ability to integrate with other systems and monitoring tools makes it a versatile and powerful solution for maintaining network stability and performance.

Zabbix is another network and system monitoring tool that provides a comprehensive solution for monitoring the performance and availability of IT resources. Zabbix allows for real-time metric collection, historical data analysis, and alert generation based on user-defined thresholds. Additionally, Zabbix supports a wide range of devices and protocols, facilitating integration with heterogeneous infrastructures. Zabbix's web interface provides a centralized view of the network's status, enabling administrators to make informed and proactive decisions to maintain operational efficiency.

CDN Networks (Content Delivery Network)

CDNs are distributed networks of servers that work together to deliver Internet content quickly and efficiently to users. CDNs cache copies of content at various geographically dispersed points, reducing latency and improving access speed. They are essential for enhancing user experience in web applications and streaming services by reducing load times and increasing reliability. CDNs also help mitigate DDoS (Denial of Service) attacks by distributing traffic evenly and ensuring greater content availability.

Basic information commands on the network

Linux

1. **ifconfig:** Displays and configures network interfaces.

-
- Parameters:
 - **-a:** Displays all interfaces, including inactive ones.
 - **<interface>:** Shows configuration of a specific interface (e.g., eth0 or lo).
 - **up:** Activates the network interface.
 - **down:** Deactivates the network interface.
 - **netmask <mask>:** Sets the subnet mask for the interface.
 - **broadcast <address>:** Sets the broadcast address for the interface.
 - **inet <IP>:** Assigns an IP address to the interface.
 - **mtu <size>:** Sets the maximum transmission unit (MTU) size.
-
-

2. **Netstat:** Displays network statistics and connections.

UNIVERSIDAD

- Parameters:
 - **-i**: Displays information about network interfaces.
 - **-r**: Displays the routing table.
 - **-a**: Displays all active connections and listening ports.
 - **-n**: Displays IP addresses and ports in numeric format (without resolving names).
 - **-t**: Displays only TCP connections.
 - **-u**: Displays only UDP connections.
 - **-p**: Displays the name of the process using the connection.
- - **route**: Displays and manipulates the routing table.
- Parameters:
 - **add**: Adds a new route.
 - **delete**: Deletes an existing route.
 - **default**: Sets a default route.
 - **-n**: Displays the routing table in numeric format (without resolving names).
 - **-v**: Displays the routing table with more detail (verbose).
- **ping**: Verifies network connectivity using ICMP packets.
 - Parameters:
 - **-c <n>**: Specifies the number of ICMP packets to send.
 - **-s <size>**: Sets the size of the ICMP packets.
 - **-i <interval>**: Sets the interval between ICMP packets.
 - **-t <ttl>**: Sets the TTL (Time to Live) value for the packets.
 - **-w <timeout>**: Sets a timeout in seconds to wait for a response.
- **traceroute**: Displays the route that packets take to reach a destination.
 - Parameters:
 - **-n**: Displays the hop addresses in numeric format (without resolving names).
 - **-m <hops>**: Limits the number of hops.
 - **-q <queries>**: Sets the number of queries per hop.
- **nslookup**: Performs DNS queries.
 - Parameters:
 - **<domain>**: Queries a specific domain.
 - **server <dns-server>**: Specifies a DNS server to query.
 - **set type=<type>**: Sets the query type (e.g., A, MX, NS).
- **ethtool**: Displays or configures parameters for Ethernet network interfaces.
 - Parameters:
 - **-i <interface>**: Displays information about the network interface, such as the driver.

- ○ -s <interface>: Configures settings for the interface (e.g., speed and duplex).
- ○ -a <interface>: Displays the auto-negotiation status of the interface.
- ○ -p <interface>: Blinks the network interface LED for identification.

8. hostname: Displays or sets the system's hostname.

- ○ Parameters:
 - ○ -f: Displays the fully qualified domain name (FQDN).
 - ○ <hostname>: Sets a new hostname.
- ○

PowerShell

1. ipconfig: Displays the current network configuration of all network interfaces.

- ○ Parameters:
 - ○ /all: Displays full information for all interfaces (including DNS and MAC addresses).
 - ○ /release: Releases the current DHCP lease.
 - ○ /renew: Renews the DHCP lease.
 - ○ /flushdns: Clears the DNS resolver cache.
 - ○ /displaydns: Displays the contents of the DNS resolver cache.
 - ○ /registerdns: Refreshes all DHCP leases and re-registers DNS names.
- ○
- ○ **2. netstat:** Displays network statistics and current active connections.
- ○
- ○ Parameters:
 - ○ -a: Displays all active connections and listening ports.
 - ○ -n: Displays the addresses and port numbers in numeric format (without resolving names).
 - ○ -o: Displays the owning process ID (PID) associated with each connection.
 - ○ -p: Displays the protocol used for each connection.
 - ○ -r: Displays the routing table.
 - ○ -s: Displays network statistics by protocol.
- ○
- ○ **3. tracert:** Traces the route packets take to a destination.
- ○
- ○ Parameters:
 - ○ -h <max_hops>: Limits the number of hops to the destination.
 - ○ -d: Prevents DNS resolution, shows IP addresses only.
 - ○ -w <timeout>: Sets the timeout in milliseconds for each reply.

4. Ping: Sends ICMP echo requests to verify network connectivity.

- ○
- ○ Parameters:
 - ○ -t: Pings the target indefinitely until stopped (Ctrl + C).
 - ○ -n <count>: Specifies the number of echo requests to send.
 - ○ -l <size>: Specifies the size of the ping packet.

- **-4:** Forces the use of IPv4.
- **-6:** Forces the use of IPv6.
-
- **5. Route:** Displays and modifies the network routing table.
-
- Parameters:
 - **print:** Displays the routing table.
 - **add:** Adds a new route.
 - **delete:** Deletes a route.
 - **change:** Modifies an existing route.
 - **-f:** Clears the routing table and deletes all routes.
-
- **6. nslookup:** Queries the DNS to resolve hostnames to IP addresses.
-
- Parameters:
 - **<hostname>:** Queries a specific domain name.
 - **server <dns_server>:** Specifies a DNS server to use for the query.
 - **set type=<record_type>:** Specifies the record type (e.g., A, MX, NS).
 - **-type=<type>:** Specifies the record type (e.g., A, MX, PTR).
-
- **7. getmac:** Displays the MAC (Media Access Control) address of the network interfaces.
-
- Parameters:
 - **/v:** Displays additional information (e.g., interface description).
 - **/fo <format>:** Specifies the output format (e.g., table, csv).
 - **/s <computer>:** Specifies a remote computer to query.
-

Base Software Installation

1. Other Useful Commands

a. Network Information Shell

Study the functionality of commands that provide network information, e.g., netstat, vnstat, route, and ethtool (or equivalents in Slackware, Solaris, Centos, and PowerShell). Review different parameters that can be used and create a Shell program that uses them.

- First, we created our shell script and designed a menu with the options shown in *Figure 1*. We used **netstat** to display network information such as ports, services, statistics, and UDP and TCP connections. We also used **route** to view the routing table and **ethtool** to check network interface information

```

GNU nano 6.0                                     netstat.sh
#!/bin/bash
while true; do
    echo "Network information menu:"
    echo "1) Show network connections and listening ports"
    echo "2) Show network usage"
    echo "3) Show routing table"
    echo "4) Show network interface details"
    echo "5) Show connections with an specific protocol"
    echo "6) Exit"
    read option
    case $option in
        1)
            echo "Showing listening ports and services:"
            netstat -tuln
            ;;
        2)
            echo "Network statistics: "
            netstat -i
            ;;
        3)
            echo "Routing table:"
            route -n
            ;;
        4)
            echo "Network interface details"
            ethtool eth1
            ;;
        5)
            echo "Select protocol:"
            echo "1. UDP"
            echo "2. TCP"
            read protocol
            case $protocol in
                ^G Help      ^O Write Out   ^W Where Is   ^K Cut       ^T Execute   ^C Location   M-U Undo
                ^X Exit      ^R Read File    ^H Replace    ^U Paste     ^J Justify   ^- Go To Line M-E Redo

```

Figure 1. First part of the shell script for network information

```

GNU nano 6.0                               netstat.sh

;;
4) echo "Network interface details"
ethtool eth1
;;
5) echo "Select protocol:"
echo "1. UDP"
echo "2. TCP"
read protocol
case $protocol in
    1) echo "TCP Connections: "
       netstat -tan
    ;;
    2) echo "UDP Connections: "
       netstat -uan
    ;;
    *) echo "Invalid Protocol Option"
    ;;
esac
;;
6) echo "Killing Program..."
break
;;
*) echo "Invalid Option"
;;
esac
done

^G Help      ^O Write Out   ^W Where Is   ^K Cut        ^T Execute   ^C Location   ^U Undo
^X Exit      ^R Read File   ^H Replace   ^U Paste     ^J Justify   ^- Go To Line ^E Redo

```

Figure 2. Second part of the shell script for network information

- We tested option 1 on the menu (Network connections and listening ports). As we can see, the transport layer protocols are displayed with the ports they are listening on, along with the allowed IP addresses

Active Internet connections (only servers)					
Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	0.0.0.0:80	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:5432	0.0.0.0:*	LISTEN
tcp	0	0	192.168.20.100:53	0.0.0.0:*	LISTEN
tcp	0	0	192.168.20.100:53	0.0.0.0:*	LISTEN
tcp	0	0	192.168.20.100:53	0.0.0.0:*	LISTEN
tcp	0	0	192.168.20.100:53	0.0.0.0:*	LISTEN
tcp6	0	0	:::22	:::*	LISTEN
tcp6	0	0	:::153	:::*	LISTEN
tcp6	0	0	:::153	:::*	LISTEN
tcp6	0	0	:::153	:::*	LISTEN
tcp6	0	0	:::153	:::*	LISTEN
tcp6	0	0	:::5432	:::*	LISTEN
tcp6	0	0	fe80::a00:27ff:fe98::53	:::*	LISTEN
tcp6	0	0	fe80::a00:27ff:fe98::53	:::*	LISTEN
tcp6	0	0	fe80::a00:27ff:fe98::53	:::*	LISTEN
tcp6	0	0	fe80::a00:27ff:fe98::53	:::*	LISTEN
udp	0	0	192.168.20.100:53	0.0.0.0:*	
udp	0	0	192.168.20.100:53	0.0.0.0:*	
udp	0	0	192.168.20.100:53	0.0.0.0:*	
udp	0	0	192.168.20.100:53	0.0.0.0:*	
udp	0	0	192.168.20.100:123	0.0.0.0:*	
udp	0	0	127.0.0.1:123	0.0.0.0:*	
udp	0	0	0.0.0.0:123	0.0.0.0:*	
udp	0	0	0.0.0.0:42114	0.0.0.0:*	
udp	0	0	0.0.0.0:161	0.0.0.0:*	
udp	0	0	0.0.0.0:55768	0.0.0.0:*	
udp6	0	0	:::153	:::*	
udp6	0	0	:::153	:::*	
udp6	0	0	:::153	:::*	
udp6	0	0	:::153	:::*	
udp6	0	0	fe80::a00:27ff:fe98::53	:::*	
udp6	0	0	fe80::a00:27ff:fe98::53	:::*	
udp6	0	0	fe80::a00:27ff:fe98::53	:::*	
			:		

Figure 3. Testing the first option of the script

- We tested option 2 on the menu (Network usage). We can see information about eth1 (Network interface) and lo (Loopback)

```

○ Network information menu:
1) Show network connections and listening ports
2) Show network usage
3) Show routing table
4) Show network interface details
5) Show connections with an especific protocol
6) Exit
2
Network statistics:
Kernel Interface table
Iface      MTU     RX-OK RX-ERR RX-DRP RX-OVR     TX-OK TX-ERR TX-DRP TX-OVR Flg
eth1      1500    43797     0     0 0      4088     0     0     0 0 BMRU
lo        65536       12     0     0 0      12       0     0     0 0 LRU
(CEND)

```

Figure 4. Testing second option of the script

- We tested option 3 on the menu (Routing table). We can see the routes at the routing table

```

Network information menu:
1) Show network connections and listening ports
2) Show network usage
3) Show routing table
4) Show network interface details
5) Show connections with an especific protocol
6) Exit
3
Routing table:
Kernel IP routing table
Destination      Gateway          Genmask        Flags Metric Ref    Use Iface
0.0.0.0          192.168.20.1   0.0.0.0        UG   0      0        0 eth1
127.0.0.0         0.0.0.0        255.0.0.0      U     0      0        0 lo
192.168.20.0     0.0.0.0        255.255.255.0  U     0      0        0 eth1
(END)

```

Figure 5. Testing third option of the script

- We tested option 4 on the menu (Network interface details). We can see the details of eth1

```

Network information menu:
1) Show network connections and listening ports
2) Show network usage
3) Show routing table
4) Show network interface details
5) Show connections with an especific protocol
6) Exit
4
Network interface details
Settings for eth1:
    Supported ports: [ TP ]
    Supported link modes:  10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    MDI-X: off (auto)
    Supports Wake-on: umbg
    Wake-on: d
    Current message level: 0x00000007 (?)           drv probe link
    Link detected: yes
(END)

```

Figure 6. Testing fourth option of the script

- We tested option 5 on the menu (UDP connections and TCP connections)
 - First, we select the TCP option. We can see the ports that TCP is listening to

```

Network information menu:
1) Show network connections and listening ports
2) Show network usage
3) Show routing table
4) Show network interface details
5) Show connections with an especific protocol
6) Exit
5
Select protocol:
1. UDP
2. TCP
1
TCP Connections:
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address      State
tcp     0      0 0.0.0.0:80                0.0.0.0:*           LISTEN
tcp     0      0 0.0.0.0:22                0.0.0.0:*           LISTEN
tcp     0      0 0.0.0.0:5432              0.0.0.0:*           LISTEN
tcp     0      0 192.168.20.100:53        0.0.0.0:*           LISTEN
tcp6    0      0 ::1:22                  ::*:*
tcp6    0      0 ::1:53                  ::*:*
tcp6    0      0 ::5432                 ::*:*
tcp6    0      0 fe80::a00:27ff:fe98::53 ::*:*
(CEND)

```

Figure 7. Testing fifth option with TCP

- Then select UDP option. We can see the ports that UDP is listening to

```

UDP Connections:
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address      State
udp     0      0 192.168.20.100:53        0.0.0.0:*
udp     0      0 192.168.20.100:123       0.0.0.0:*
udp     0      0 127.0.0.1:123            0.0.0.0:*
udp     0      0 0.0.0.0:123             0.0.0.0:*
udp     0      0 0.0.0.0:42114           0.0.0.0:*
udp     0      0 0.0.0.0:161            0.0.0.0:*
udp     0      0 0.0.0.0:55768           0.0.0.0:*
udp6    0      0 ::1:53                 ::*:*
udp6    0      0 fe80::a00:27ff:fe98::53 ::*:*
udp6    0      0 fe80::a00:27ff:fe98:123 ::*:*
udp6    0      0 ::1:123                ::*:*
(CEND)

```

Figure 8. Testing fifth option with UDP

b. Open and running ports

- Additionally, we create a Shell program that indicates (yes or no) whether a port is open and which service is running on that port. We use **tuln** command



```
GNU nano 6.0                               port_listener.sh
#!/bin/bash
echo "Enter the port"
read port
#Check if the port is open
result=$(ss -tuln | grep ":$port ")
if [ ! -z "$result" ]; then
    echo "No service id running on port $port"
else
    service=$(echo "$result" | awk '{print $1, $5}')
    echo "Service running on port $port : $service"
fi
```

Figure 9. shell script for open and running ports

- We tested the script by entering port 80 and unknown ports

```
root@andrea:~# ./port_listener.sh
Enter the port
80
Service running on port 80 : tcp 0.0.0.0:80
root@andrea:~#
```

Figure 10. Testing port 80 in the second shell script

```
root@andrea:~# ./port_listener.sh
Enter the port
111
No service id running on port 111
root@andrea:~# ./port_listener.sh
Enter the port
8080
No service id running on port 8080
root@andrea:~#
```

Figure 11. Testing unknown ports in the second shell script

2. Network Management

On the other hand, part of an organization's technological platform is the monitoring server. Through it, administrators can check the status of the network equipment they manage. These platform management tools enable remote monitoring of disk space, CPU usage, network performance, memory usage, installed software, among other aspects of the network devices.

2.1. Solaris

- To monitor the Solaris virtual machine, we installed Nagios, a real-time monitoring software. First, we need to install some libraries:

- ✓ gd
- ✓ gcc-53

```
root@solaris:~# pkg install gd
          Paquetes que instalar: 15
          Servicios que cambiar: 2
          Crear entorno de inicio: No
          Crear copia de seguridad de entorno de inicio: No

DOWNLOADED                               PKGS      FILES      XFER (MB)      SPEED
Completado                                15/15    434/434    73.2/73.2   3.0M/s

PHASE                                         ITEMS
Instalando acciones nuevas                995/995
Actualizando base de datos de estado de paquete     Listo
Actualizando cachÃ© de paquete            0/0
Actualizando estado de imagen              Listo
Creando base de datos de bÃ³squeda rÃ¡pida en proceso -Loading smf(7) service d
Creando base de datos de bÃ³squeda rÃ¡pida en proceso -
Creando base de datos de bÃ³squeda rÃ¡pida           Listo
Actualizando cachÃ© de paquete            1/1
root@solaris:~#
```

Figure 12. Installing gd on Solaris

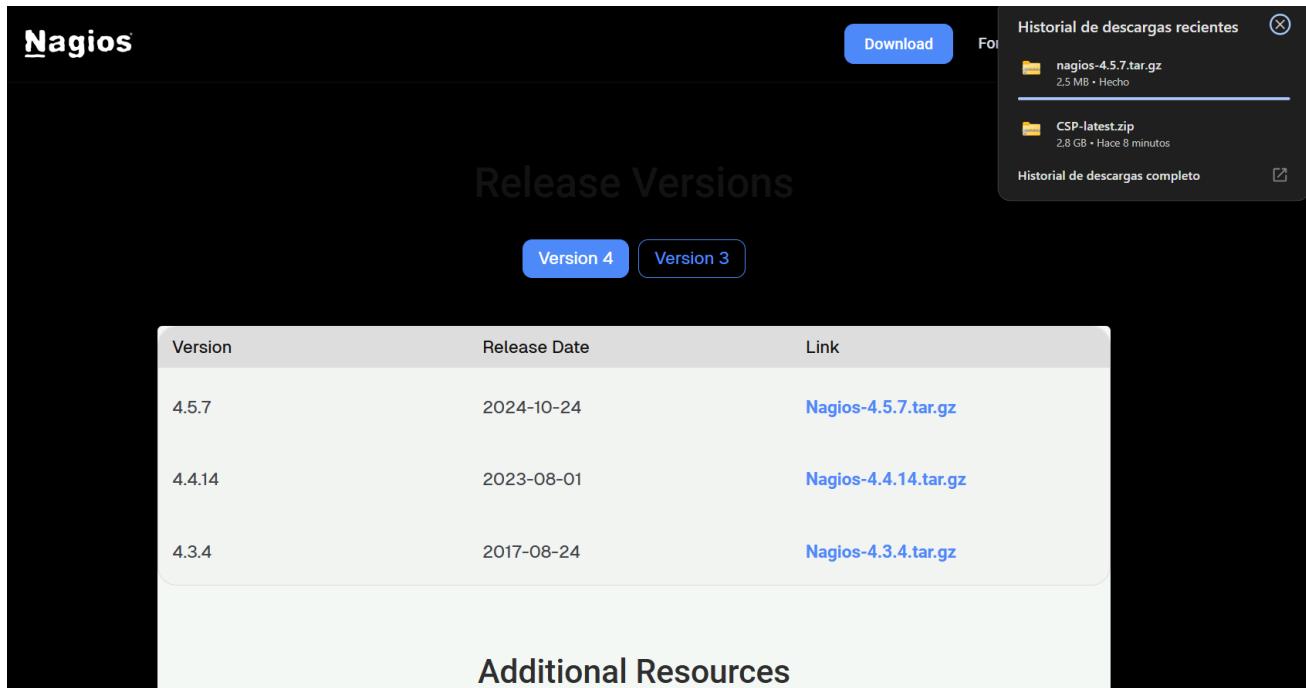
```
root@solaris:~# pkg install gcc-53
          Paquetes que instalar: 11
          Mediadores que cambiar: 1
          Servicios que cambiar: 1
          Crear entorno de inicio: No
          Crear copia de seguridad de entorno de inicio: No

DOWNLOADED                               PKGS      FILES      XFER (MB)      SPEED
Completado                                11/11   1679/1679  263.4/263.4  1.5M/s

PHASE                                         ITEMS
Instalando acciones nuevas                2028/2028
Actualizando base de datos de estado de paquete     Listo
Actualizando cachÃ© de paquete            0/0
Actualizando estado de imagen              Listo
Creando base de datos de bÃ³squeda rÃ¡pida           Listo
Actualizando cachÃ© de paquete            1/1
root@solaris:~#
```

Figure 13. Installing gcc on Solaris

- Then, we go to the official Nagios website, download the **.tar.gz** file and transfer it to the Solaris machine using Samba



Version	Release Date	Link
4.5.7	2024-10-24	Nagios-4.5.7.tar.gz
4.4.14	2023-08-01	Nagios-4.4.14.tar.gz
4.3.4	2017-08-24	Nagios-4.3.4.tar.gz

Additional Resources

Figure 14. Downloading nagios.tar.gz file

- We decompress the file using `tar -xzf nagios-4.5.7.tar.gz`

```
root@solaris:~# ls
CSP-latest.zip      nagios-4.5.7.tar.gz
root@solaris:~# tar -xzf nagios-4.5.7.tar.gz
```

Figure 15. Decompressing .tar.gz file

- We move the generated file (nagios-4.5.7) to the /nagios directory and navigate to it

```
root@solaris:~# mkdir /nagios
root@solaris:~# mv nagios-4.5.7 nagios
root@solaris:~# cd nagios
root@solaris:~/nagios# ls
acllocal.m4          debian           LICENSE          t
autoconf-macros     docs              make-tarball    t-tap
base                doxy.conf        Makefile.in     tap
cgi                 functions        mkgpackage      test
Changelog           html              module          THANKS
common               include           nagios.spec    update-version
config.guess         indent-all.sh  nagios.sysconfig UPGRADING
config.sub           indent.sh       pkginfo.in     worker
configure           install-sh     README.md      xdata
configure.ac         INSTALLING    sample-config
contrib              LEGAL           startup
CONTRIBUTING.md     lib              subst.in
```

Figure 16. Moving nagios-4.5.7 into Nagios directory

- We configure Nagios package using `./configure`

```

config.status: creating lib/iobroker.h
Creating sample config files in sample-config/ ...

*** Configuration summary for nagios 4.5.7 2024-10-24 ***:

General Options:
-----
Nagios executable: nagios
Nagios user/group: nagios,nagios
Command user/group: nagios,nagios
Event Broker: yes
Install ${prefix}: /usr/local/nagios
Install ${includedir}: /usr/local/nagios/include/nagios
Lock file: /var/run/nagios.lock
Check result directory: /usr/local/nagios/var/spool/checkresults
Init directory: /etc/init.d
Apache conf.d directory: /etc/httpd/conf.d
Mail program: /usr/bin/mail
Host OS: solaris2.11
IOBroker Method: poll

Web Interface Options:
-----
HTML URL: http://localhost/nagios/
CGI URL: http://localhost/nagios/cgi-bin/
Traceroute (used by WAP): /usr/sbin/traceroute

```

Review the options above for accuracy. If they look okay, type 'make all' to compile the main program and CGIs.

root@solaris:~/nagios# █

Figure 17. Configuring Nagios

- Before we can start compiling Nagios, there are a couple of source file changes that need to be made, since this version of Nagios defines a structure (**struct comment**) that conflicts with a system structure of the same name in **/usr/include/sys/pwd.h**.
- For that reason, we add the following line as line 28 of the **./worker/ping/worker-ping.c** file:
- `#include <pwd.h>`

root@solaris:~/nagios# nano ./worker/ping/worker-ping.c █

Figure 18. Opening worker-ping.c file

```

Modificado

/*
 * worker-ping.c - Nagios Core 4 worker to handle ping checks
 *
 * Program: Nagios Core
 * License: GPL
 *
 * First Written: 01-03-2013 (start of development)
 *
 * Description:
 *
 * License:
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License version 2 as
 * published by the Free Software Foundation.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software
 * Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
 */

#include <pwd.h>
#include "config.h" [ 466 lÃ±eas leÃ±adas ]
^G Ver ayuda ^O Guardar ^W
^X Salir ^R Leer fich.^N Reemplazar^U Pegar txt ^I OrtografÃ¡a^C Ir a lÃ±ea

```

Figure 19. Changing worker-ping.c file

- We change the following line (line 27) in the ./base/utils.c file from this:
- #include "../include/comments.h" to this "#include <pwd.h>"

```

root@solaris:~/nagios# nano ./base/utils.c

```

Figure 20. Opening utils.c file

```

Modificado

/*****
*
* UTILS.C - Miscellaneous utility functions for Nagios
*
*
* License:
*
* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License version 2 as
* published by the Free Software Foundation.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program; if not, write to the Free Software
* Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
*
****/

#include "../include/config.h"
#include "../include/common.h"
#include "../include/objects.h"
#include "../include/statusdata.h"
#include <pwd.h>
#include "../include/downtime.h"
#include "../include/macros.h" [3830 1Ã±neas leÃ±adas ]
^G Ver ayuda ^O Guardar ^W
^X Salir      ^R Leer fich. ^\ Reemplazar^U Pegar txt ^T OrtografÃ¡a^C Ir a lÃ±ea

```

Figure 21. Changing utils.c file

- We change the following line (line 29) in the `./cgi/Makefile` file from this:
- `CFLAGS=-Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCGI`
- To this:
- `CFLAGS=-Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCGI -I/usr/include/gd2`
-

```

root@solaris:~/nagios# nano ./cgi/Makefile

```

Figure 22. Opening `/cgi/Makefile` file

```

Modificado

BLD_INCLUDE=../include
BLD_LIB=../lib

prefix=/usr/local/nagios
exec_prefix=/usr/local/nagios
LOGDIR=/usr/local/nagios/var
CFGDIR=/usr/local/nagios/etc
BINDIR=/usr/local/nagios/bin
CGIDIR=/usr/local/nagios/sbin
HTMLDIR=/usr/local/nagios/share
INSTALL=/usr/bin/ginstall -c
INSTALL_OPTS=-o nagios -g nagios
COMMAND_OPTS=-o nagios -g nagios
STRIP=/usr/bin/strip

CGIEXTRAS= statuswrl.cgi statusmap.cgi trends.cgi histogram.cgi

CP=@CP@
CC=gcc
CFLAGS=-Wall -I.. -I$(SRC_INCLUDE) -I.. -I$(BLD_INCLUDE)
-I$(BLD_LIB) -g -O2 -I/usr/include/gd2 -DHAVE_CONFIG_H -DNSCGI
JSONFLAGS=-DJSON_NAGIOS_4X

# Compiler flags for optimization (overrides default)
#CFLAGS=-O3 -Wall -Wshadow -Wpointer-arith -Wcast-qual -Wcast-align -Wstrict-pr$

# Compiler flags for optimization (complements default)
#CFLAGS_WARN=-Wall -Wshadow -Wpointer-arith -Wcast-qual -Wcast-align -Wstrict-p$

^G Ver ayuda ^O Guardar ^W
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía-a ^I Ir a lÃ-nea

```

Figure 23. Changing /cgi/Makefile file

- Then we run the following command to compile Nagios

```
root@solaris:~/nagios# gmake all
```

Figure 24. Compile Nagios command

```
web interface

make install-classicui
- This installs the classic theme for the Nagios
  web interface

*** Support Notes *****
If you have questions about configuring or running Nagios,
please make sure that you:
- Look at the sample config files
- Read the documentation on the Nagios Library at:
  https://library.nagios.com

before you post a question to one of the mailing lists.
Also make sure to include pertinent information that could
help others help you. This might include:
- What version of Nagios you are using
- What version of the plugins you are using
- Relevant snippets from your config files
- Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:
https://support.nagios.com
*****
```

Enjoy.

root@solaris:~/nagios-4.5.7# █

Figure 25. Compiling Nagios

- Once everything has been completed, we use the **gmake install**, **gmake install-commandmode**, and **gmake install-config** commands. But first, we need to create the nagios user and nagios group using **groupadd** and **useradd** commands

```
root@solaris:~/nagios# groupadd nagios
root@solaris:~/nagios# useradd -g nagios nagios
root@solaris:~/nagios# gmake install DESTDIR=/root/PROTO█
```

Figure 26. Creating Nagios user

```
*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme

gmake[1]: se sale del directorio '/root/nagios'
gmake install-basic
gmake[1]: se entra en el directorio '/root/nagios'
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /root/PROTO/usr/local/nagios/
libexec
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /root/PROTO/usr/local/nagios/
var
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /root/PROTO/usr/local/nagios/
var/archives
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /root/PROTO/usr/local/nagios/
var/spool/checkresults
chmod g+s /root/PROTO/usr/local/nagios/var/spool/checkresults

*** Main program, CGIs and HTML files installed ***

You can continue with installing Nagios as follows (type 'make'
without any arguments for a list of all possible options):

make install-init
  - This installs the init script in /root/PROTO/etc/init.d

make install-commandmode
  - This installs and configures permissions on the
    directory for holding the external command file

make install-config
  - This installs sample config files in /root/PROTO/usr/local/nagios/etc

gmake[1]: se sale del directorio '/root/nagios'
root@solaris:~/nagios#
```

Figure 27. Installing Nagios using 'gmake install' command

```
root@solaris:~/nagios# gmake install-commandmode
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

root@solaris:~/nagios#
```

Figure 28. Finishing the Nagios configuration with the command 'gmake install-commandmode'

```

root@solaris:~/nagios# gmake install-config
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/ginstall -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr
/local/nagios/etc/nagios.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/lo
cal/nagios/etc/cgi.cfg
/usr/bin/ginstall -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /u
sr/local/nagios/etc/resource.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/ginstall -c -b -m 664 -o nagios -g nagios sample-config/template-object
/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

```

Remember, these are *SAMPLE* config files. You'll need to read the documentation for more information on how to actually define services, hosts, etc. to fit your particular needs.

```
root@solaris:~/nagios# █
```

Figure 29. Finishing the Nagios configuration with the command ‘gmake install-config’

- We go to the created directory and open the **cgi.cfg** file, and in the “**default_user_name**” variable we set it **nagiosadmin** to get the correct authorization for Nagios

```

root@solaris:~/nagios# cd /root/PROTO/usr/local/nagios/etc/
root@solaris:~/PROTO/usr/local/nagios/etc# ls
cgi.cfg      nagios.cfg      objects      resource.cfg
root@solaris:~/PROTO/usr/local/nagios/etc# nano cgi.cfg █

```

Figure 30. Opening cgi.cfg file

```

Modificado

# access pages without authentication. This allows people within a
# secure domain (i.e., behind a firewall) to see the current status
# without authenticating. You may want to use this to avoid basic
# authentication if you are not using a secure server since basic
# authentication transmits passwords in the clear.
#
# Important: Do not define a default username unless you are
# running a secure web server and are sure that everyone who has
# access to the CGIs has been authenticated in some manner! If you
# define this variable, anyone who has not authenticated to the web
# server will inherit all rights you assign to this user!
#
default_user_name=nagiosadmin

#
#
# SYSTEM/PROCESS INFORMATION ACCESS
# This option is a comma-delimited list of all usernames that
# have access to viewing the Nagios process information as
# provided by the Extended Information CGI (extinfo.cgi). By
# default, *no one* has access to this unless you choose to
# not use authorization. You may use an asterisk (*) to
# authorize any user who has authenticated to the web server.

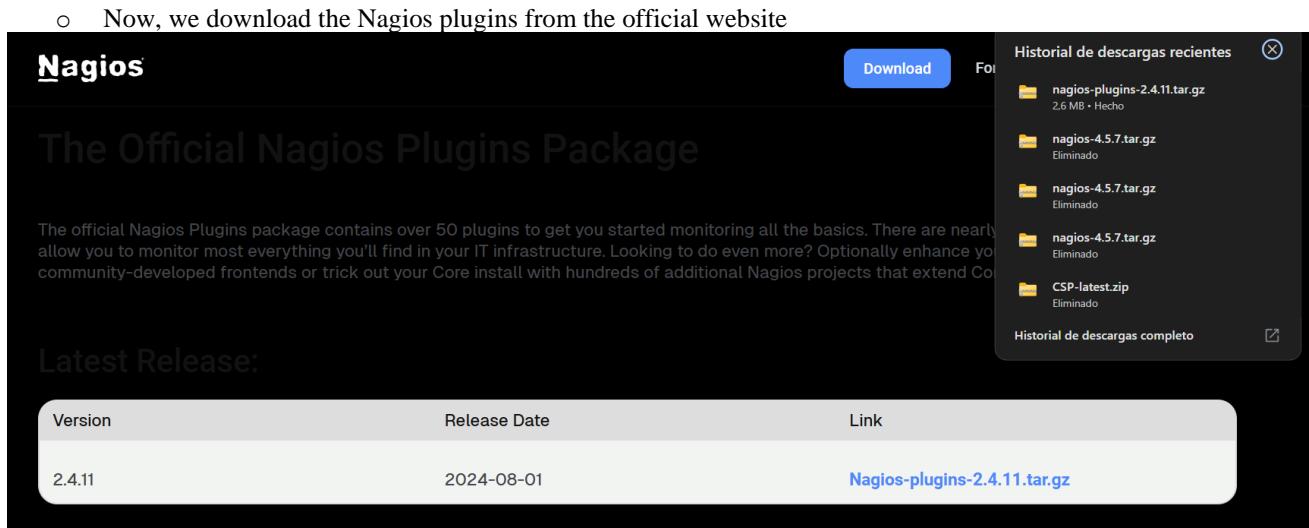
authorized_for_system_information=nagiosadmin

#
#
# CONFIGURATION INFORMATION ACCESS
^G Ver ayuda ^O Guardar ^W
^X Salir      ^R Leer fich.^V Reemplazar^U Pegar txt ^T OrtografÃ¡-a^I Ir a lÃnea

```

Figure 31. Configuring cgi.cfg file

- Now, we download the Nagios plugins from the official website



The screenshot shows the official Nagios Plugins package page. At the top, there's a navigation bar with 'Nagios' and a 'Download' button. Below it, a large heading says 'The Official Nagios Plugins Package'. A section titled 'Latest Release:' lists the version 2.4.11, release date 2024-08-01, and a download link to 'Nagios-plugins-2.4.11.tar.gz'. To the right, a sidebar titled 'Historial de descargas recientes' shows a list of previously downloaded files, including 'nagios-plugins-2.4.11.tar.gz', 'nagios-4.5.7.tar.gz', 'nagios-4.5.7.tar.gz', 'nagios-4.5.7.tar.gz', and 'CSP-latest.zip'. There's also a link to 'Historial de descargas completa'.

Figure 32. Downloading nagios plugins

- We decompress the .tar.gz file, then we use ./configure and make install

```
root@solaris:~# mv /compartir/nagios-plugins-2.4.11.tar.gz /root/
root@solaris:~# ls
nagios                               pax_global_header
nagios-4.5.7.tar.gz                  PROTO
nagios-plugins-2.4.11.tar.gz
root@solaris:~# tar -zxf nagios-plugins-2.4.11-tar.gz
```

Figure 33. Decompressing nagios-plugins

```
--with-gnutls: no
--enable-extra-opts: yes
--with-perl: /usr/bin/perl
--enable-perl-modules: no
--with-cgiurl: /nagios/cgi-bin
--with-trusted-path: /usr/local/sbin:/usr/local/bin:/sbin:/bin:/u
sr/sbin:/usr/bin
--enable-libtap: no
checking that generated files are newer than configure... done
configure: creating ./config.status
config.status: creating gl/Makefile
config.status: creating nagios-plugins.spec
config.status: creating tools/build_perl_modules
config.status: creating Makefile
config.status: creating tap/Makefile
config.status: creating lib/Makefile
config.status: creating plugins/Makefile
config.status: creating lib/tests/Makefile
config.status: creating plugins-root/Makefile
config.status: creating plugins-scripts/Makefile
config.status: creating plugins-scripts/utils.pm
config.status: creating plugins-scripts/utils.sh
config.status: creating perlmods/Makefile
config.status: creating test.pl
config.status: creating pkg/solaris/pkginfo
config.status: creating po/Makefile.in
config.status: creating config.h
config.status: config.h is unchanged
config.status: executing depfiles commands
config.status: executing libtool commands
config.status: executing po-directories commands
config.status: creating po/POTFILES
config.status: creating po/Makefile
root@solaris:~/nagios-plugins-2.4.11#
```

Figure 34. Configuring nagios plugins

```
root@solaris:~/nagios-plugins-2.4.11# gmake install DESTDIR=/root/PROTO
```

Figure 35. Using 'gmake install' to install nagios plugins

```

chmod ug=rx,u+s /root/PROTO/usr/local/nagios/libexec/check_icmp
/usr/bin/ginstall -c pst3 /root/PROTO/usr/local/nagios/libexec/pst3
chown root /root/PROTO/usr/local/nagios/libexec/pst3
chmod ug=rx,u+s /root/PROTO/usr/local/nagios/libexec/pst3
gmake[2]: No se hace nada para 'install-data-am'.
gmake[2]: se sale del directorio '/root/nagios-plugins-2.4.11/plugins-root'
gmake[1]: se sale del directorio '/root/nagios-plugins-2.4.11/plugins-root'
Making install in po
gmake[1]: se entra en el directorio '/root/nagios-plugins-2.4.11/po'
/usr/bin/gmkdir -p /root/PROTO/usr/local/nagios/share
installing fr.gmo as /root/PROTO/usr/local/nagios/share/locale/fr/LC_MESSAGES/nagios-plugins.mo
installing de.gmo as /root/PROTO/usr/local/nagios/share/locale/de/LC_MESSAGES/nagios-plugins.mo
if test "nagios-plugins" = "gettext-tools"; then \
  /usr/bin/gmkdir -p /root/PROTO/usr/local/nagios/share/gettext/po; \
  for file in Makefile.in.in remove-potdate.sin Makevars.template; do \
    /usr/bin/ginstall -c -m 644 ./${file} \
      /root/PROTO/usr/local/nagios/share/gettext/po/${file}; \
  done; \
  for file in Makevars; do \
    rm -f /root/PROTO/usr/local/nagios/share/gettext/po/${file}; \
  done; \
else \
: ; \
fi
gmake[1]: se sale del directorio '/root/nagios-plugins-2.4.11/po'
gmake[1]: se entra en el directorio '/root/nagios-plugins-2.4.11'
gmake[2]: se entra en el directorio '/root/nagios-plugins-2.4.11'
gmake[2]: No se hace nada para 'install-exec-am'.
gmake[2]: No se hace nada para 'install-data-am'.
gmake[2]: se sale del directorio '/root/nagios-plugins-2.4.11'
gmake[1]: se sale del directorio '/root/nagios-plugins-2.4.11'
root@solaris:~/nagios-plugins-2.4.11# █

```

Figure 36. Installing nagios plugins

- We open the file **/usr/local/nagios/etc/objects/hosts.cfg** to configure the hosts (machines to be monitored) and add the following configuration (we must specify the IP address of the Solaris machine):

```
Modificado

define host {
    use          linux-server
    host_name    solaris
    alias        solaris server
    address      192.168.20.101
    max_check_attempts 3
    check_interval     5
    retry_interval      1
    check_command       check-host-alive
    contact_groups     admins
    notification_interval 30
    notification_period 24x7
}

[ Nuevo fichero ]
^G Ver ayuda ^O Guardar ^W Bu
^X Salir   ^R Leer fich.^V Reemplazar^U Pegar txt ^T Ortografía-a^I Ir a lÃnea
```

Figure 37. Hosts monitoring configuration

- We open the file **/usr/local/nagios/etc/objects/services.cfg** to configure the services (CPU,disk, etc) and add the following configuration (we must specify the IP address of the Solaris machine):

```

define service {
    use generic-service
    host_name solaris
    service_description solaris snmp
    check_command check_snmp!-H 192.168.20.101 -C public -o sysUp$5
    normal_check_interval 1
    retry_check_interval 3
    max_check_attempts 24x7
    check_period 30
    notification_interval 24x7
    notification_period 24x7
}

[ 11 lÃ±eas leÃ±adas ]
^G Ver ayuda ^O Guardar ^W B
^X Salir      ^R Leer fich.^` Reemplazar^U Pegar txt ^T OrtografÃ¡-a^I Ir a lÃ±ea

```

Figure 38. Nagios services configuration

- We verify that the configuration has no syntax errors with the following command

```
root@solaris:~# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

Figure 39. Command to verify the Nagios configuration

- We can see that there's no errors

```

License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 9 services.
  Checked 2 hosts.
  Checked 1 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 24 commands.
  Checked 5 time periods.
  Checked 0 host escalations.
  Checked 0 service escalations.

Checking for circular paths...
  Checked 2 hosts
  Checked 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timeperiods

Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
root@solaris:~# 

```

Figure 40. Verifying Nagios configuration

- We set a password for the user nagiosadmin using the command “htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin”

```

root@solaris:~# htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
root@solaris:~# 

```

Figure 41. Creating a password for the user nagiosadmin

- Now, we need to configure SNMP, we open **/etc/net-snmp/snmp/snmpd.conf**

```

root@solaris:~# nano /etc/net-snmp/snmp/snmpd.conf#

```

Figure 42. Opening snmpd.conf

- We add the following configuration to the file:
- Rocommunity public
- Syslocation “Solaris Machine”
- Syscontact “[email to contact]”

```

Modificado

# This section defines who is allowed to talk to your running
# snmp agent.

# rocommunity: a SNMPv1/SNMPv2c read-only access community name
#   arguments: community [default|hostname|network/bits] [oid]

rocommunity public

#####
# SEA subagents dynamically register with the master agent via port 161,
# supplying a read-write community string on the request (e.g. 'private'
# for DMI). If the community strings used are not defined in the
# snmpd.conf file, the registration request will not be forwarded to
# the SEA master agent.
#
# rwcommunity: a SNMPv1/SNMPv2c read-write access community name
#   arguments: community [default|hostname|network/bits] [oid]
#
# The following entry provides minimum access for successful
# SEA subagent registration.
#
#rwcommunity private localhost .1.3.6.1.4.1.42.2.15
#rwcommunity private

#####

# SECTION: System Information Setup
#
^G Ver ayuda ^O Guardar ^W
^X Salir    ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía-a ^_ Ir a lÃnea

```

Figure 43. Configuring SNMP

- Additionally, we add the following configuration to monitor the disk, eth0 and lo0 interfaces, and allow the connection with Nagios

```

Modificado

# the agent return the "notWritable" error code. IE, including
# this token in the snmpd.conf file will disable write access to
# the variable.
# arguments: contact_string

syscontact andreacamit@gmail.com
sysservices 72

#
# dlmods entries
# for 32bit agent
#
#dlmod seaExtensions /usr/lib/libseaExtensions.so
#
# for 64bit agent
#dlmod seaExtensions /usr/lib/amd64/libseaExtensions.so
#dlmod seaExtensions /usr/lib/sparcv9/libseaExtensions.so

#
#Monitoreo del sistema
load 5
disk / 10% #Alerta si el espacio disponible es menor al 10
#Monitoreo Interfaces de Red
interface eth0 #Se monitorea la interfaz de red eth0
interface lo0 #Monitoreo del loopback

proc nagios 1 1

^G Ver ayuda ^O Guardar ^W
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía-a ^_ Ir a lÃnea

```

Figure 44. Adding additional information on snmpd.conf

- We test the functionality with **snmpwalk -v 2c -c public [ip]**

```

-EVENT-MIB::mteTriggerFired
DISMAN-EVENT-MIB::mteEventNotification."_snmpd".'_mteTriggerRising' = OID: DISMA
N-EVENT-MIB::mteTriggerRising
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_linkDown' = STRING
: _snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_linkUp' = STRING:
_snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_mteTriggerFailure'
= STRING: _snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_mteTriggerFalling'
= STRING: _snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_mteTriggerFired' =
STRING: _snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjectsOwner."_snmpd".'_mteTriggerRising'
= STRING: _snmpd
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_linkDown' = STRING: _li
nkUpDown
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_linkUp' = STRING: _link
UpDown
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_mteTriggerFailure' = ST
RING: _triggerFail
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_mteTriggerFalling' = ST
RING: _triggerFire
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_mteTriggerFired' = STRI
NG: _triggerFire
DISMAN-EVENT-MIB::mteEventNotificationObjects."_snmpd".'_mteTriggerRising' = STR
ING: _triggerFire
NOTIFICATION-LOG-MIB::nlmConfigGlobalEntryLimit.0 = Gauge32: 1000
NOTIFICATION-LOG-MIB::nlmConfigGlobalAgeOut.0 = Gauge32: 1440 minutes
NOTIFICATION-LOG-MIB::nlmStatsGlobalNotificationsLogged.0 = Counter32: 0 notific
ations
NOTIFICATION-LOG-MIB::nlmStatsGlobalNotificationsBumped.0 = Counter32: 0 notific
ations
root@solaris:~# ■

```

Figure 45. Testing SNMP service on Solaris

- We start SNMP service

```

root@solaris:~# svcadm enable svc:/application/management/net-snmp:default
root@solaris:~# svcadm restart svc:/application/management/net-snmp:default
root@solaris:~# ■

```

Figure 46. Starting SNMP service on Solaris

- Now we need to configure Apache to access Nagios services from the web browser. To do this, we open the file **/etc/apache2/2.4/httpd.conf** and add the configuration shown in *image 45*

```

<IfModule alias_module>
    ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin/"
    Alias /nagios /usr/local/nagios/share
</IfModule>
<Directory "/usr/local/nagios/sbin/">
    Options +ExecCGI
    AddHandler cgi-script.cgi
    AllowOverride None
    Require all granted
    AuthType Basic
    AuthName "Nagios Access"
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    Require valid-user
</Directory>
<Directory "/usr/local/nagios/share">
    Options None
    AllowOverride None
    Require all granted
    DirectoryIndex index.php index.html
    AuthType Basic
    AuthName "Nagios Access"
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    Require valid-user
</Directory>

```

Figure 47. Apache file configuration

- We restart the Apache service with the command **svcadm restart apache24**, then we restart the SNMP service with the command **svcadm restart svc:/application/management/net-snmp:default**. Finally, we start Nagios with the command **/usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg**

```

root@solaris:~# svcadm restart apache24
root@solaris:~# svcadm -v enable /network/http:apache24
svc:/network/http:apache24 activado.
root@solaris:~# svcadm restart svc:/application/management/net-snmp:default
root@solaris:~# svcadm enable svc:/application/management/net-snmp:default
root@solaris:~# /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.5.7
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-10-24
License: GPL

Website: https://www.nagios.org
Nagios 4.5.7 starting... (PID=1281)
Local time is Thu Nov 14 20:28:36 -05 2024
wproc: Successfully registered manager as @wproc with query handler
wproc: Registry request: name=Core Worker 1283;pid=1283
wproc: Registry request: name=Core Worker 1284;pid=1284
wproc: Registry request: name=Core Worker 1282;pid=1282
wproc: Registry request: name=Core Worker 1285;pid=1285

```

Figure 48. Initializing Nagios, SNMP and Apache

- We open the browser and enter the Solaris IP along with the Nagios address (**<solaris_ip>/nagios**). We type in the previously configured user along with the assigned password

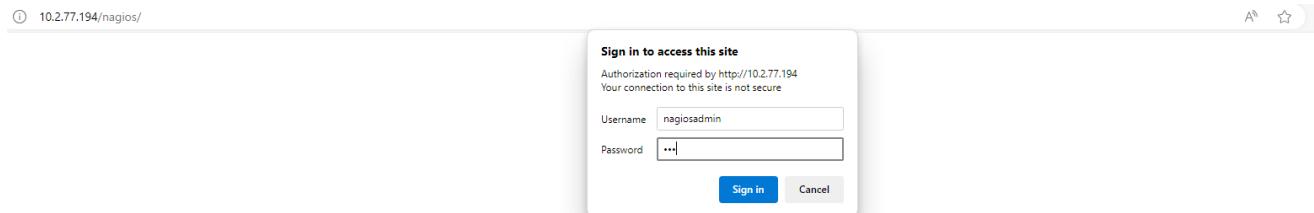


Figure 49. Accessing the Nagios web interface

- Once authenticated, we access the Nagios web interface with the services it offers

Figure 50. Nagios web interface

Host Alert History
 Last Updated: Thu Nov 14 22:35:48 -05 2024
 Nagios® Core™ 4.5.7 - www.nagios.org
 Logged in as nagiosadmin

General
 Home Documentation Current Status
 Tactical Overview Map Hosts Services Host Groups Summary Grid Service Groups Summary Grid Problems Services (Unhandled) Hosts (Unhandled) Network Outages

Quick Search: solaris

Reports
 Availability Trends Alerts History Summary Histogram Notifications Event Log System Comments

Host 'solaris-server'
 Latest Archive Log File Navigation
 Thu Nov 14 00:00:00 -05 2024 to Present..
 File: /usr/local/nagios/var/nagios.log

noviembre 14, 2024 22:00

[11-14-2024 22:35:19] Nagios 4.5.7 starting... (PID=1785)
 [11-14-2024 22:21:08] Nagios 4.5.7 starting... (PID=1734)
 [11-14-2024 22:19:34] Nagios 4.5.7 starting... (PID=1715)
 [11-14-2024 22:18:29] Bailing out due to one or more errors encountered in the configuration files. Run Nagios from the command line with the -v option to verify your config before restarting. (PID=1707)
 [11-14-2024 22:18:28] Nagios 4.5.7 starting... (PID=1707)
 [11-14-2024 22:12:13] Nagios 4.5.7 starting... (PID=1650)
 [11-14-2024 22:04:56] Bailing out due to one or more errors encountered in the configuration files. Run Nagios from the command line with the -v option to verify your config before restarting. (PID=1656)
 [11-14-2024 22:04:56] Nagios 4.5.7 starting... (PID=1656)

noviembre 14, 2024 21:00

[11-14-2024 21:51:59] Nagios 4.5.7 starting... (PID=1545)
 [11-14-2024 21:44:27] Nagios 4.5.7 starting... (PID=1487)
 [11-14-2024 21:35:57] Nagios 4.5.7 starting... (PID=1404)
 [11-14-2024 21:32:53] Nagios 4.5.7 starting... (PID=1363)
 [11-14-2024 21:31:43] Nagios 4.5.7 starting... (PID=1354)
 [11-14-2024 21:20:30] Nagios 4.5.7 starting... (PID=1301)
 [11-14-2024 21:16:56] Nagios 4.5.7 starting... (PID=1292)
 [11-14-2024 21:06:54] Nagios 4.5.7 starting... (PID=1262)
 [11-14-2024 21:04:58] Nagios 4.5.7 starting... (PID=1250)

noviembre 14, 2024 20:00

[11-14-2024 20:28:36] Nagios 4.5.7 starting... (PID=1281)
 [11-14-2024 20:24:48] Nagios 4.5.7 starting... (PID=1262)

State type options:
 All state types
 Hide Flapping Alerts
 Hide Downtime Alerts
 Hide Process Messages
 Older Entries First
 Update

History detail level for this host:
 All alerts
 Hide Flapping Alerts
 Hide Downtime Alerts
 Hide Process Messages
 Older Entries First

Figure 51. Solaris Host log information into Nagios

Host Alert Histogram
 Last Updated: Thu Nov 14 22:36:55 -05 2024
 Nagios® Core™ 4.5.7 - www.nagios.org
 Logged in as nagiosadmin

General
 Home Documentation Current Status
 Tactical Overview Map Hosts Services Host Groups Summary Grid Service Groups Summary Grid Problems Services (Unhandled) Hosts (Unhandled) Network Outages

Quick Search: solaris

Reports
 Availability Trends Alerts History Summary Histogram Notifications Event Log System

Host 'solaris-server'
 Report period: Last 7 Days Assume state retention: yes
 Breakdown type: Day of the Month Initial states logged: no
 Events to graph: All host events Events to ignore: no
 State types to graph: Hard and soft states Update

Event History For Host 'solaris-server'
 Thu Nov 7 22:36:55 2024 to Thu Nov 14 22:36:55 2024

Number of Events

Day of the Month

EVENT TYPE	MIN	MAX	SUM	Avg
Recovery (Up):	0	0	0	0,00
Down:	0	0	0	0,00
Unreachable:	0	0	0	0,00

Figure 52. Solaris Host Graphic

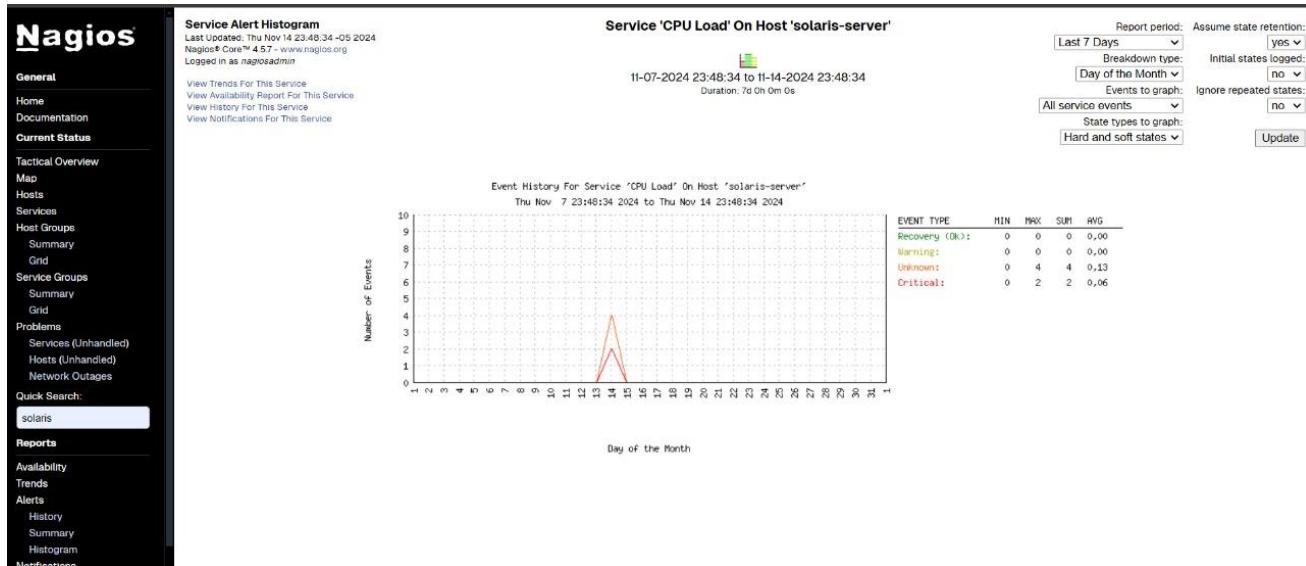


Figure 53. CPU Service of Solaris Graphic

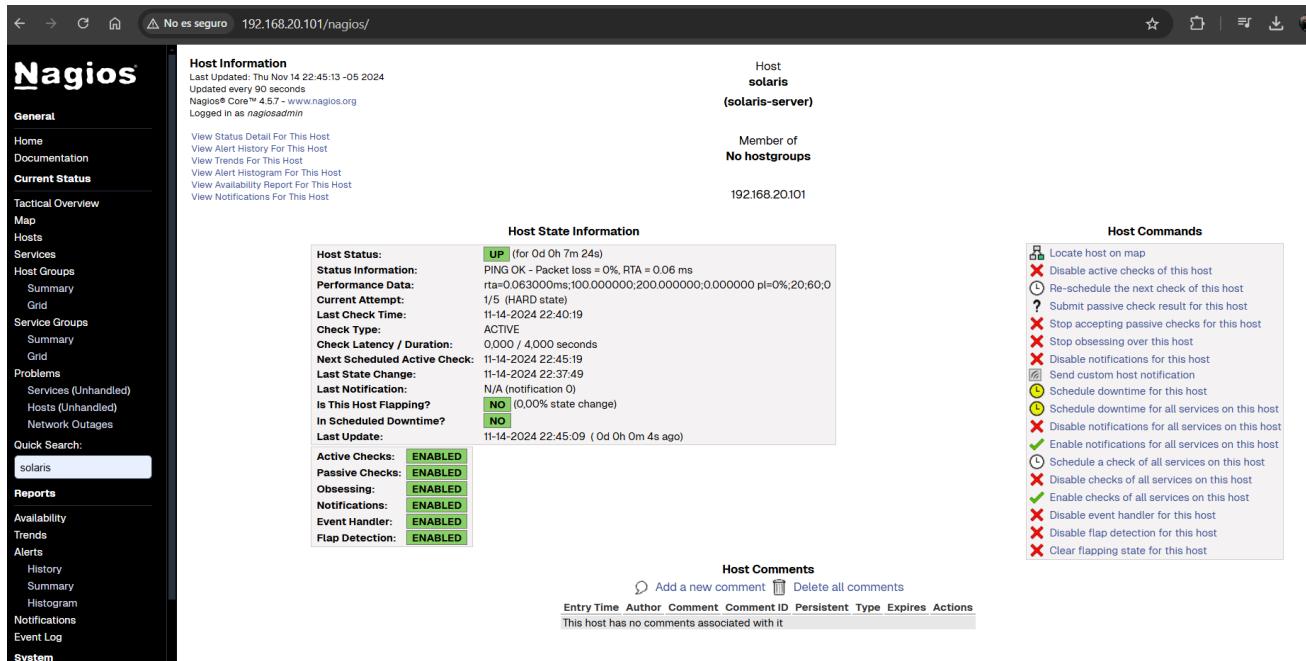


Figure 54. Solaris Host information into Nagios

2.2. Linux Slackware

- We install SNMP using slackpkg command

```
root@andrea:~# slackpkg install net-snmp
```

Figure 55. Command to install SNMP

- We select the package and wait for it to install

```
slackpkg 15.0.10
```

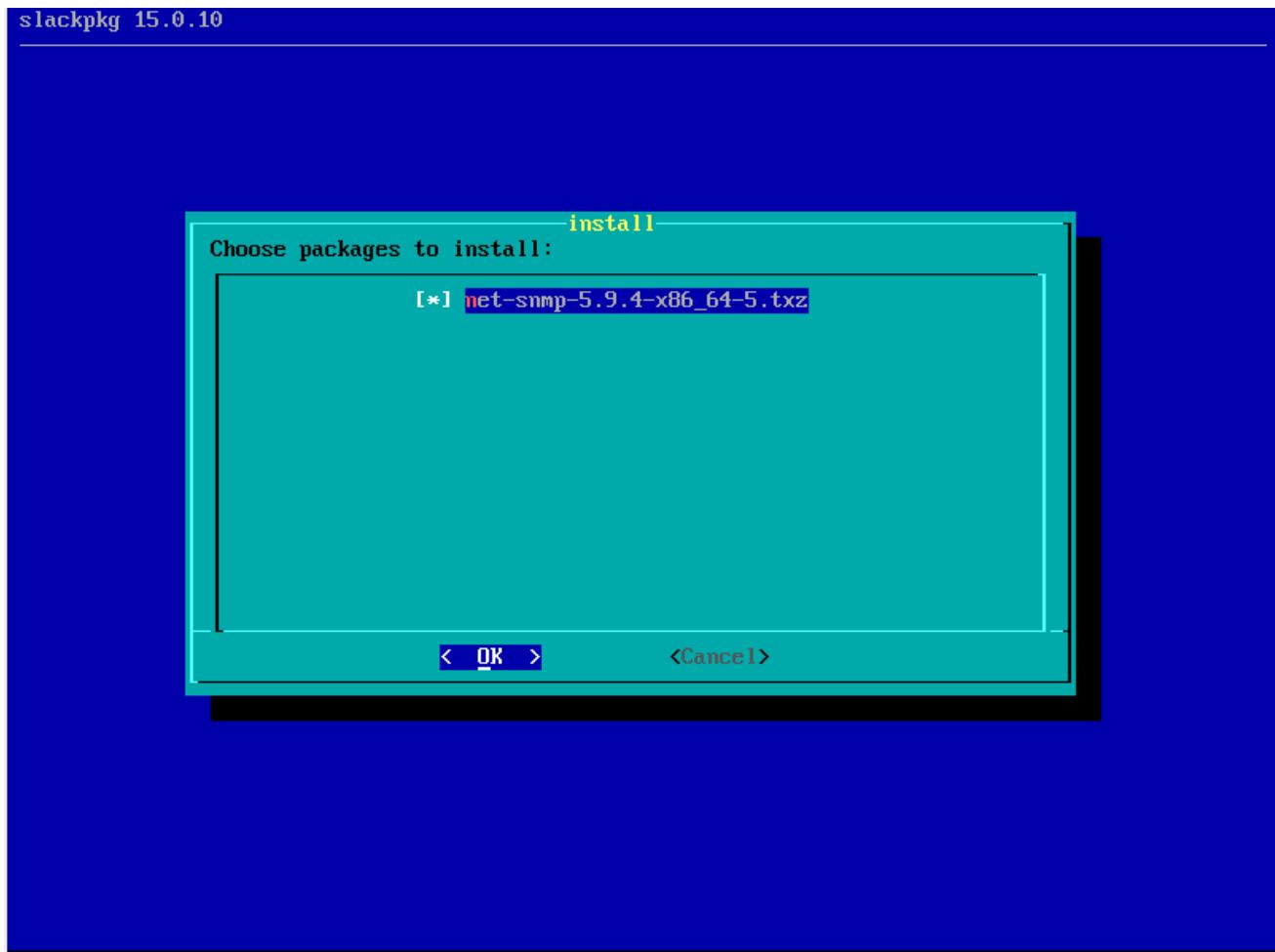


Figure 56. Selecting SNMP package

```

re64/n/net-snmp-5.9.4-x86_64-5.txz.asc...
--2024-11-08 13:39:57-- http://ftp.slackware-brasil.com.br/slackware64-current/slackware64/n/net-snmp-5.9.4-x86_64-5.txz.asc
Resolving ftp.slackware-brasil.com.br (ftp.slackware-brasil.com.br)... 200.137.217.134
Connecting to ftp.slackware-brasil.com.br (ftp.slackware-brasil.com.br)|200.137.217.134|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 195 [text/plain]
Saving to: //var/cache/packages//slackware64/n/net-snmp-5.9.4-x86_64-5.txz.asc

//var/cache/packages//s 100%[=====] 195 --.-KB/s in 0s

2024-11-08 13:39:57 (16.8 MB/s) - //var/cache/packages//slackware64/n/net-snmp-5.9.4-x86_64-5.txz.asc saved [195/195]

      Package net-snmp-5.9.4-x86_64-5.txz is already in cache - not downloading
      Installing net-snmp-5.9.4-x86_64-5...
Verifying package net-snmp-5.9.4-x86_64-5.txz.
Installing package net-snmp-5.9.4-x86_64-5.txz:
PACKAGE DESCRIPTION:
# net-snmp (Simple Network Management Protocol tools)
#
# Various tools relating to the Simple Network Management Protocol:
#
# An extensible agent
# An SNMP library
# Tools to request or set information from SNMP agents
# Tools to generate and handle SNMP traps
# A version of the UNIX 'netstat' command using SNMP
# A graphical Perl/Tk/SNMP based mib browser
#
Executing install script for net-snmp-5.9.4-x86_64-5.txz.
Package net-snmp-5.9.4-x86_64-5.txz installed.
Searching for NEW configuration files...
      No .new files found.

root@andrea:~# _

```

Figure 57. SNMP installation

- We open /etc/snmp/snmpd.conf file

```

root@andrea:~# nano /etc/snmp/snmpd.conf

```

Figure 58. Opening snmpd.conf file

- At the end of the file, we add the configuration to monitor the devices and, additionally, the port on which it listens

```

#           script in the right location. (its not installed by default)

# % snmpwalk -v 1 localhost -c public .1.3.6.1.4.1.2021.255
# enterprises.ucdavis.255.1 = "life the universe and everything"
# enterprises.ucdavis.255.2.1 = 42
# enterprises.ucdavis.255.2.2 = OID: 42.42.42
# enterprises.ucdavis.255.3 = Timeticks: (363136200) 42 days, 0:42:42
# enterprises.ucdavis.255.4 = IpAddress: 127.0.0.1
# enterprises.ucdavis.255.5 = 42
# enterprises.ucdavis.255.6 = Gauge: 42
#
# % snmpget -v 1 localhost public .1.3.6.1.4.1.2021.255.5
# enterprises.ucdavis.255.5 = 42
#
# % snmpset -v 1 localhost public .1.3.6.1.4.1.2021.255.1 s "New string"
# enterprises.ucdavis.255.1 = "New string"
#
# For specific usage information, see the man/snmpd.conf.5 manual page
# as well as the local/passtest script used in the above example.

# Added for support of bcm5820 cards.
pass .1.3.6.1.4.1.4413.4.1 /usr/bin/ucd5820stat

#####
# Further Information
#
# See the snmpd.conf manual page, and the output of "snmpd -H".
rocommunity public
rocommunity public 192.168.20.101
rocommunity public 192.168.20.160
rocommunity public 192.168.20.100
agentAddress udp:161

root@andrea:~#

```

Figure 59. Configuring SNMP

- We start the service, but as we can see, we are missing some libraries
 - ✓ Libnsl
 - ✓ Libnl3.0.200
 - ✓ Libsensors.so.5
 - ✓ pciutils

```

root@andrea:~# /etc/rc.d/rc.snmpd start
Starting snmpd: /usr/sbin/snmpd: error while loading shared libraries: libnsl.so.3: cannot open shared object file: No such file or directory
/usr/sbin/snmpd -A -p /var/run/snmpd -a -c /etc/snmp/snmpd.conf
root@andrea:~#

```

Figure 60. SNMP starting failed

```
-2.0.1-x86_64-1.txz.asc
Resolving ftp.slackware-brasil.com.br (ftp.slackware-brasil.com.br)... 200.137.217.134
Connecting to ftp.slackware-brasil.com.br (ftp.slackware-brasil.com.br)|200.137.217.134|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 163 [text/plain]
Saving to: //var/cache/packages//slackware64/l/libnsl-2.0.1-x86_64-1.txz.asc

//var/cache/packages//slackware64/l/libnsl-2.0.1-x86_64-1.txz.asc 100%[=====] 163 --.-KB/s in 0s

2024-11-12 23:16:46 (23.9 MB/s) - //var/cache/packages//slackware64/l/libnsl-2.0.1-x86_64-1.txz.asc saved [163/163]

      Package libnsl-2.0.1-x86_64-1.txz is already in cache - not downloading
      Installing libnsl-2.0.1-x86_64-1...
Verifying package libnsl-2.0.1-x86_64-1.txz.
Installing package libnsl-2.0.1-x86_64-1.txz:
PACKAGE DESCRIPTION:
# libnsl (NIS/YP library)
#
# This package contains the libnsl library. This library contains the
# public client interface for NIS(YP). This code was formerly part of
# glibc, but is now standalone to be able to link against TI-RPC for
# IPv6 support.
#
# Homepage: https://github.com/thkukuk/libnsl
#
Executing install script for libnsl-2.0.1-x86_64-1.txz.
Package libnsl-2.0.1-x86_64-1.txz installed.
Searching for NEW configuration files...
      No .new files found.
```

Figure 61. Installing libnsl

```
root@andrea:~# installpkg libnl3-3.5.0-x86_64-3.txz
Verifying package libnl3-3.5.0-x86_64-3.txz.
Installing package libnl3-3.5.0-x86_64-3.txz:
PACKAGE DESCRIPTION:
# libnl3 (Netlink Protocol Library Suite version 3)
#
# The libnl suite is a collection of libraries providing APIs to
# netlink protocol based Linux kernel interfaces. Netlink is a IPC
# mechanism primarily between the kernel and user space processes.
# It was designed to be a more flexible successor to ioctl to provide
# mainly networking related kernel configuration and monitoring
# interfaces.
#
# Homepage: https://github.com/thom311/libnl
#
Executing install script for libnl3-3.5.0-x86_64-3.txz.
Package libnl3-3.5.0-x86_64-3.txz installed.
root@andrea:~#
```

Figure 62. Installing libnl

```
root@andrea:~# installpkg lib64lm_sensors5-3.6.0-2-omv4090.x86_64.tgz
Verifying package lib64lm_sensors5-3.6.0-2-omv4090.x86_64.tgz.
Installing package lib64lm_sensors5-3.6.0-2-omv4090.x86_64.tgz:
PACKAGE DESCRIPTION:
Package lib64lm_sensors5-3.6.0-2-omv4090.x86_64.tgz installed.
root@andrea:~#
```

Figure 63. Installing libsensors

```
.asc■ saved [195/195]

      Package pciutils-3.13.0-x86_64-1.txz is already in cache - not downloading
      Installing pciutils-3.13.0-x86_64-1...
Verifying package pciutils-3.13.0-x86_64-1.txz.
Installing package pciutils-3.13.0-x86_64-1.txz:
PACKAGE DESCRIPTION:
# pciutils (PCI utilities)
#
# lspci displays detailed information about all PCI buses and devices
# in the system, replacing the original /proc/pci interface.
#
# setpci allows reading from and writing to PCI device configuration
# registers. For example, you can adjust the latency timers with it.
#
# See the manual pages for more details.
#
Executing install script for pciutils-3.13.0-x86_64-1.txz.
Package pciutils-3.13.0-x86_64-1.txz installed.
Searching for NEW configuration files...
Some packages had new configuration files installed (1 new files):
/etc/snmp/snmpd.conf.new

What do you want (K/O/R/P)?
      (K)eep the old files and consider .new files later
      (O)verwrite all old files with the new ones. The
          old files will be stored with the suffix .orig
      (R)emove all .new files
      (P)rompt K, O, R selection for every single file
O

root@andrea:~# _
```

Figure 64. Installing pciutils

- Now we can start the SNMP service

```
root@andrea:~# /etc/rc.d/rc.snmpd restart
Shutting down snmpd: DONE
Starting snmpd: /usr/sbin/snmpd -A -p /var/run/snmpd -a -c /etc/snmp/snmpd.conf
root@andrea:~# /usr/sbin/snmpd -A -p /var/run/snmpd -a -c /etc/snmp/snmpd.conf
root@andrea:~# _
```

Figure 65. Starting SNMP on Slackware

- We test the functionality with **snmpwalk -v 2c -c public [ip]**

```

SNMPv2-MIB::sysLocation.0 = STRING: Unknown (edit /etc/snmp/snmpd.conf)
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORID.1 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.2 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.3 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORID.4 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.5 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORID.6 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.7 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.8 = OID: IP-MIB::ip
SNMPv2-MIB::sysORID.9 = OID: SNMP-NOTIFICATION-MIB::snmpNotifyFullCompliance
SNMPv2-MIB::sysORID.10 = OID: NOTIFICATION-LOG-MIB::notificationLogMIB
SNMPv2-MIB::sysORDescr.1 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.3 = STRING: The management information definitions for the SNMP User-based Security Model.
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for SNMPv2 entities
SNMPv2-MIB::sysORDescr.5 = STRING: View-based Access Control Model for SNMP.
SNMPv2-MIB::sysORDescr.6 = STRING: The MIB module for managing TCP implementations
SNMPv2-MIB::sysORDescr.7 = STRING: The MIB module for managing UDP implementations
SNMPv2-MIB::sysORDescr.8 = STRING: The MIB module for managing IP and ICMP implementations
SNMPv2-MIB::sysORDescr.9 = STRING: The MIB modules for managing SNMP Notification, plus filtering.
SNMPv2-MIB::sysORDescr.10 = STRING: The MIB module for logging SNMP Notifications.
SNMPv2-MIB::sysORUpTime.1 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.2 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.3 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.4 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.5 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.6 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.7 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.8 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.9 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.10 = Timeticks: (0) 0:00:00.00
HOST-RESOURCES-MIB::hrSystemUptime.0 = Timeticks: (148539) 0:24:45.39
HOST-RESOURCES-MIB::hrSystemUptime.0 = No more variables left in this MIB View (It is past the end of the MIB tree)
root@andrea:~# 

```

Figure 66. Testing SNMP on slackware

- We verify that port 161 is open

```

root@andrea:~# netstat -tuln | grep 161
      udp        0      0 0.0.0.0:161                    0.0.0.0:*
root@andrea:~# 

```

Figure 67. Verifying port 161 is open

- Now we install the Zabbix monitoring software from the official website and transfer it to the Slackware machine using Samba
-



Figure 68. Downloading zabbix.tar.gz file

- We decompress the file

```
root@andrea:~# tar -zxvf zabbix-7.0.5.tar.gz
```

Figure 69. Command to decompress the .tar.gz file

```

zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_get_raid_sda_areca_2.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_scan_basic.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_get_raid_sda_cciss_0.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_get_raid_sda_areca_1.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_get_raid_sda_cciss_1.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_get_basic_csmi0_0.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs/env1/env_1_scan_raid.json
zabbix-7.0.5/src/go/plugins/smart/mock/outputs.go
zabbix-7.0.5/src/go/plugins/smart/smartfs.go
zabbix-7.0.5/src/go/plugins/smart/controller.go
zabbix-7.0.5/src/go/plugins/smart/smart_test.go
zabbix-7.0.5/src/go/plugins/smart/smart.go
zabbix-7.0.5/src/go/plugins/log/
zabbix-7.0.5/src/go/plugins/log/log.go
zabbix-7.0.5/src/go/Makefile.in
zabbix-7.0.5/src/go/LICENSE
zabbix-7.0.5/src/go/conf/
zabbix-7.0.5/src/go/conf/zabbix_web_service.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.win.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/docker.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/oracle.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/ceph.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/smart.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/mqtt.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/mysql.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/redis.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/modbus.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.d/plugins.d/memcached.conf
zabbix-7.0.5/src/go/conf/zabbix_agent2.conf
zabbix-7.0.5/src/go/conf/mock_server.conf
zabbix-7.0.5/src/go/go.mod
zabbix-7.0.5/src/go/Makefile.am
zabbix-7.0.5/src/go/go.sum
zabbix-7.0.5/src/go/bin/
root@andrea:~#

```

Figure 70. Decompressing zabbix.tar.gz file

- We must create a user Zabbix and a related group, we use **groupadd** and **useradd** commands

```

root@andrea:~# groupadd -g 200 zabbix
root@andrea:~# useradd -u 200 -g zabbix -d /usr/lib/zabbix -s /bin/false -c "Zabbix Monitoring System" zabbix
root@andrea:~# mkdir -m 777 -p /usr/lib/zabbix
root@andrea:~# zabbix:zabbix /usr/lib/zabbix
-bash: zabbix:zabbix: command not found
root@andrea:~# chown zabbix:zabbix /usr/lib/zabbix
root@andrea:~#

```

Figure 71. Configuring user for Zabbix

- Zabbix requires a database to store all the monitoring information, so with the PostgreSQL we installed in previous labs, we create the user and the database for it. Then we exit the database

```

andrea login: postgres
Password:
Last login: Mon Oct 14 03:08:48 on tty1
Linux 5.15.19.
postgres@andrea:~$ psql
psql (15.3)
Type "help" for help.

postgres=# CREATE USER zabbix WITH PASSWORD 'rec0.79.';
CREATE ROLE
postgres=# CREATE DATABASE zabbix OWNER zabbix;
CREATE DATABASE
postgres=# GRANT ALL PRIVILEGES ON DATABASE zabbix TO zabbix;
GRANT
postgres=#

```

Figure 72. Creating database for Zabbix

- We navigate to the directory generated after decompressing the file and configure using "../configure --enable-server --enable-agent --with-postgresql --with-net-snmp --with-lubcurl --with-libxml2"

```

root@andrea:~# cd zabbix-7.0.5
root@andrea:~/zabbix-7.0.5# ./configure --enable-server --enable-agent --with-postgresql --with-net-snmp --with-lubcurl --with-libxml2_

```

Figure 73. Configure Zabbix command

```
Detected OS:          linux-gnu
Install path:        /usr/local
Compilation arch:    linux

Compiler:            cc
Compiler flags:      -g -O2

Library-specific flags:
libXML2:             -I/usr/include/libxml2

Enable server:       no
Enable proxy:        no
Enable agent:        no
Enable agent 2:      no
Enable web service: no
Enable Java gateway: no

LDAP support:        no
IPv6 support:        no
cmocka support:      no
yaml support:        no

*****
*           Now run 'make install' *
*
*           Thank you for using Zabbix!
*           <https://www.zabbix.com>
*****
```

root@andrea:~/zabbix-7.0.5# _

Figure 74. Configuring Zabbix

- Now we compile using the “**make**” command

```

make[5]: Entering directory '/root/zabbix-7.0.5/database/postgresql/timescaledb/option-patches/without-compression'
make[5]: Nothing to be done for 'all'.
make[5]: Leaving directory '/root/zabbix-7.0.5/database/postgresql/timescaledb/option-patches/without-compression'
make[5]: Entering directory '/root/zabbix-7.0.5/database/postgresql/timescaledb/option-patches'
make[5]: Nothing to be done for 'all-am'.
make[5]: Leaving directory '/root/zabbix-7.0.5/database/postgresql/timescaledb/option-patches'
make[4]: Leaving directory '/root/zabbix-7.0.5/database/postgresql/timescaledb/option-patches'
make[4]: Entering directory '/root/zabbix-7.0.5/database/postgresql/timescaledb'
make[4]: Nothing to be done for 'all-am'.
make[4]: Leaving directory '/root/zabbix-7.0.5/database/postgresql/timescaledb'
make[3]: Leaving directory '/root/zabbix-7.0.5/database/postgresql/timescaledb'
make[3]: Entering directory '/root/zabbix-7.0.5/database/postgresql'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/root/zabbix-7.0.5/database/postgresql'
make[2]: Leaving directory '/root/zabbix-7.0.5/database/postgresql'
Making all in sqlite3
make[2]: Entering directory '/root/zabbix-7.0.5/database/sqlite3'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/root/zabbix-7.0.5/database/sqlite3'
make[2]: Entering directory '/root/zabbix-7.0.5/database'
make[2]: Nothing to be done for 'all-am'.
make[2]: Leaving directory '/root/zabbix-7.0.5/database'
make[1]: Leaving directory '/root/zabbix-7.0.5/database'
Making all in man
make[1]: Entering directory '/root/zabbix-7.0.5/man'
make[1]: Nothing to be done for 'all'.
make[1]: Leaving directory '/root/zabbix-7.0.5/man'
Making all in misc
make[1]: Entering directory '/root/zabbix-7.0.5/misc'
make[1]: Nothing to be done for 'all'.
make[1]: Leaving directory '/root/zabbix-7.0.5/misc'
make[1]: Entering directory '/root/zabbix-7.0.5'
make[1]: Nothing to be done for 'all-am'.
make[1]: Leaving directory '/root/zabbix-7.0.5'
root@andrea:~/zabbix-7.0.5#

```

Figure 75. Compiling Zabbix

- We finalize the installation using “**make install**”

```

make[4]: Leaving directory '/root/zabbix-7.0.5/database/postgresql'
make[3]: Leaving directory '/root/zabbix-7.0.5/database/postgresql'
make[2]: Leaving directory '/root/zabbix-7.0.5/database/postgresql'
Making install in sqlite3
make[2]: Entering directory '/root/zabbix-7.0.5/database/sqlite3'
make[3]: Entering directory '/root/zabbix-7.0.5/database/sqlite3'
make[3]: Nothing to be done for 'install-exec-am'.
make[3]: Nothing to be done for 'install-data-am'.
make[3]: Leaving directory '/root/zabbix-7.0.5/database/sqlite3'
make[2]: Leaving directory '/root/zabbix-7.0.5/database/sqlite3'
make[2]: Entering directory '/root/zabbix-7.0.5/database'
make[3]: Entering directory '/root/zabbix-7.0.5/database'
make[3]: Nothing to be done for 'install-exec-am'.
make[3]: Nothing to be done for 'install-data-am'.
make[3]: Leaving directory '/root/zabbix-7.0.5/database'
make[2]: Leaving directory '/root/zabbix-7.0.5/database'
make[1]: Leaving directory '/root/zabbix-7.0.5/database'
Making install in man
make[1]: Entering directory '/root/zabbix-7.0.5/man'
make[2]: Entering directory '/root/zabbix-7.0.5/man'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Leaving directory '/root/zabbix-7.0.5/man'
make[1]: Leaving directory '/root/zabbix-7.0.5/man'
Making install in misc
make[1]: Entering directory '/root/zabbix-7.0.5/misc'
make[2]: Entering directory '/root/zabbix-7.0.5/misc'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/root/zabbix-7.0.5/misc'
make[1]: Leaving directory '/root/zabbix-7.0.5/misc'
make[1]: Entering directory '/root/zabbix-7.0.5'
make[2]: Entering directory '/root/zabbix-7.0.5'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/root/zabbix-7.0.5'
make[1]: Leaving directory '/root/zabbix-7.0.5'
root@andrea:~/zabbix-7.0.5#

```

Figure 76. Installing Zabbix

- This creates a folder where all the SQL scripts needed by Zabbix are stored in /database/postgresql

```

root@andrea:~/zabbix/zabbix-7.0.5# cd database
root@andrea:~/zabbix/zabbix-7.0.5/database# ls
Makefile Makefile.am Makefile.in elasticsearch/ mysql/ oracle/ postgresql/ sqlite3/
root@andrea:~/zabbix/zabbix-7.0.5/database# cd postgresql
root@andrea:~/zabbix/zabbix-7.0.5/database/postgresql# ls
Makefile Makefile.am Makefile.in data.sql images.sql option-patches/ schema.sql timescaledb/
root@andrea:~/zabbix/zabbix-7.0.5/database/postgresql# _

```

Figure 77. Zabbix SQL files

- We log in again with our postgres user (**su - postgres**) and use the command **psql -U zabbix -d zabbix -f /path/to/sql/files** to copy the SQL files that were generated into the database we created for Zabbix

```

postgres@andrea:~$ psql -U zabbix -d zabbix -f /root/zabbix/zabbix-7.0.5/database/postgresql/schema.sql

```

Figure 78. Moving the schema.sql file into the database

- When inserting the file, we can observe that several tables were altered

```
ALTER TABLE
```

Figure 79. Modification of the Zabbix database after inserting schema.sql

- We follow the same procedure for images.sql and data.sql
- ```
postgres@andrea:~$ psql -U zabbix -d zabbix -f /root/zabbix/zabbix-7.0.5/database/postgresql/images.sql
```

*Figure 80. Moving the images.sql file into the database*

```
INSERT 0 1
postgres@andrea:~$ _
```

Figure 81. Modification of the Zabbix database after inserting images.sql

```
postgres@andrea:~$ psql -U zabbix -d zabbix -f /root/zabbix/zabbix-7.0.5/database/postgresql/data.sql
```

Figure 82. Moving the data.sql file into the database

```
INSERT 0 1
DELETE 86506
COMMIT
postgres@andrea:~$
```

Figure 83. Modification of the Zabbix database after inserting data.sql

○ We open the zabbix\_server.conf file to configure the database name and password  
`zabbix@andrea:/zabbix/zabbix-7.0.5/conf# nano zabbix_server.conf`

Figure 84. Opening zabbix\_server.conf

- In the DBUser and DBName fields, we write the user and database we created earlier

```

GNU nano 6.0 zabbix_server.conf Modified
#
Mandatory: no
Default:
DBHost=localhost

Option: DBName
Database name.
If the Net Service Name connection method is used to connect to Oracle database, specify the
tnsnames.ora file or set to empty string; also see the TWO_TASK environment variable if>
empty string.
#
Mandatory: yes
Default:
DBName=

DBName=zabbix

Option: DBSchema
Schema name. Used for PostgreSQL.
#
Mandatory: no
Default:
DBSchema=

Option: DBUser
Database user.
#
Mandatory: no
Default:
DBUser=

DBUser=zabbix

[G] Help [O] Write Out [W] Where Is [X] Cut [T] Execute [C] Location [U-U] Undo
[X] Exit [R] Read File [R] Replace [U] Paste [J] Justify [-] Go To Line [U-E] Redo

```

Figure 85. Configuring database and username into zabbix\_server.conf

- Finally, in the DBPassword field, we assign a password

```

GNU nano 6.0 zabbix_server.conf Modified
empty string.
#
Mandatory: yes
Default:
DBName=

DBName=zabbix

Option: DBSchema
Schema name. Used for PostgreSQL.
#
Mandatory: no
Default:
DBSchema=

Option: DBUser
Database user.
#
Mandatory: no
Default:
DBUser=

DBUser=zabbix

Option: DBPassword
Database password.
Comment this line if no password is used.
#
Mandatory: no
Default:
DBPassword=rec0.79.

Option: DBSocket
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location ^U Undo
^X Exit ^R Read File ^H Replace ^U Paste ^J Justify ^L Go To Line ^E Redo

```

Figure 86. Assigning a password for the Zabbix database

- Now we open the **zabbix\_agentd.conf** file to configure the machines that Zabbix will monitor, and we write the IP of the Slackware machine

```
root@andrea:~/zabbix/zabbix-7.0.5/conf# nano zabbix_agentd.conf
```

Figure 87. Opening zabbix\_agentd.conf

```

GNU nano 6.0 zabbix_agentd.conf Modified

Option: Server
List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of Zabbix servers.
Incoming connections will be accepted only from the hosts listed here.
If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated as valid.
and '::/0' will allow any IPv4 or IPv6 address.
'0.0.0.0/0' can be used to allow any IPv4 address.
Example: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
#
Mandatory: yes, if StartAgents is not explicitly set to 0
Default:
Server=

Server=192.168.20.100

Option: ListenPort
Agent will listen on this port for connections from the server.
#
Mandatory: no
Range: 1024-32767
Default:
ListenPort=10050

Option: ListenIP
List of comma delimited IP addresses that the agent should listen on.
First IP address is sent to Zabbix server if connecting to it to retrieve list of active checks.
#
Mandatory: no
Default:
ListenIP=0.0.0.0

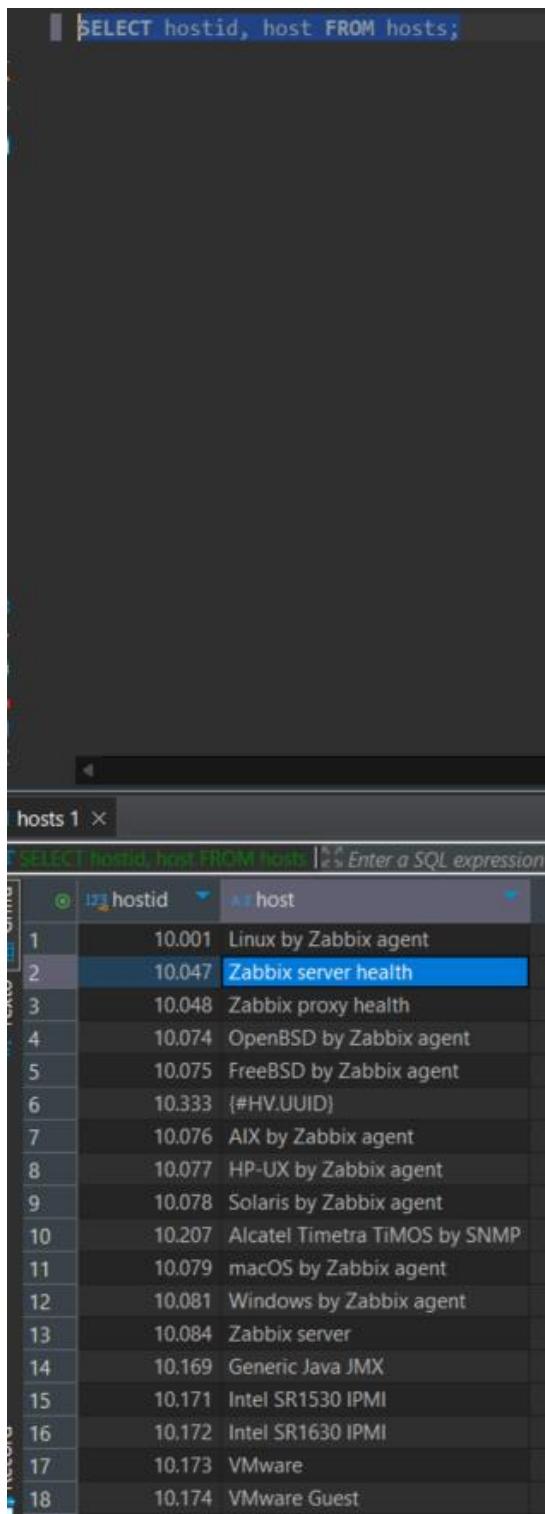
Option: StartAgents
Number of pre-forked instances of zabbix_agentd that process passive checks.

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^N Replace ^U Paste ^J Justify ^- Go To Line M-E Redo

```

Figure 88. configuring Slackware machine that Zabbix will monitor

- For better visualization, we open DBeaver and connect to the database with the credentials we configured in *Figure 77*. Then, we run the following query to verify that the Zabbix database is properly configured and to check the hosts it contains



The screenshot shows a MySQL command-line interface window. At the top, there is a status bar with the text "hostid host". Below it, a query is being typed into the command line:

```
SELECT hostid, host FROM hosts;
```

Below the command line, the results of the query are displayed in a table:

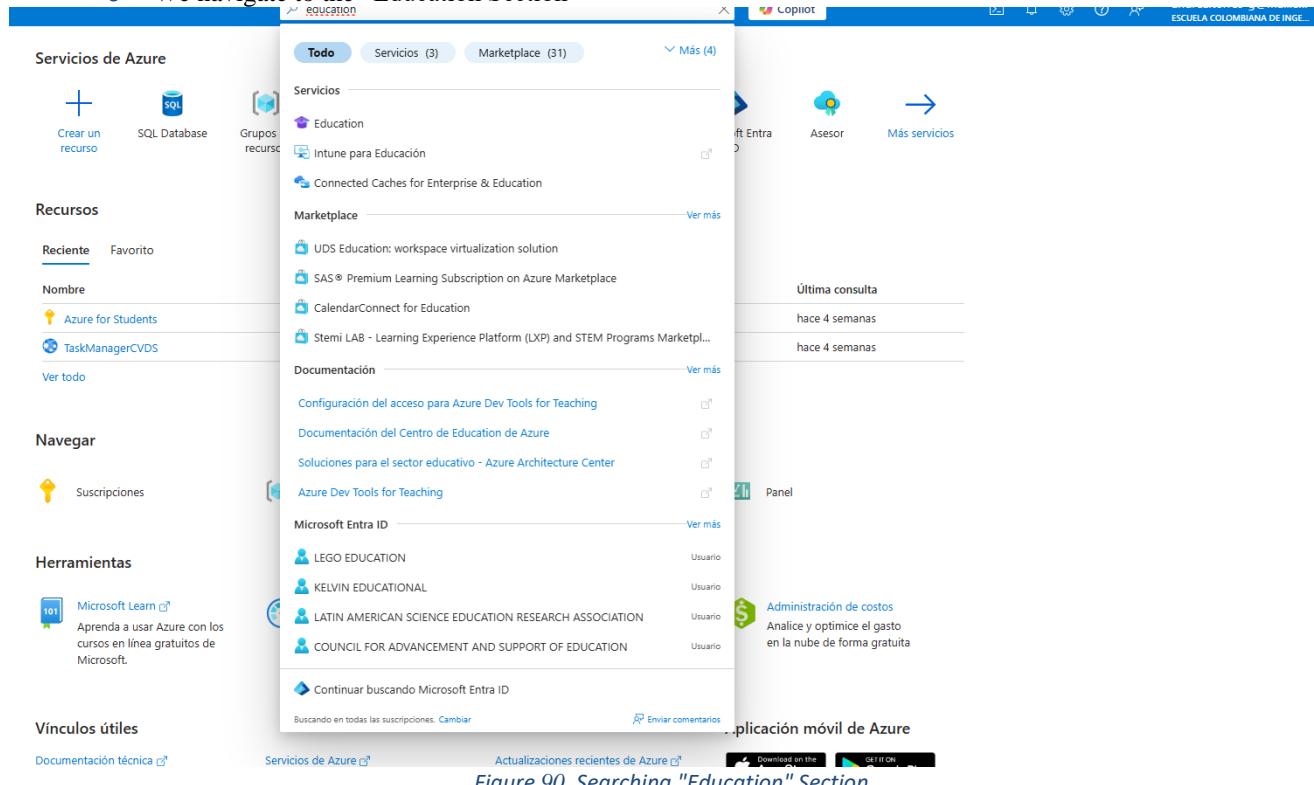
| hostid | host                                 |
|--------|--------------------------------------|
| 1      | 10.001 Linux by Zabbix agent         |
| 2      | <b>Zabbix server health</b>          |
| 3      | 10.048 Zabbix proxy health           |
| 4      | 10.074 OpenBSD by Zabbix agent       |
| 5      | 10.075 FreeBSD by Zabbix agent       |
| 6      | 10.333 {#HV.UUID}                    |
| 7      | 10.076 AIX by Zabbix agent           |
| 8      | 10.077 HP-UX by Zabbix agent         |
| 9      | 10.078 Solaris by Zabbix agent       |
| 10     | 10.207 Alcatel Timetra TiMOS by SNMP |
| 11     | 10.079 macOS by Zabbix agent         |
| 12     | 10.081 Windows by Zabbix agent       |
| 13     | 10.084 Zabbix server                 |
| 14     | 10.169 Generic Java JMX              |
| 15     | 10.171 Intel SR1530 IPMI             |
| 16     | 10.172 Intel SR1630 IPMI             |
| 17     | 10.173 VMware                        |
| 18     | 10.174 VMware Guest                  |

Figure 89. Executing a query on the Zabbix database

### 3. Network Administration – Azure

#### 3.1. Configuration of the web application service

- We navigate to the “Education Section”

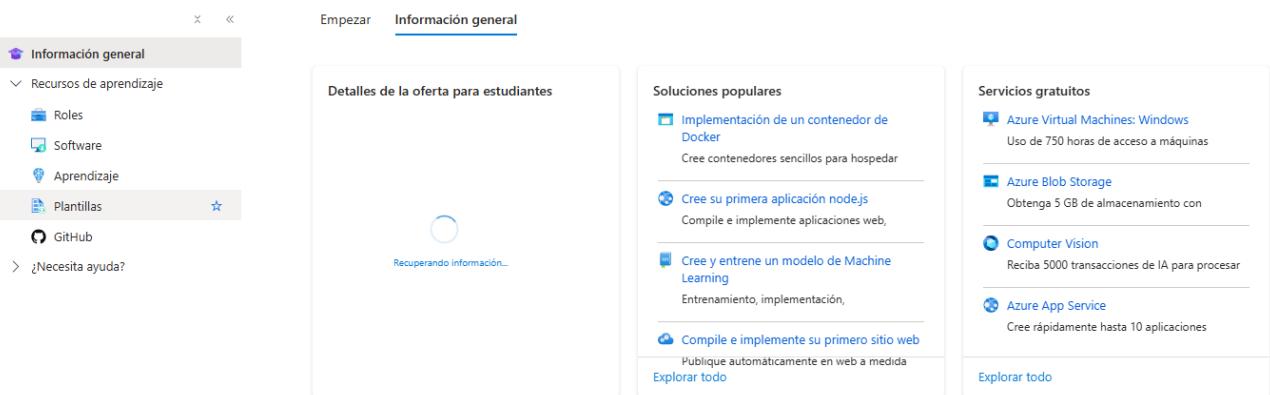


The screenshot shows the Azure portal interface with a search bar at the top containing the text "Education". The search results are displayed in a grid format. The first result is "Education" with a subtitle "Intune para Educación". Below it is "Connected Caches for Enterprise & Education". Under the "Marketplace" section, there are several items including "UDS Education: workspace virtualization solution", "SAS® Premium Learning Subscription on Azure Marketplace", and "CalendarConnect for Education". The "Documentación" section contains links to "Configuración del acceso para Azure Dev Tools for Teaching" and "Documentación del Centro de Educación de Azure". The "Última consulta" section shows two entries: "hace 4 semanas" and "hace 4 semanas". On the left sidebar, under "Navegar", there is a "Suscripciones" section with "Azure for Students" selected. Under "Herramientas", there is a "Microsoft Learn" section with "Aprenda a usar Azure con los cursos en línea gratuitos de Microsoft.". At the bottom, there is a "Vínculos útiles" section with a link to "Documentación técnica".

Figure 90. Searching "Education" Section

- We navigate to the “Templates” Section located in the left menu

 Education | Información general



The screenshot shows the "Información general" template page. The left sidebar has a "Plantillas" section with "GitHub" selected. The main content area has a "Detalles de la oferta para estudiantes" section with a "Recuperando información..." message. To the right, there are three columns: "Soluciones populares" with items like "Implementación de un contenedor de Docker", "Cree su primera aplicación node.js", "Cree y entrene un modelo de Machine Learning", and "Compile e implemente su primer sitio web"; "Servicios gratuitos" with items like "Azure Virtual Machines: Windows", "Azure Blob Storage", "Computer Vision", and "Azure App Service"; and a "Explorar todo" button.

Figure 91. Navigating to "Templates" Section

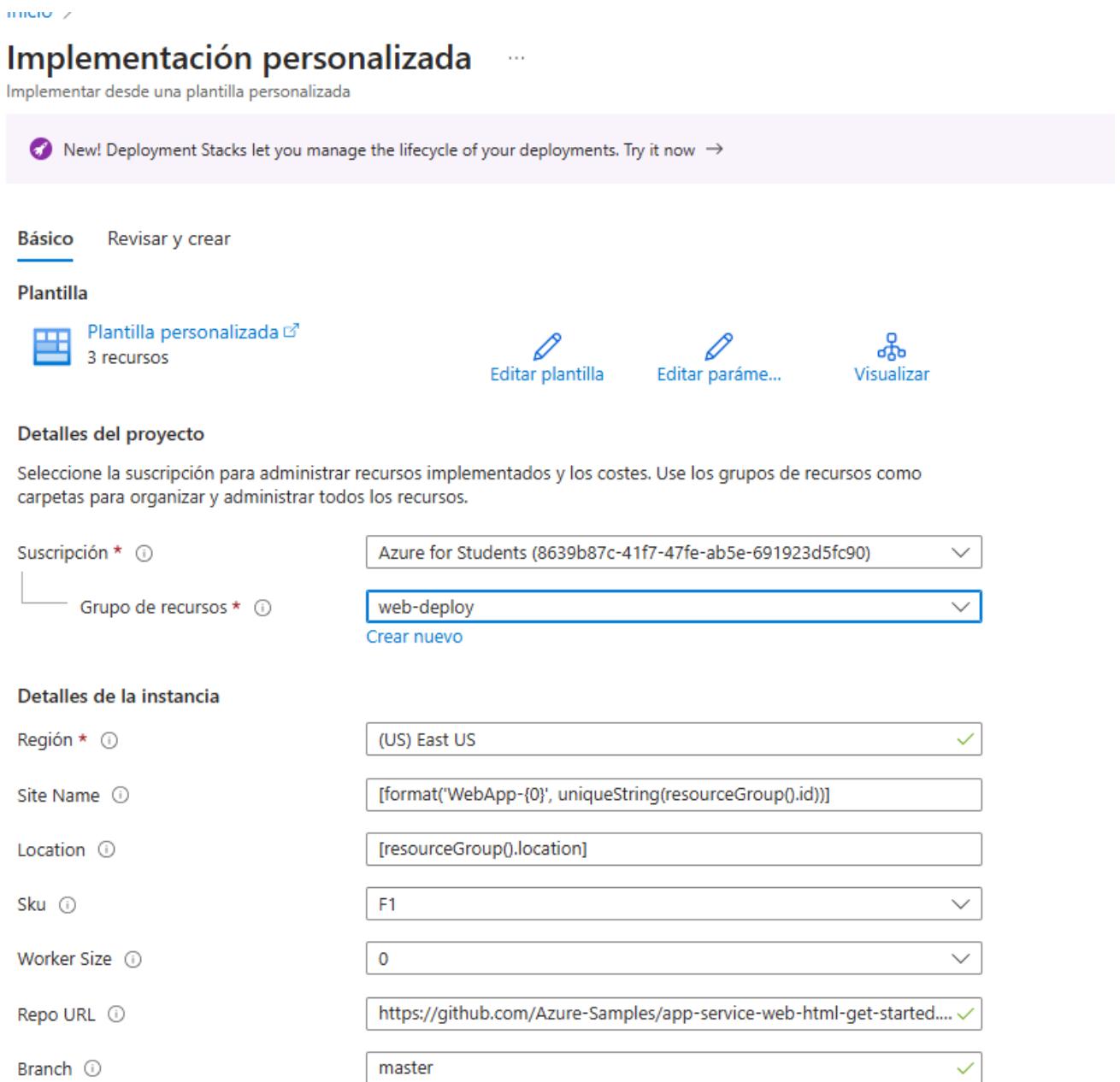
- We search for “Web app deployment from GitHub” and click on it

**Plantillas de inicio rápido de Azure**  
 Implemente recursos de Azure a través de Azure Resource Manager con las plantillas aportadas de la comunidad. Implemente, aprenda y obtenga más. [Más información](#)

| <input type="text" value="web app deployment"/>                                                                                                                                                                                                                                                           | Escenario : <b>Todo</b> | Dificultad : <b>Todo</b> | Costo : <b>Todo</b> | Roles : <b>Todo</b> |           |              |               |          |          |                                |                   |              |           |        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|--------------------------|---------------------|---------------------|-----------|--------------|---------------|----------|----------|--------------------------------|-------------------|--------------|-----------|--------|
| <b>1 Elemento</b>                                                                                                                                                                                                                                                                                         |                         |                          |                     |                     |           |              |               |          |          |                                |                   |              |           |        |
| <table border="1"> <thead> <tr> <th>Nombre ↑↓</th><th>Escenario ↑↓</th><th>Dificultad ↑↓</th><th>Roles ↑↓</th><th>Costo ↑↓</th></tr> </thead> <tbody> <tr> <td>Web App Deployment from GitHub</td><td>Azure App Service</td><td>Intermediate</td><td>Developer</td><td>GRATIS</td></tr> </tbody> </table> |                         |                          |                     |                     | Nombre ↑↓ | Escenario ↑↓ | Dificultad ↑↓ | Roles ↑↓ | Costo ↑↓ | Web App Deployment from GitHub | Azure App Service | Intermediate | Developer | GRATIS |
| Nombre ↑↓                                                                                                                                                                                                                                                                                                 | Escenario ↑↓            | Dificultad ↑↓            | Roles ↑↓            | Costo ↑↓            |           |              |               |          |          |                                |                   |              |           |        |
| Web App Deployment from GitHub                                                                                                                                                                                                                                                                            | Azure App Service       | Intermediate             | Developer           | GRATIS              |           |              |               |          |          |                                |                   |              |           |        |

Figure 92. Selecting "Web App Deployment from GitHub" template

- We create a resource group and leave the default values as they are



The screenshot shows the 'Implementación personalizada' (Custom Deployment) blade in the Azure portal. It includes sections for 'Básico' (Basic), 'Plantilla' (Template), 'Detalles del proyecto' (Project Details), 'Detalles de la instancia' (Instance Details), and 'Resumen' (Summary). Key details include:

- Básico:** Revisar y crear
- Plantilla:** Plantilla personalizada (Custom Template) - 3 recursos
- Detalles del proyecto:**
  - Suscripción: Azure for Students (8639b87c-41f7-47fe-ab5e-691923d5fc90)
  - Grupo de recursos: web-deploy
- Detalles de la instancia:**
  - Región: (US) East US
  - Site Name: [format('WebApp-{0}', uniqueString(resourceGroup().id))]
  - Location: [resourceGroup().location]
  - Sku: F1
  - Worker Size: 0
  - Repo URL: <https://github.com/Azure-Samples/app-service-web-html-get-started....>
  - Branch: master

Figure 93. Deploying web service

- Click on “Review + create” and subsequently “Create.”

## Implementación personalizada

...

Implementar desde una plantilla personalizada

Básico    Revisar y crear

### Resumen



Plantilla personalizada

3 recursos

### Términos

[Términos de Azure Marketplace](#) | [Azure Marketplace](#)

Al hacer clic en "Crear", (a) acepto los términos legales aplicables asociados con la oferta; (b) autorizo a Microsoft a que me cobre o facture mediante mi método de pago actual las cuotas asociadas con las ofertas, incluidos los impuestos correspondientes y con la misma frecuencia de facturación que mi suscripción a Azure, hasta que decida interrumpir el uso de estas ofertas; y (c) acepto que, si la implementación implica ofertas de terceros, Microsoft puede compartir mi información de contacto y otros detalles de dicha implementación con el editor de esa oferta.

Microsoft no asume ninguna responsabilidad por las acciones realizadas por plantillas de terceros ni proporciona los derechos de los productos o servicios de terceros. Consulte los [Términos de Azure Marketplace](#) para obtener términos adicionales.

La implementación de esta plantilla creará uno o más recursos de Azure u ofertas de Marketplace. Reconoce que es responsable de revisar los precios aplicables y los términos legales aplicables asociados con todos los recursos y ofertas implementados como parte de esta plantilla. Los precios y los términos legales asociados de cualquier oferta de Marketplace pueden encontrarse en [Azure Marketplace](#); ambos están sujetos a cambios en cualquier momento antes de su implementación.

No se pueden usar los créditos de la suscripción ni los fondos del compromiso monetario para comprar ofertas que no sean de Microsoft. Estas compras se facturan a parte.

Si se incluye cualquier producto de Microsoft en una oferta de Marketplace (p. ej., Windows Server o SQL Server), dichos productos tendrán licencia de Microsoft y no de terceros.

### Básico

Suscripción                      Azure for Students

Grupo de recursos              web-deploy

Región                            East US

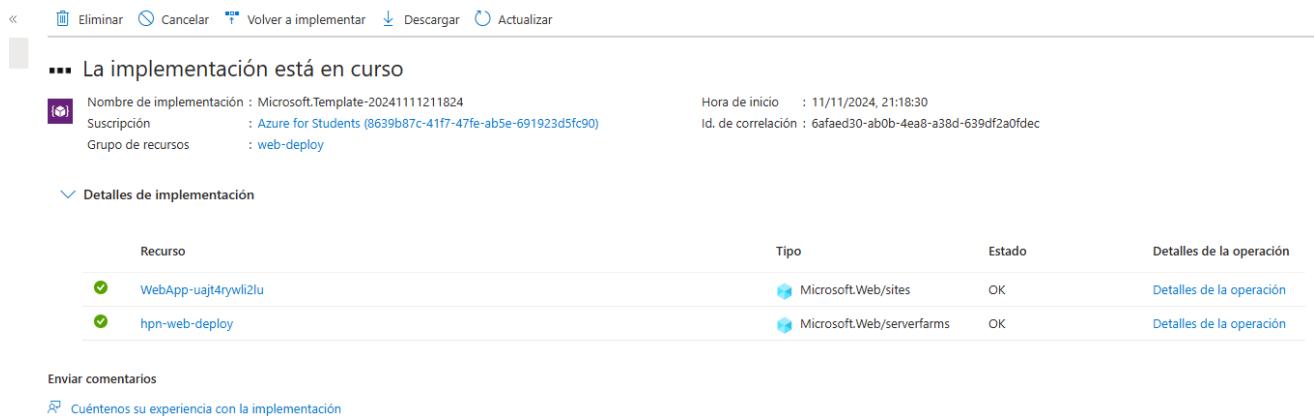
< Anterior

Siguiente

**Crear**

Figure 94. Verifying deployment

- We wait for the deployment to be completed



La implementación está en curso

Nombre de implementación : Microsoft.Template-2024111211824  
 Suscripción : Azure for Students (8639b87c-41f7-47fe-ab5e-691923d5fc90)  
 Grupo de recursos : web-deploy

Hora de inicio : 11/11/2024, 21:18:30  
 Id. de correlación : 6afaed30-ab0b-4ea8-a38d-639df2a0fdec

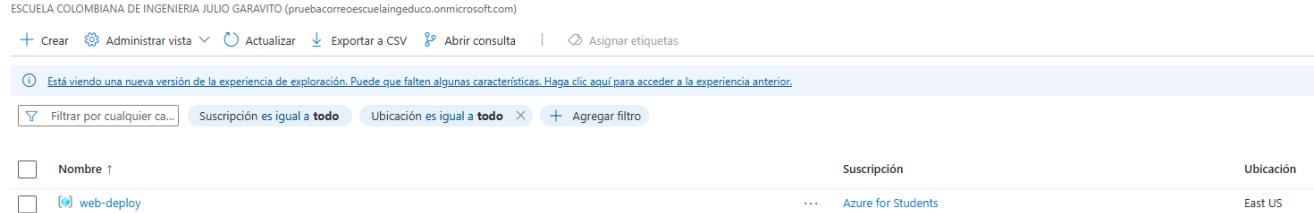
**Detalles de implementación**

| Recurso              | Tipo                      | Estado | Detalles de la operación                 |
|----------------------|---------------------------|--------|------------------------------------------|
| WebApp-ujjt4rywli2lu | Microsoft.Web/sites       | OK     | <a href="#">Detalles de la operación</a> |
| hpn-web-deploy       | Microsoft.Web/serverfarms | OK     | <a href="#">Detalles de la operación</a> |

Enviar comentarios  
[Cuéntenos su experiencia con la implementación](#)

Figure 95. Finalizing deployment

- We navigate to the created resource and access the “Web App”
- Grupos de recursos** ...



ESCUOLA COLOMBIANA DE INGENIERÍA JULIO GARAVITO (pruebacorreoescuelaing.edu.co.onmicrosoft.com)

+ Crear   Administrar vista   Actualizar   Exportar a CSV   Abrir consulta   Asignar etiquetas

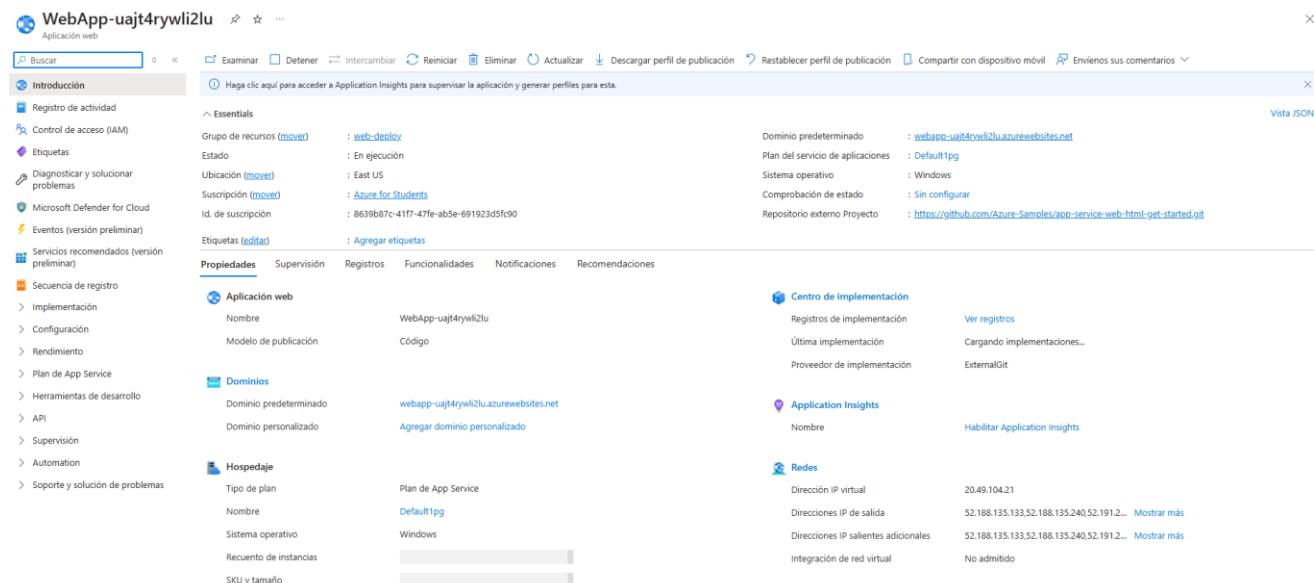
Está viendo una nueva versión de la experiencia de exploración. Puede que falten algunas características. Haga clic aquí para acceder a la experiencia anterior.

Filtrar por cualquier campo   Suscripción es igual a todo   Ubicación es igual a todo   Agregar filtro

| Nombre     | Suscripción        | Ubicación |
|------------|--------------------|-----------|
| web-deploy | Azure for Students | East US   |

Figure 96. Navigating to Web App resource

- We explore the website by clicking the “Default domain” at the “Overview” section



**WebApp-ujjt4rywli2lu** Aplicación web

Introducción Registro de actividad Control de acceso (IAM) Diagnóstico y solucionar problemas Microsoft Defender for Cloud Eventos (versión preliminar) Servicios recomendados (versión preliminar) Secuencia de registro Implementación Configuración Rendimiento Plan de App Service Herramientas de desarrollo API Supervisión Automatización Soporte y solución de problemas

Essentials Grupo de recursos (mover) : web-deploy Estado : En ejecución Ubicación (mover) : East US Suscripción (mover) : Azure for Students Id. de suscripción : 8639b87c-41f7-47fe-ab5e-691923d5fc90 Etiquetas (editar) : Agregar etiquetas

Aplicación web Nombre : WebApp-ujjt4rywli2lu Modelo de publicación : Código

Dominios Dominio predeterminado : webapp-ujjt4rywli2lu.azurewebsites.net Dominio personalizado : Agregar dominio personalizado

Hospedaje Tipo de plan : Plan de App Service Nombre : Default1p9 Sistema operativo : Windows Recuento de instancias : SKU y tamaño :

Centro de implementación Registros de implementación : Ver registros Última implementación : Cargando implementaciones... Proveedor de implementación : ExternalGit

Application Insights Nombre : Habilitar Application insights

Redes Dirección IP virtual : 20.49.104.21 Direcciones IP de salida : 52.188.135.133.52.188.135.240.52.191.2... Mostrar más Direcciones IP salientes adicionales : 52.188.135.133.52.188.135.240.52.191.2... Mostrar más Integración de red virtual : No admitido

Figure 97. Exploring the web application

- We can see the website that was deployed

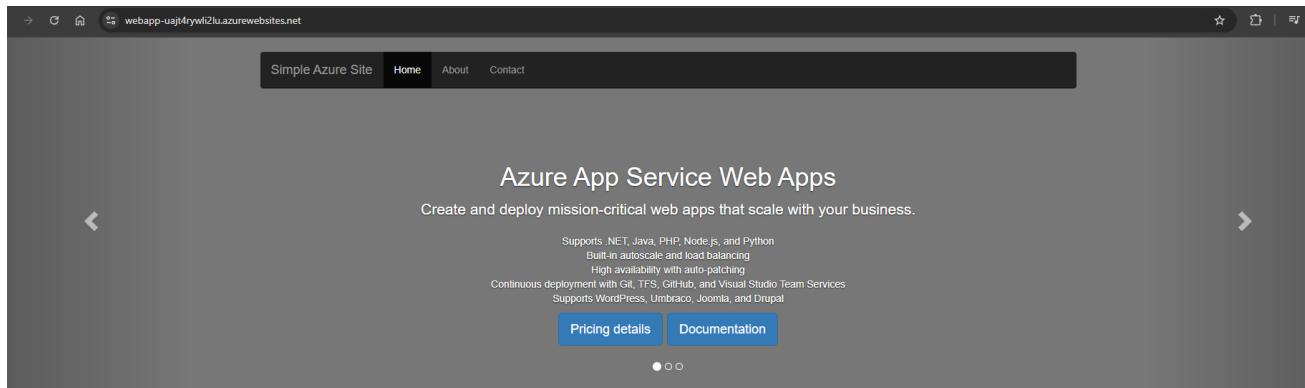


Figure 98. Verifying that Web Service is working

- Now, we go to the “**Monitoring**” section located in the left menu, then navigate to “**Application Insights**”

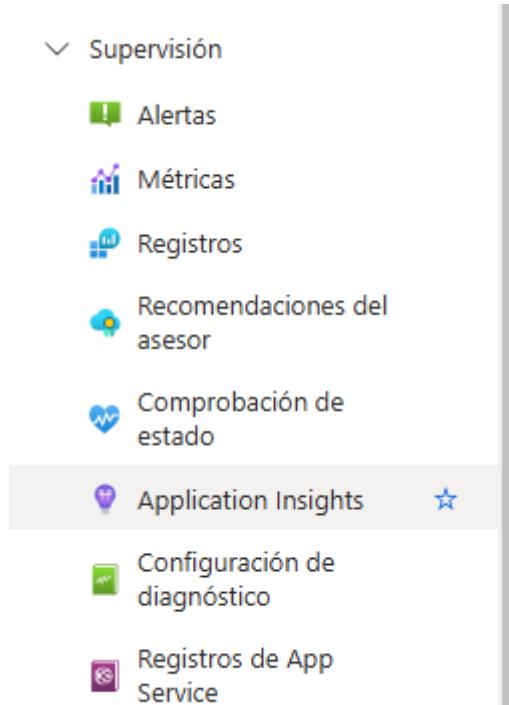


Figure 99. Navigating to Application Insights

- We enable the service clicking on “Enable Application Insights”

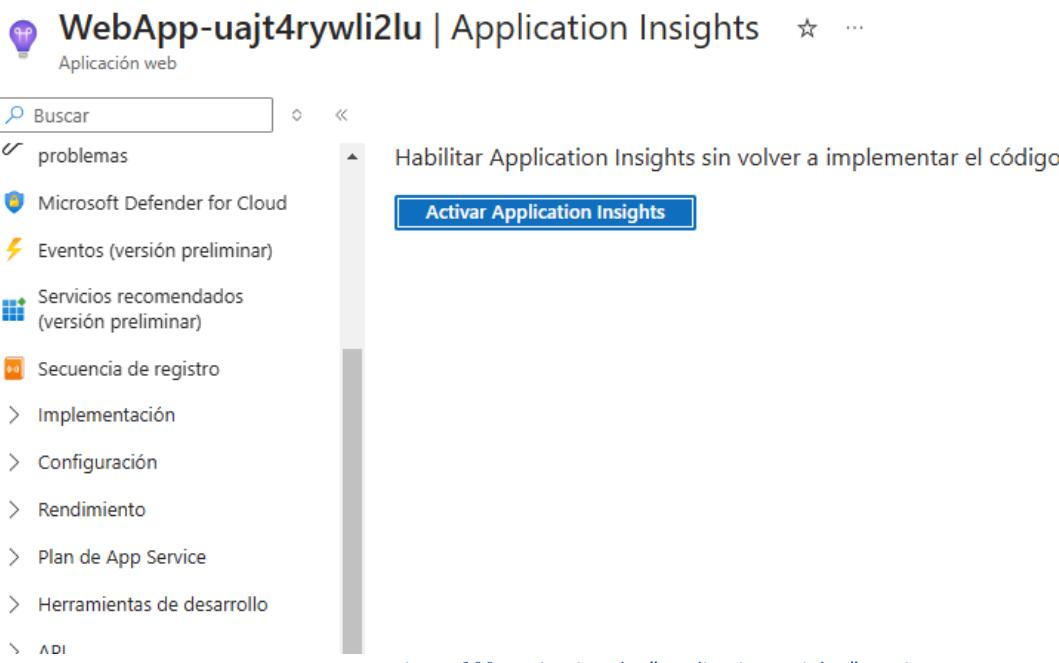


Figure 100. Activating the "Application Insights" service

- Once there, click on “Enable” and then click on “Apply”

Figure 101. Enabling the "Application Insights" service

- When we click on “Apply”, a confirmation window appears, click on “Yes”

## Application Insights

Permite recopilar datos de supervisión de aplicaciones con Application Insights.

Habilitar

Deshabilitar



Comentarios ▾

### Vincular a un recurso de Application Insights

**i** La aplicación se conectará a un recurso de Application Insights creado automáticamente: **WebApp-uajt4ry**. Se agregará una clave de instrumentación a la configuración de la aplicación. Al hacerlo, se sobrescribirá cu...

**i** Como parte del uso de la instrumentación de Application Insights, recopilamos y enviamos datos de diagnóstico y recopilación de datos no esencial. [Más información](#)

^ Cambie su recurso

**○** Crear nuevo recurso

**i** Las conclusiones de la aplicación y los recursos del área de trabajo se crean en el ámbito de la suscripción. [Crear un recurso de Application Insights nuevo](#) y vuelva a esta página.

Nuevo nombre de recurso \*

WebApp-uajt4rywli2lu

### Aplicar la configuración de supervisión

Ahora vamos a aplicar cambios a la configuración de la aplicación e instalaremos nuestras herramientas para vincular el recurso de Application Insights a la aplicación web. Al hacerlo, se reiniciará el sitio. ¿Quiere continuar?

Sí

No

Aplicar

Figure 102. Verifying configuration of "Application Insights"

- Once there, click on “View Application Insights data”

Validation passed

Ver datos de Application Insights

### Application Insights

Permite recopilar datos de supervisión de aplicaciones con Application Insights.

Habilitar Deshabilitar

Comentarios

Vincular a un recurso de Application Insights

La aplicación está conectada al recurso de Application Insights: [WebApp-uj4t4rywli2lu](#)

Como parte del uso de la instrumentación de Application Insights, recopilamos y enviamos datos de diagnóstico a Microsoft. Estos datos nos ayudan a ejecutar y mejorar Application Insights. Tiene la opción de recopilación de datos no esencial. [Más información](#)

Cambie su recurso

### Instrumentar una aplicación

Información .NET .NET Core Node.js Java Python

Seleccione el idioma que eligió durante la creación de la aplicación para ver los detalles de instrumentación y las configuraciones adicionales si están disponibles.

Aplicar

Figure 103. Navigating to Application Insights data

- We can see the overview of the service

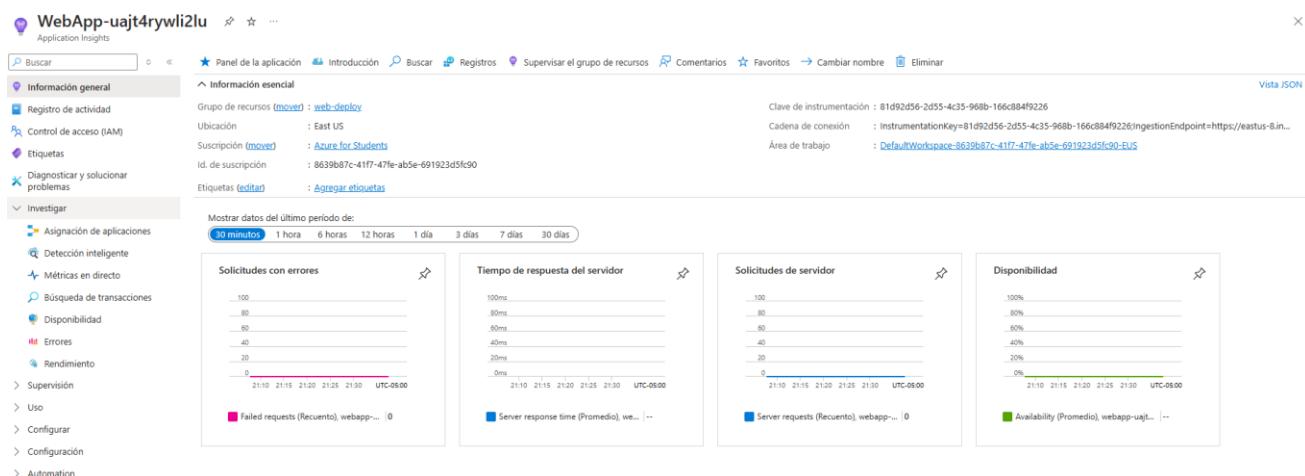


Figure 104. Overview of Application Insights

- In the left menu, we go to “Investigate” and select “Live Metric”. We can observe a real-time view of the performance and behavior of our application



Figure 105. Real-time view of the performance of the web service application

### 3.2. Questionnaire

1. Explore the "Overview" and "Live Metric" tabs in the "Investigate" section.
  - What do you observe when you refresh the website repeatedly?
- Each refresh generates new data, updating metrics in real time, like request rates, request duration, committed memory and CPU total.

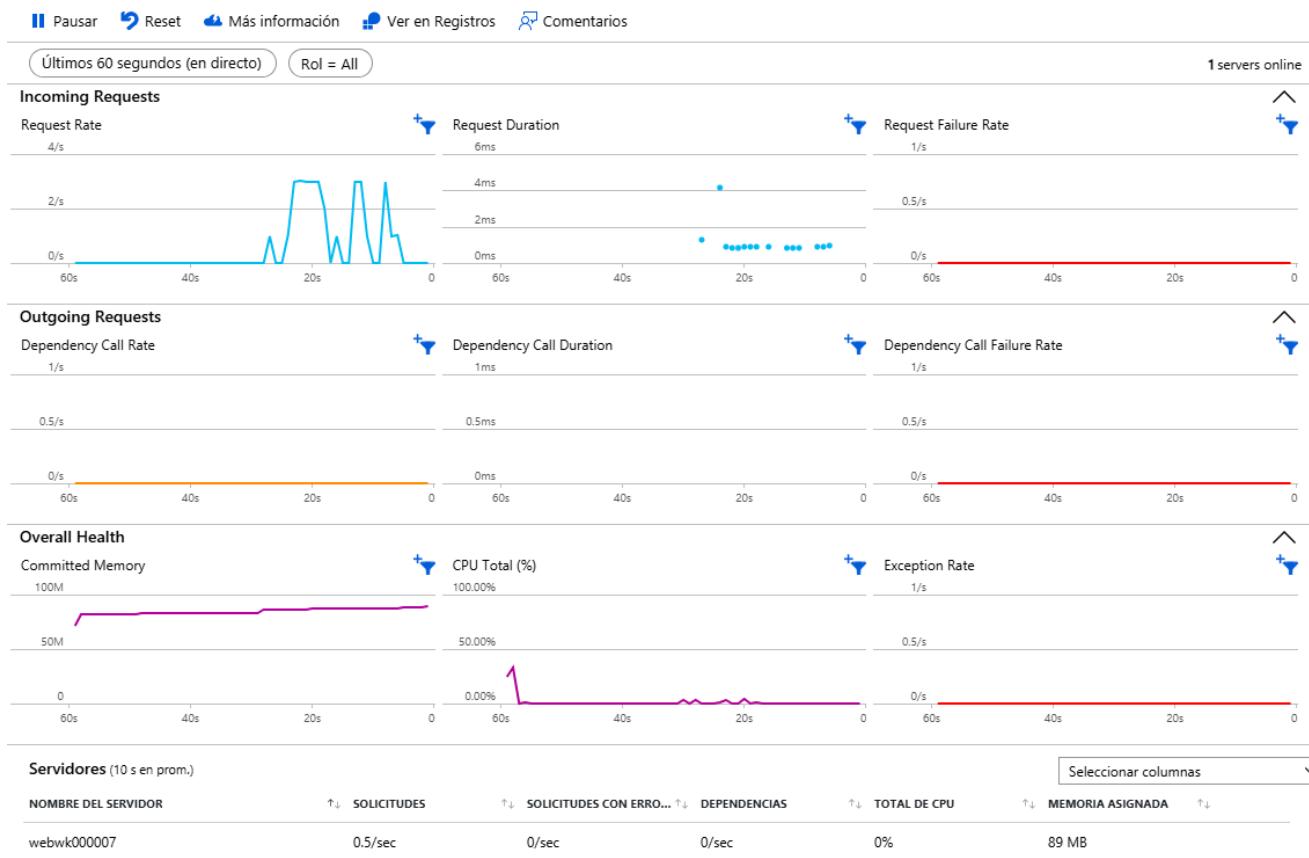


Figure 106. Metrics after refreshing the website repeatedly

- What does each of the items displayed there represent?
  - Incoming Request
    - Request Rate: Number of requests made to the app
    - Request Duration: Represents the average time the application takes to process each request
    - Request Failure Rate: Measures the percentage of requests that did not complete successfully
  - Outgoing Request
    - Dependency Call Rate: Indicates the number of requests the application makes to other external services, such as databases or third-party APIs
    - Dependency Call Duration: It is the average time taken for requests sent to external services to complete
    - Dependency Call Failure Rate: Shows the percentage of requests to external services that failed
  - Overall Health
    - Committed Memory: Represents the amount of memory that is allocated or committed to the application at a given moment
    - CPU Total (%): Indicates the percentage of CPU usage on the server or resource hosting the application
    - Exception Rate: It is the percentage of exceptions or unexpected errors that occur in the application
  - What other functionalities does the "Application Insights" system offer for web services, databases, and other cloud-deployed systems?
  - Application Insights provides many experiences to enhance the performance, reliability, and quality of our applications.
  - Investigate

- Application dashboard: An at-a-glance assessment of your application's health and performance.
- Application map: A visual overview of application architecture and components' interactions.
- Live metrics: A real-time analytics dashboard for insight into application activity and performance.
- Transaction search: Trace and diagnose transactions to identify issues and optimize performance.
- Availability view: Proactively monitor and test the availability and responsiveness of application endpoints.
- Failures view: Identify and analyze failures in your application to minimize downtime.
- Performance view: Review application performance metrics and potential bottlenecks.
- Monitoring
  - Alerts: Monitor a wide range of aspects of your application and trigger various actions.
  - Metrics: Dive deep into metrics data to understand usage patterns and trends.
  - Diagnostic settings: Configure streaming export of platform logs and metrics to the destination of your choice.
  - Logs: Retrieve, consolidate, and analyze all data collected into Azure Monitoring Logs.
  - Workbooks: Create interactive reports and dashboards that visualize application monitoring data.
- Usage
  - Users, sessions, and events: Determine when, where, and how users interact with your web app.
  - Funnels: Analyze conversion rates to identify where users progress or drop off in the funnel.
  - Flows: Visualize user paths on your site to identify high engagement areas and exit points.
  - Cohorts: Group users by shared characteristics to simplify trend identification, segmentation, and performance troubleshooting.
- Code analysis
  - Profiler: Capture, identify, and view performance traces for the application.
  - Code optimizations: Harness AI to create better and more efficient applications.
  - Snapshot debugger: Automatically collect debug snapshots when exceptions occur in .NET application
    - How might this functionality be beneficial in a corporate environment?
- It helps detect issues early, optimize resource usage, enhance user experience, and support proactive monitoring, which minimizes downtime and improves overall performance.
  - How does the network layer contribute to the transmission of data between the web app and clients in different geographic locations?
- The network layer handles IP addressing and routing, ensuring data packets are transmitted to the correct locations across various networks, optimizing speed and reliability based on geographic distance and network conditions.
  - In what ways do the application and transport layers ensure reliable data transfer and correct interpretation of web app responses?
    - **Transport Layer:** Uses protocols like TCP to ensure packets are delivered reliably and in the correct order. Error checking and retransmission requests are included.
    - **Application Layer:** Converts data to and from formats understandable by the application and user, ensuring data consistency and correct interpretation across devices.

## Experiments

Perform the following tests in groups or individually, as instructed by the professor, and document the experience.

### 1. Use of ICMP Messages

- a. We go to <https://traceroute-online.com/> and search for the Computer Science Laboratory page and the Stanford University page. Show the results:

o

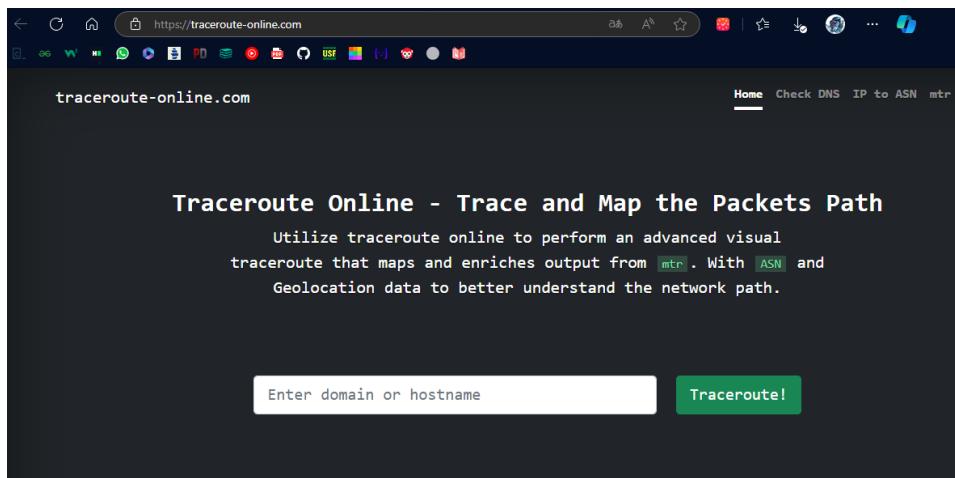


Figure 107 traceroute-online page

- o Then, we searched the Computer Science Laboratory page

o

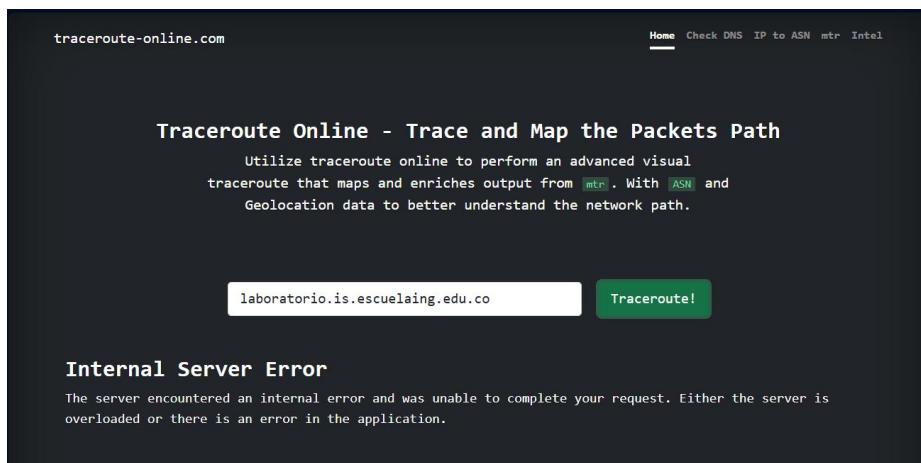
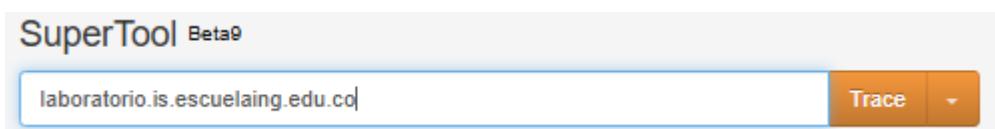


Figure 108 Search for the laboratory page

We can observe an internal server error, so we use the following page: MXToolbox.



| HopCount | IP Address                                                                 | HostName                                   |
|----------|----------------------------------------------------------------------------|--------------------------------------------|
| 1        | 10.140.10.147                                                              | ***                                        |
| 2        | 3.236.63.11<br>Amazon.com, Inc. (AS14618)                                  | ec2-3-236-63-11.compute-1.amazonaws.com    |
| 3        | ***                                                                        | ***                                        |
| 4        | ***                                                                        | ***                                        |
| 5        | ***                                                                        | ***                                        |
| 6        | ***                                                                        | ***                                        |
| 7        | 99.83.66.22                                                                | ***                                        |
| 8        | 99.83.69.241                                                               | ***                                        |
| 9        | 69.79.100.53<br>Columbus Networks USA, Inc. (AS23520)                      | ***                                        |
| 10       | 69.79.100.5<br>Columbus Networks USA, Inc. (AS23520)                       | ae2.brx-mx2020-2.boca-raton.fl.usa.cwc.com |
| 11       | 190.131.207.5<br>LIBERTY NETWORKS DE COLOMBIA S.A.S (AS262191)             | ***                                        |
| 12       | 190.131.207.183<br>LIBERTY NETWORKS DE COLOMBIA S.A.S (AS262191)           | ***                                        |
| 13       | ***                                                                        | ***                                        |
| 14       | 45.239.88.78<br>Unknown (AS266662)                                         | ***                                        |
| 15       | ***                                                                        | ***                                        |
| 16       | 45.239.88.78<br>ESCUOLA COLOMBIANA DE INGENIERIA JULIO GARAVITO (AS266662) | ***                                        |
| 17       | 45.239.88.88<br>ESCUOLA COLOMBIANA DE INGENIERIA JULIO GARAVITO (AS266662) | ***                                        |

From the information provided on the page, we can see the following:

**Hop 1:** The packet leaves your local network, passing through a private IP address (10.140.10.147), suggesting it is your router or a device within your local network.

**Hop 2:** The packet enters the Amazon Web Services (AWS) infrastructure with the IP 3.236.63.11, belonging to Amazon.com, Inc., indicating that AWS is handling part of the traffic.

**Hops 3 to 6:** These hops show no response from the routers. This could be due to routers configured not to respond to traceroute or packet loss, which is not unusual.

**Hop 7:** The packet continues within AWS infrastructure, reaching IP 99.83.66.22, without a hostname available, indicating it is still within Amazon's network.

**Hop 8:** Still within AWS, the packet reaches IP 99.83.69.241, again without a hostname available.

**Hop 9:** The packet leaves AWS and enters the network of Columbus Networks USA, Inc. with IP 69.79.100.53, showing it has reached a service provider in the United States.

**Hop 10:** Within Columbus Networks, the packet reaches IP 69.79.100.5, with the hostname ae2.brx-mx2020-2.boca-raton.fl.usa.cwc.com, indicating its passage through Boca Raton, Florida.

**Hop 11:** The packet enters the network of Liberty Networks of Colombia with IP 190.131.207.5, showing it has reached a service provider in Colombia.

**Hop 12:** Continuing within Liberty Networks, the packet reaches IP 190.131.207.183, confirming its presence within the Colombian network.

**Hop 13:** There is no response at this hop, which could be due to packet filtering or routers configured not to respond to traceroute requests.

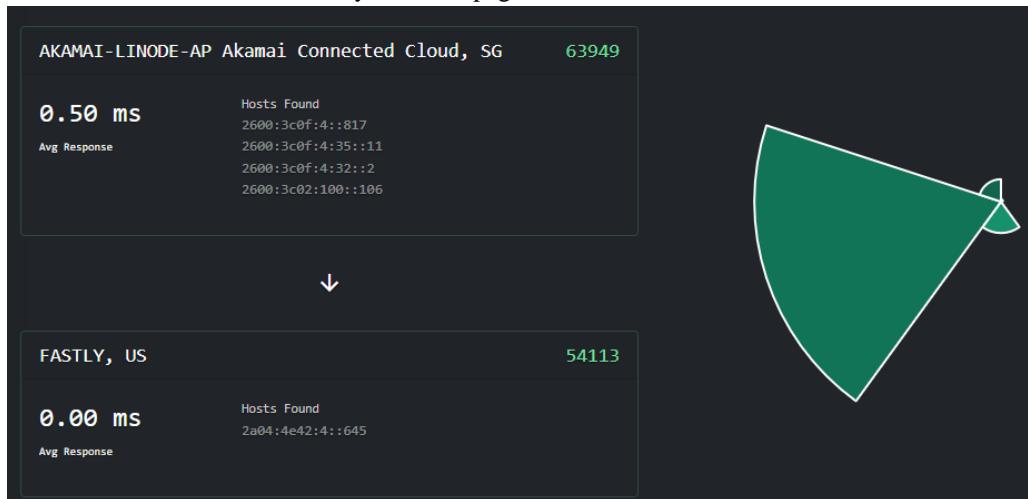
**Hop 14:** The packet reaches IP 45.239.88.78, registered to an unknown ASN, but continues its way to the destination.

**Hop 15:** No response at this hop, similar to previous hops where routers do not respond.

**Hop 16:** The packet reaches IP 45.239.88.78 again, now identified as part of the network of Escuela Colombiana de Ingeniería Julio Garavito.

**Hop 17:** Finally, the packet reaches IP 45.239.88.88, belonging to the network of Escuela Colombiana de Ingeniería Julio Garavito, thus reaching its destination.

- Now we search for the University Stanford page results



- 

**The information we can obtain is the following:**

**Hop 1:** Device managed by Akamai Connected Cloud in Singapore (IP: 2600:3c0f:4::817), with a response time of 0.50 ms, indicating a fast connection.

**Hop 2:** Another Akamai Connected Cloud device in Singapore (IP: 2600:3c0f:4:35::11), with a response time of 0.57 ms, similar to the first hop, suggesting physical proximity.

**Hop 3:** Akamai Connected Cloud node (IP: 2600:3c0f:4:32::2) in Singapore, with a response time of 1.43 ms, showing a slight increase in latency.

**Hop 4:** Another Akamai Connected Cloud node (IP: 2600:3c02:100::106) in Singapore, with a response time of 9.05 ms, indicating higher latency, likely due to greater distance or network congestion.

**Hop 5:** Intermediate node with no ISP information (IP: 2001:504:17:110::215), with a response time of 0.72 ms, suggesting an efficient connection, though the ISP is unspecified.

**Hop 6:** Node managed by Fastly in the US (IP: 2a04:4e42:4::645), with a response time of 0.65 ms, indicating a fast connection on the final stretch before reaching the destination.

○



○

- The only point on the map corresponds to a Fastly node, a content delivery network (CDN) that optimizes content delivery from servers located close to the destination. The IP address 2a04:4e42:4::645 represents the last hop before reaching the University of Stanford server, which explains the display of a single point on the map due to the proximity and efficiency of the node. CDNs like Fastly hide intermediate hops, resulting in only the final node being shown.
- 
- b. Using the tracert or traceroute command, search for a page in France and check the route.

```
jgamb ➤ System32 ➤ tracert www.info.gouv.fr

Traza a la dirección cs964.wpc.upsiloncdn.net [152.199.55.123]
sobre un máximo de 30 saltos:

 1 5 ms 5 ms 5 ms 192.168.1.254
 2 * * * Tiempo de espera agotado para esta solicitud.
 3 19 ms 35 ms 16 ms 10.166.12.58
 4 22 ms 21 ms 19 ms 10.166.12.57
 5 39 ms 40 ms 33 ms static-adsl200-24-35-183.epm.net.co [200.24.35.183]
 6 81 ms 81 ms 82 ms static-adsl200-24-33-90.epm.net.co [200.24.33.90]
 7 82 ms 81 ms 92 ms ae-112.border1.min.edgecastcdn.net [152.195.89.204]
 8 82 ms 73 ms 74 ms po-67.core1.mid.edgecastcdn.net [152.195.89.145]
 9 69 ms 69 ms 68 ms 152.199.55.123

c. Traza completa.
```

**Hop 1:** The first hop is the local network gateway (192.168.1.254), which responds quickly at 5 ms, indicating that the packet hasn't left the local network yet.

**Hop 2:** The second hop does not respond, which suggests that the device is configured not to reply to traceroute requests or that it is blocking this type of traffic.

**Hop 3:** The third hop shows an internal IP address of the ISP network (10.166.12.58), with response times ranging from 16 ms to 35 ms, indicating that the packet is still within the ISP's infrastructure.

**Hop 4:** The fourth hop shows another internal IP address of the ISP (10.166.12.57), with similar response times, indicating that the packet is still within the ISP's network.

**Hop 5:** The fifth hop shows a public IP address (200.24.35.183) from the ISP, with response times of 39 ms to 40 ms, suggesting that the packet has exited the ISP's internal network and is now on the public internet.

**Hop 6:** The sixth hop shows another public IP address (200.24.33.90) from the ISP, with response times of 81 ms to 82 ms, indicating that the packet is traveling through a network with greater distance or congestion.

**Hop 7:** The seventh hop corresponds to a node in the Edgecast CDN (152.195.89.204), with response times ranging from 81 ms to 92 ms, suggesting that the packet is passing through a Content Delivery Network (CDN).

**Hop 8:** The eighth hop remains in the Edgecast CDN network (152.195.89.145), with slightly lower response times (73 ms to 74 ms), bringing the packet closer to the final destination.

**Hop 9:** The ninth hop reaches the destination server (152.199.55.123), with response times of 69 ms, indicating that the packet has successfully arrived at the target website, [www.info.gouv.fr](http://www.info.gouv.fr).

- 
- d. Download and install software such as VisualRoute, Open Visual Traceroute, or similar. They can be free tools or demos.
- We go to [Open Visual Traceroute](#) and install the application

 | Home | | Installation | | Release Notes | | License | | About |

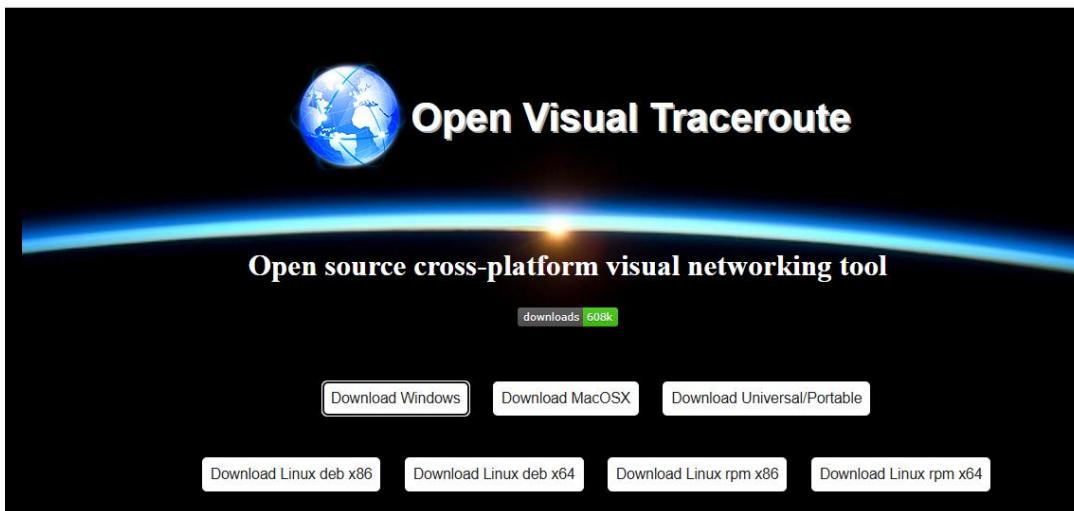
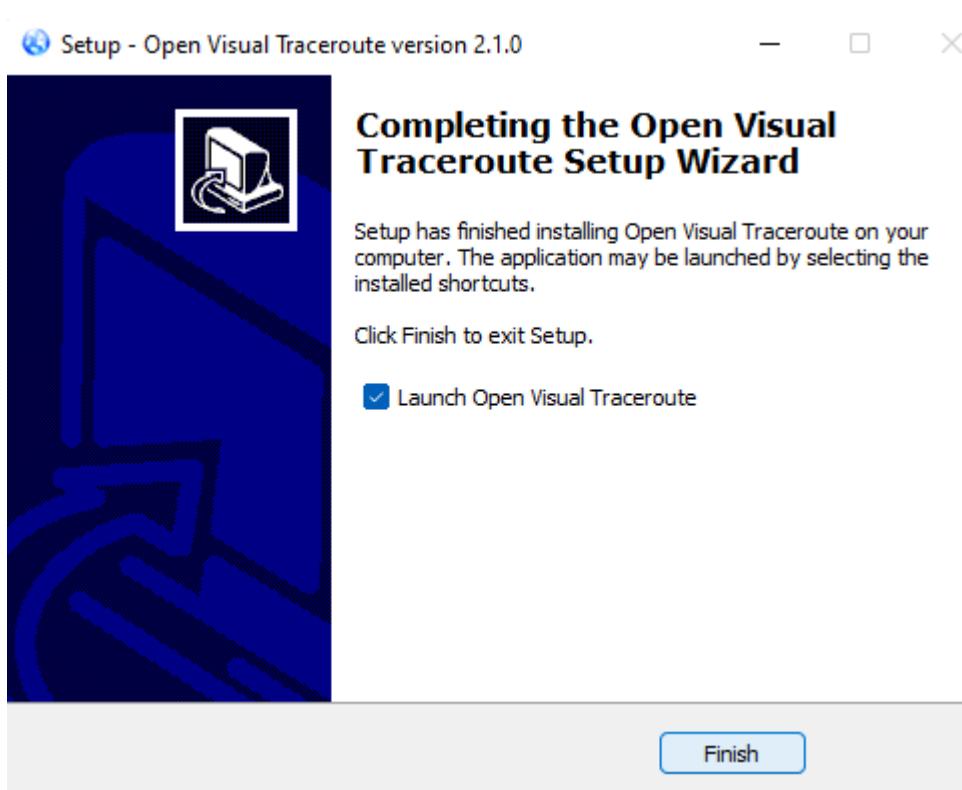


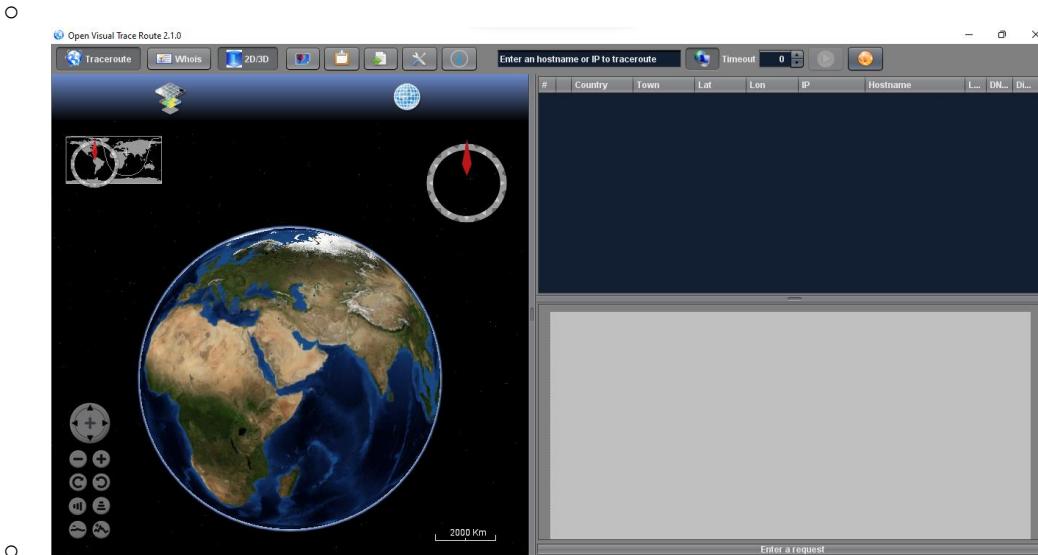
Figure 109 Open Visual Traceroute page

- We configure the installation features and click on finish.
-



*Figure 110 Completion of the program installation*

Now we open the program, and the following interface will appear.



*Figure 111 Initial interface of Open Visual Traceroute*

- Now we will make 5 queries to different places in the world to see how it works, with the queries being to car manufacturers' websites
- 
- In Australia
  - [www.volksvagen-group.com/en](http://www.volksvagen-group.com/en)

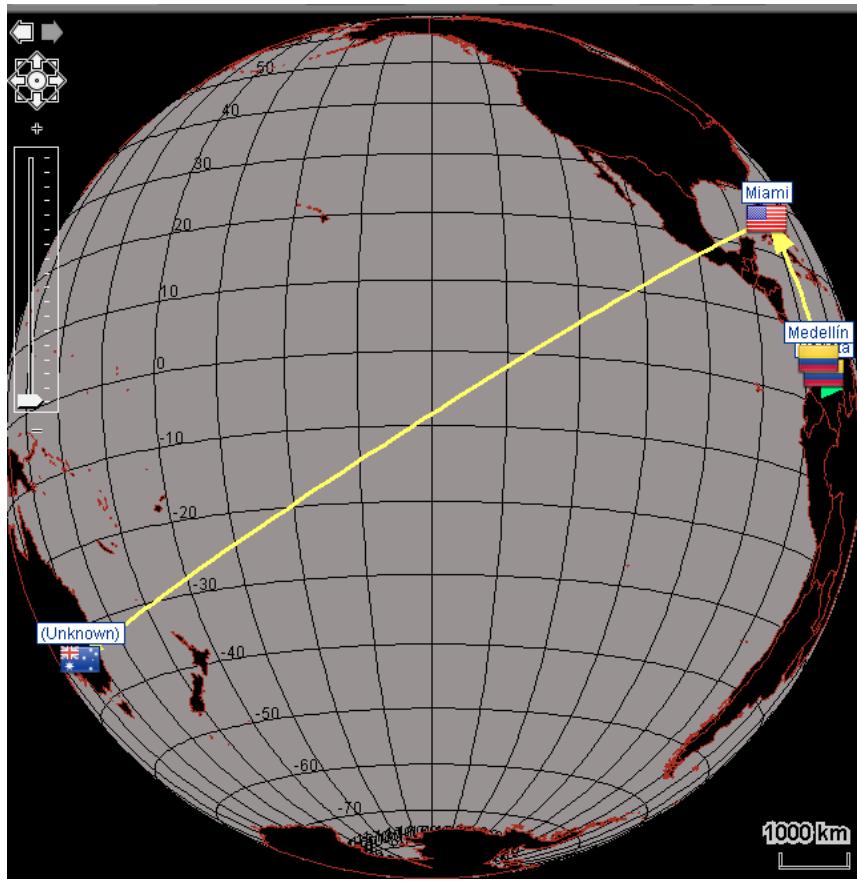


Figure 112 Route from Bogotá to Australia

| # | Country        | Town      | Lat     | Lon      | IP              | Hostname                | L... | DN... | Di... | ... |
|---|----------------|-----------|---------|----------|-----------------|-------------------------|------|-------|-------|-----|
| 1 | Colombia       | Bogotá    | 4.6115  | -74.0833 | 192.168.1.254   | (None)                  | 10   | 21    | 0     | ?   |
| 2 | (?)            | (?)       | 4.6115  | -74.0833 | *               | *                       | 0    | <1    | 0     | ?   |
| 3 | (?)            | (?)       | 4.6115  | -74.0833 | 10.166.12.58    | (None)                  | 16   | 21    | 0     | ?   |
| 4 | (?)            | (?)       | 4.6115  | -74.0833 | 10.166.12.57    | (None)                  | 19   | 20    | 0     | ?   |
| 5 | Colombia       | Medellín  | 6.2529  | -75.5646 | 200.24.35.183   | static-adsl200-24-35... | 32   | 19    | 245   | ?   |
| 6 | Colombia       | Medellín  | 6.2529  | -75.5646 | 200.24.33.90    | static-adsl200-24-33... | 77   | 23    | 0     | ?   |
| 7 | United Stat... | Miami     | 25.7634 | -80.1886 | 152.195.89.2... | ae-112.border1.min....  | 83   | 66    | 22... | ?   |
| 8 | United Stat... | Miami     | 25.7634 | -80.1886 | 152.195.89.1... | po-67.core1.mid.edg...  | 75   | 38    | 0     | ?   |
| 9 | Australia      | (Unknown) | -33.494 | 143.2104 | 117.18.238.2... | (None)                  | 73   | 132   | 15... | ?   |

Figure 113 Packet route to Australia

- In Central America
  - [www.grupoq.com](http://www.grupoq.com)

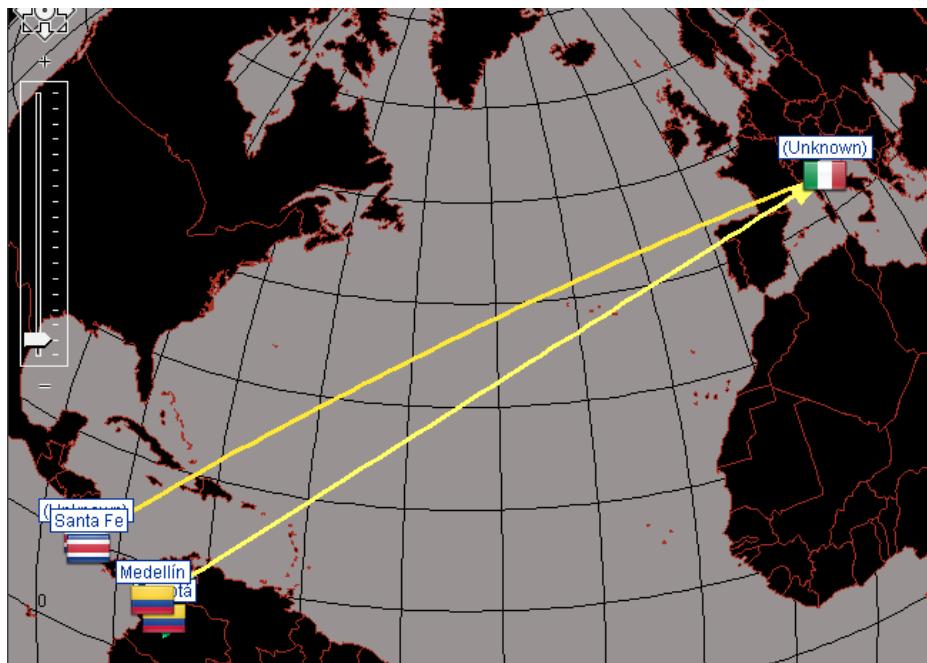


Figure 114 Route from Bogotá to Costa Rica

| #  | Country    | Town      | Lat     | Lon      | IP              | Hostname                | L... | DN... | Di... | ... |
|----|------------|-----------|---------|----------|-----------------|-------------------------|------|-------|-------|-----|
| 2  | (?)        |           | 4.6115  | -74.0833 | *               | *                       | 0    | <1    | 0     | ?   |
| 3  | (?)        |           | 4.6115  | -74.0833 | 10.166.12.58    | (None)                  | 15   | 20    | 0     | ?   |
| 4  | (?)        |           | 4.6115  | -74.0833 | 10.166.12.57    | (None)                  | 19   | 20    | 0     | ?   |
| 8  | (?)        |           | 43.1479 | 12.1097  | *               | *                       | 0    | <1    | 0     | ?   |
| 9  | (?)        |           | 43.1479 | 12.1097  | *               | *                       | 0    | <1    | 0     | ?   |
| 10 | (?)        |           | 43.1479 | 12.1097  | *               | *                       | 0    | <1    | 0     | ?   |
| 1  | Colombia   | Bogotá    | 4.6115  | -74.0833 | 192.168.1.254   | (None)                  | 5    | 23    | 0     | ?   |
| 5  | Colombia   | Medellín  | 6.2529  | -75.5646 | 200.24.35.183   | static-adsl200-24-35... | 33   | 21    | 245   | ?   |
| 6  | Colombia   | Medellín  | 6.2529  | -75.5646 | 200.24.33.99    | static-adsl200-24-33... | 74   | 21    | 0     | ?   |
| 11 | Costa Rica | (Unknown) | 10.0029 | -84.0    | 200.107.83.1... | (None)                  | 1... | 20    | 97... | ?   |
| 12 | Costa Rica | Santa Fe  | 9.0932  | -83.5572 | 190.0.224.34    | (None)                  | 1... | 121   | 112   | ?   |
| 13 | Costa Rica | Santa Fe  | 9.0932  | -83.5572 | 190.0.224.62    | (None)                  | 1... | 333   | 0     | ?   |
| 14 | Costa Rica | Santa Fe  | 9.0932  | -83.5572 | 190.0.230.180   | sitios.grupoq.co.cr     | 1... | 27    | 0     | ?   |
| 7  | Italy      | (Unknown) | 43.1479 | 12.1097  | 79.140.83.82    | (None)                  | 76   | 609   | 93... | ?   |

Figure 115 Packet route to Costa Rica

- In North America
  - [www.renaultgroup.com](http://www.renaultgroup.com)

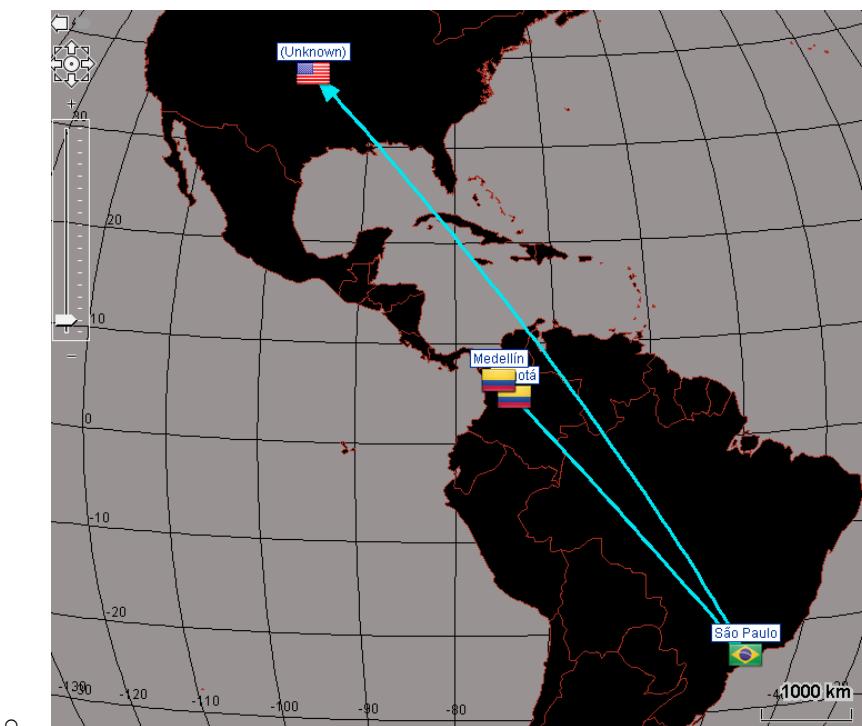


Figure 116 Route from Bogotá to the United States

| ... | Country      | Town      | Lat      | Lon      | IP            | Hostname            | L... | D... | Di... | ... |
|-----|--------------|-----------|----------|----------|---------------|---------------------|------|------|-------|-----|
| 1   | Colombia     | Bogotá    | 4.6115   | -74.0833 | 192.168.1.... | (None)              | 5    | 19   | 0     | ?   |
| 2   | ?            | *         | 4.6115   | -74.0833 | *             | *                   | 0    | <1   | 0     | ?   |
| 3   | ?            | *         | 4.6115   | -74.0833 | *             | *                   | 0    | <1   | 0     | ?   |
| 4   | ?            | *         | 4.6115   | -74.0833 | 10.166.12.... | (None)              | 16   | 19   | 0     | ?   |
| 5   | Colombia     | Medellín  | 6.2529   | -75.5646 | 200.24.35.... | static-adsl200-2... | 15   | 19   | 245   | ?   |
| 6   | ?            | *         | 6.2529   | -75.5646 | *             | *                   | 0    | <1   | 0     | ?   |
| 7   | ?            | *         | 6.2529   | -75.5646 | *             | *                   | 0    | <1   | 0     | ?   |
| 8   | ?            | *         | 6.2529   | -75.5646 | *             | *                   | 0    | <1   | 0     | ?   |
| 9   | ?            | *         | 6.2529   | -75.5646 | *             | *                   | 0    | <1   | 0     | ?   |
| ... | Brazil       | São Paulo | -23.5335 | -46.6359 | 15.230.0.8    | (None)              | 16   | 92   | 4...  | ?   |
| ... | United St... | (Unknown) | 37.751   | -97.822  | 18.155.25...  | server-18-155-2...  | 14   | 93   | 8...  | ?   |

Figure 117 Packet route to United States

- In Asia
  - [www.cheryinternational.com](http://www.cheryinternational.com)

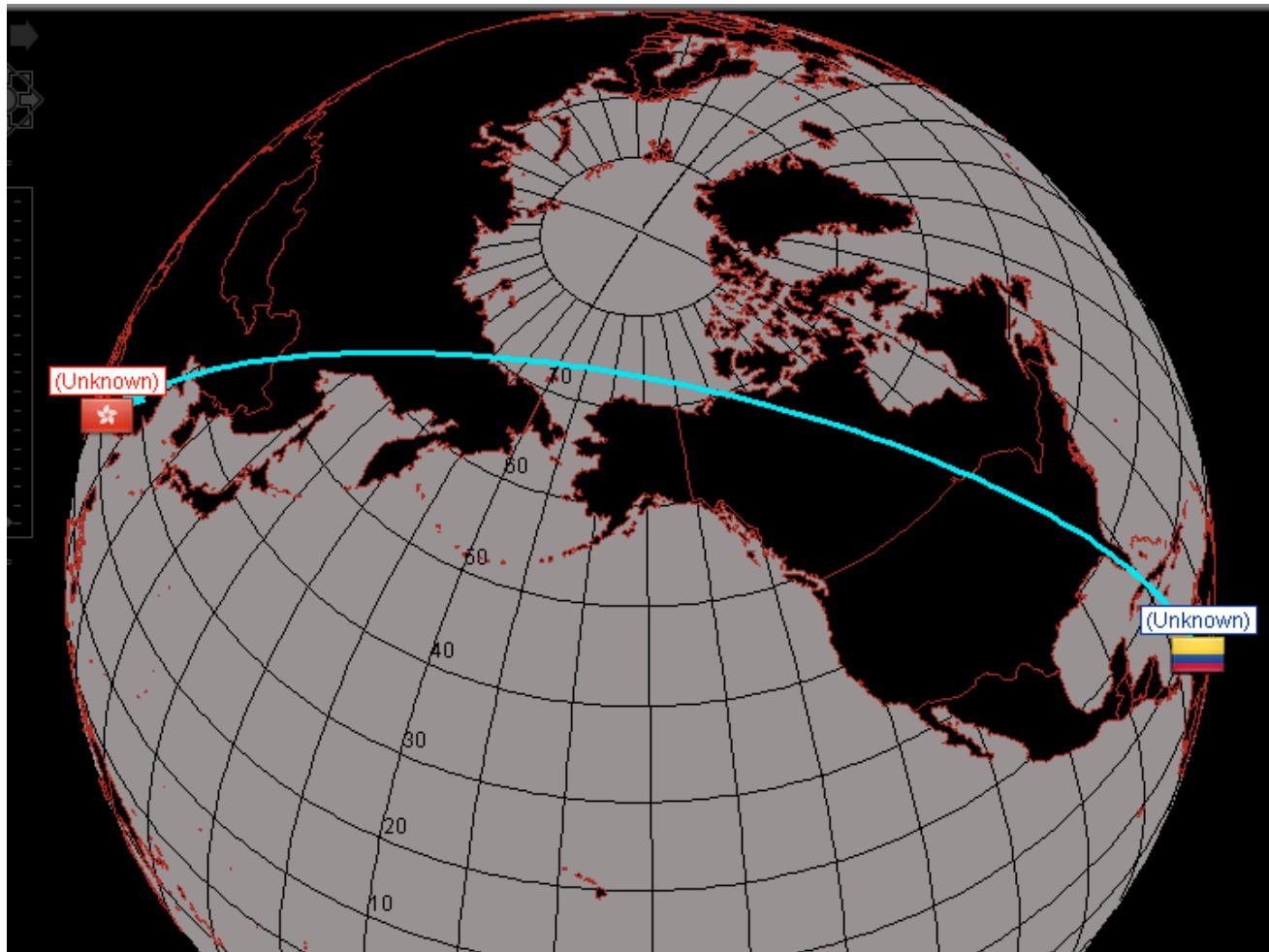


Figure 118 Route from Bogotá to Hong Kong

| #  | Country   | Town      | Lat     | Lon      | IP            | Hostname               | L... | DN... | Di... | ... |
|----|-----------|-----------|---------|----------|---------------|------------------------|------|-------|-------|-----|
| 1  | Colombia  | Bogotá    | 4.6115  | -74.0833 | 192.168.1.254 | (None)                 | 11   | 31    | 0     | ?   |
| 2  | ?         | *         | 4.6115  | -74.0833 | *             | *                      | 0    | <1    | 0     | ?   |
| 3  | ?         | *         | 4.6115  | -74.0833 | *             | *                      | 0    | <1    | 0     | ?   |
| 4  | ?         | *         | 4.6115  | -74.0833 | 10.166.12.57  | (None)                 | 23   | 35    | 0     | ?   |
| 5  | Colombia  | (Unknown) | 4.5981  | -74.0799 | 200.25.30.146 | ae3.0.edge1.bog3.as... | 34   | 162   | 1     | ?   |
| 6  | Colombia  | (Unknown) | 4.5981  | -74.0799 | 200.25.30.161 | (None)                 | 15   | 99    | 0     | ?   |
| 7  | Colombia  | (Unknown) | 4.5981  | -74.0799 | 10.64.251.164 | (None)                 | 22   | 21    | 0     | ?   |
| 8  | ?         | *         | 4.5981  | -74.0799 | *             | *                      | 0    | <1    | 0     | ?   |
| 9  | ?         | *         | 4.5981  | -74.0799 | 10.64.57.102  | (None)                 | 24   | 26    | 0     | ?   |
| 10 | Hong Kong | (Unknown) | 22.2578 | 114.1657 | 154.94.93.13  | (None)                 | 18   | 320   | 16... | ?   |

Figure 119 Packet route to Honk Kong

- In Europe
  - [www.koenigsegg.com](http://www.koenigsegg.com)

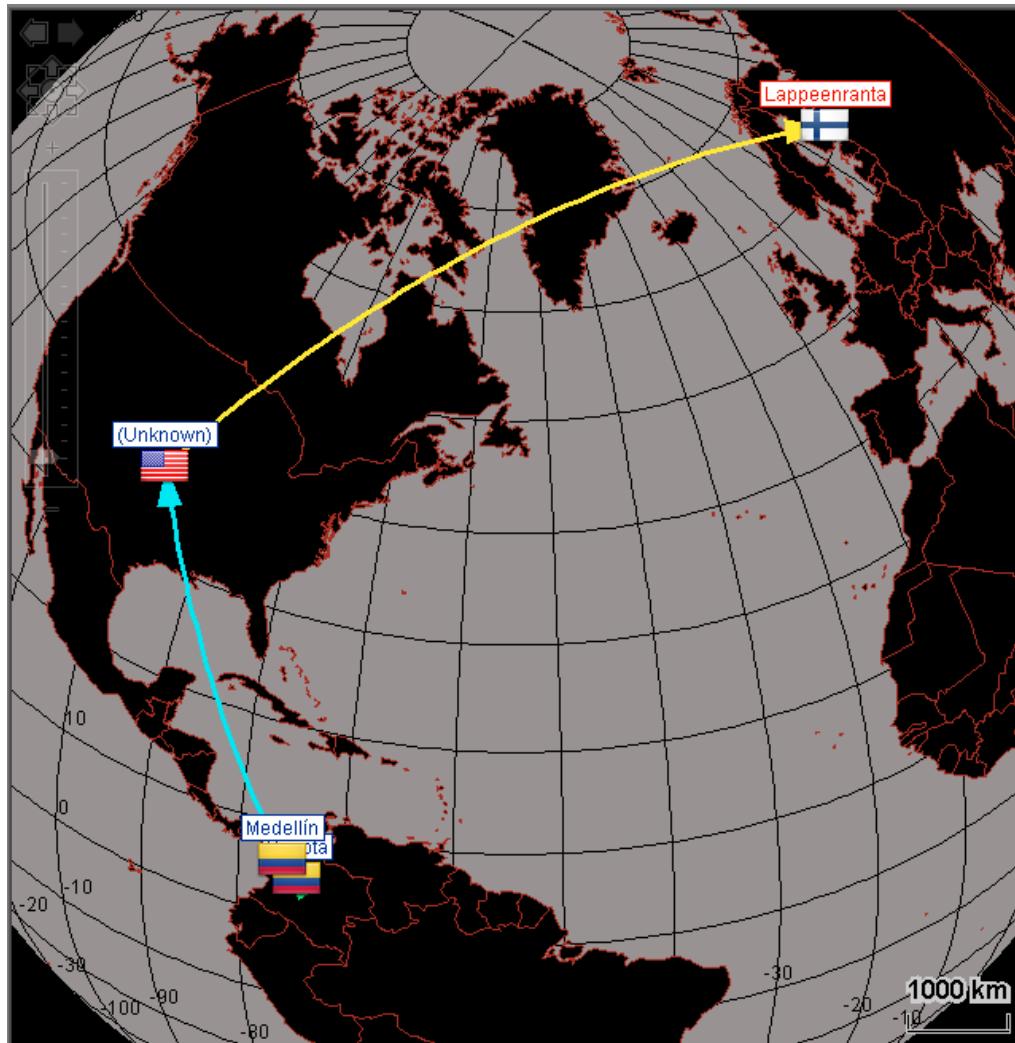


Figure 120 Route from Bogotá to Hong Kong

| # | Country       | Town         | Lat     | Lon      | IP              | Hostname                | L... | DN... | Di... | ... |
|---|---------------|--------------|---------|----------|-----------------|-------------------------|------|-------|-------|-----|
| 1 | Colombia      | Bogotá       | 4.6115  | -74.0833 | 192.168.1.254   | (None)                  | 12   | 26    | 0     | ?   |
| 2 | (Unknown)     | *            | 4.6115  | -74.0833 | *               | *                       | 0    | <1    | 0     | ?   |
| 3 | (Unknown)     | *            | 4.6115  | -74.0833 | 10.166.12.58    | (None)                  | 18   | 24    | 0     | ?   |
| 4 | (Unknown)     | *            | 4.6115  | -74.0833 | 10.166.12.57    | (None)                  | 18   | 24    | 0     | ?   |
| 5 | Colombia      | Medellín     | 6.2529  | -75.5646 | 200.24.35.179   | static-adsl200-24-35... | 20   | 22    | 245   | ?   |
| 6 | United States | (Unknown)    | 37.751  | -97.822  | 209.85.168.1... | (None)                  | 18   | 20    | 41... | ?   |
| 7 | Finland       | Lappeenranta | 61.0636 | 28.189   | 34.88.92.18     | 18.92.88.34.bc.googl... | 1... | 107   | 80... | ?   |

Figure 121 Packet route to Finland

## 2. Some Questions About Router Commands

- What is the difference between enable password and enable secret? If both are configured, which one takes precedence?
  - **Enable Password:** This command sets a password to access privileged mode on a Cisco router. The password configured with enable password is stored in plaintext in the configuration file, making it

- less secure.
  - ○ **Enable Secret:** This command also sets a password to access privileged mode, but the password configured with enable secret is stored encrypted in the configuration file, providing greater security.
  - ○ **Precedence:** If both commands are configured, enable secret takes precedence over enable password. This means that the router will use the password configured with enable secret for privileged mode authentication.
2. What is the difference between console and VTY?
- ○ **Console:** This is a direct physical connection to the router through the console port. This connection is used for initial device configuration, password recovery, and other administrative tasks requiring direct access. It does not require a network and is always available.
  - ○ **VTY (Virtual Teletype):** These are virtual connections that allow remote access to the router through a network using protocols like Telnet or SSH. VTY lines are mainly used for remote administration. A configured and accessible network is required to use VTY connections.
  -
3. What is the boot process of the routers in the Network Laboratory?
- - The boot process of a Cisco router generally follows these steps:
1. **Power-On Self-Test (POST):** When the router is powered on, it runs a diagnostic test known as the POST. This test checks the hardware components, such as the CPU, memory, and interfaces, to ensure they are functioning correctly. The POST program is stored in the router's ROM (Read-Only Memory).
  - 
  2. **Loading the Bootstrap Program:** After the POST, the router runs the bootstrap program, also stored in ROM. The bootstrap program's first task is to check the value of the configuration register, which dictates where to load the IOS (Internetwork Operating System). The default configuration register value (0x2102) instructs the router to load the IOS from the flash memory.
  3. **Loading the IOS:** The bootstrap program locates the IOS image in the flash memory and loads it into the router's RAM (Random Access Memory). The IOS is the core operating system that provides the router's functionality.
  - 
  4. **Initialization of Hardware and IOS:** Once the IOS is loaded, it initializes the router's hardware and prepares the system for operation. This includes setting up interfaces and applying initial settings.
  - 
  - 
  5. **Loading the Startup Configuration:** The IOS then searches for the startup configuration file stored in NVRAM (Non-Volatile RAM). This file, known as the startup-config, contains the saved configuration settings for the router. If the startup-config file is found, it is loaded into RAM and becomes the running configuration (running-config). If the file is not present in NVRAM, the router attempts to load a configuration file from a TFTP (Trivial File Transfer Protocol) server. If no TFTP server is found, the router enters Setup mode, allowing basic configuration.
  - 
  6. **Entering User Mode:** After the startup configuration is loaded, the router finishes its boot process and provides access to the Command-Line Interface (CLI). The router starts in user mode, where basic monitoring commands can be executed. From here, administrators can enter privileged mode to perform more advanced configurations and management tasks.
  - 
  -

4. What types of memory do the routers in the Network Laboratory have?
    - - **ROM (Read-Only Memory):** ROM in the Cisco 1841 router stores the bootstrap program, POST (Power-On Self-Test) routines, and the ROM monitor. This non-volatile memory retains its contents even when the router is powered off, ensuring critical startup instructions are always available.
      - **Flash Memory:** Flash memory holds the Cisco IOS (Internetwork Operating System) image and other system files. It is non-volatile, meaning it preserves data without power, and can be erased and reprogrammed to allow IOS updates.
      - **NVRAM (Non-Volatile RAM):** NVRAM stores the startup configuration file (startup-config). This type of memory is non-volatile, so it keeps the configuration settings intact even when the router is powered off or restarted.
      - **RAM (Random Access Memory):** RAM is used for the running configuration file (running-config), routing tables, and other operational data while the router is on. Unlike the other types, RAM is volatile, meaning it loses its data when the router is powered off or restarted.
    - 
    -
5. What is the difference between startup-configuration and running-configuration files?
    - **Startup-Configuration:** The startup-configuration file is stored in NVRAM (Non-Volatile RAM). This file contains the saved configuration of the router that is used when the device boots up or restarts. It retains its settings even when the router is powered off, ensuring that the router starts with the same configuration each time.
    - **Running-Configuration:** The running-configuration file resides in the router's RAM (Random Access Memory). This file holds the current active configuration of the router, which is used during its operation. Any changes made to the router's configuration are first applied to the running-configuration, making them effective immediately. However, these changes are temporary and will be lost if the router is rebooted unless they are saved to the startup-configuration.

### 3. Setup: Access and Basic Configuration of Routers

Review and document the different routers available in the Laboratory and the network interfaces they have.

We use the RS-232 serial cable extension and connect it to a computer, then connect it to the router



Figure 122 RS-232 serial cable extension

We open the Device Manager of the system and look for the port to which we are connected with router 10, in this case it is COM4. We select the serial connection option and change the serial line to the router's port.

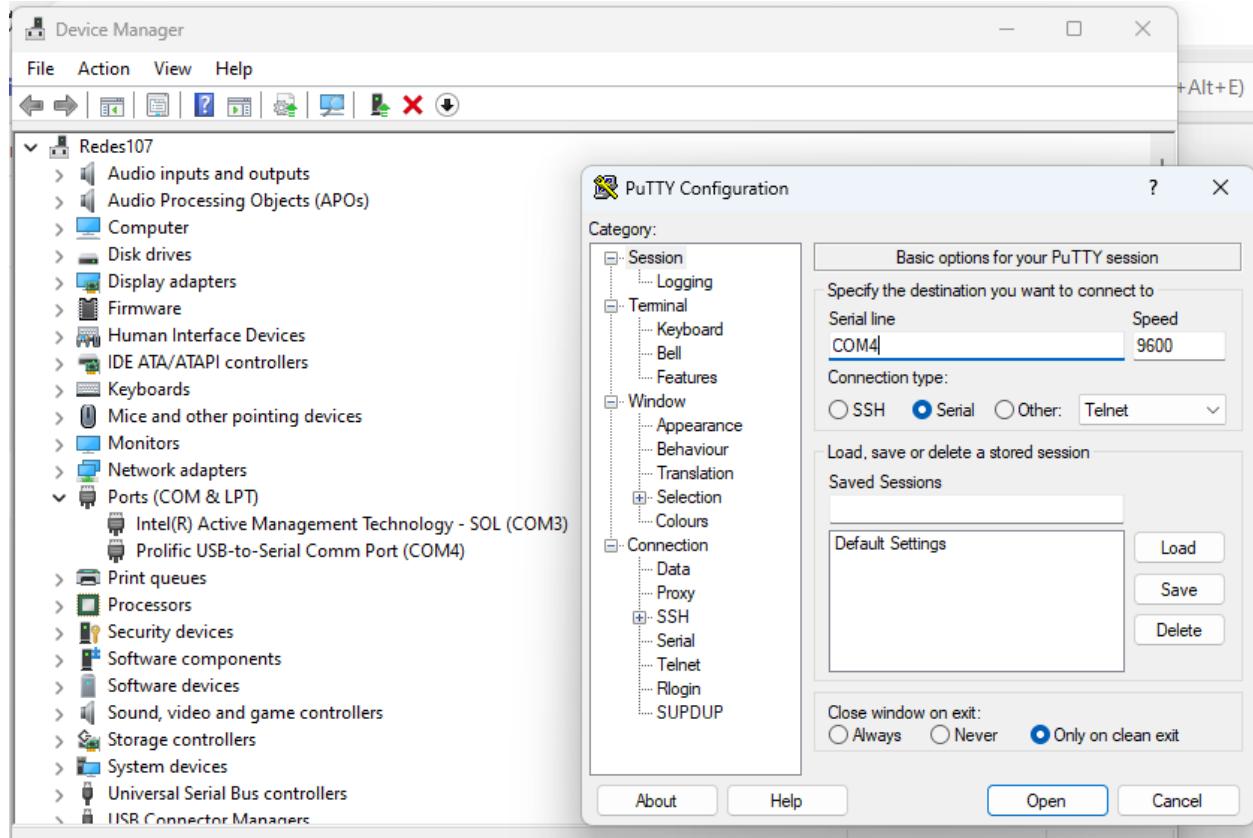
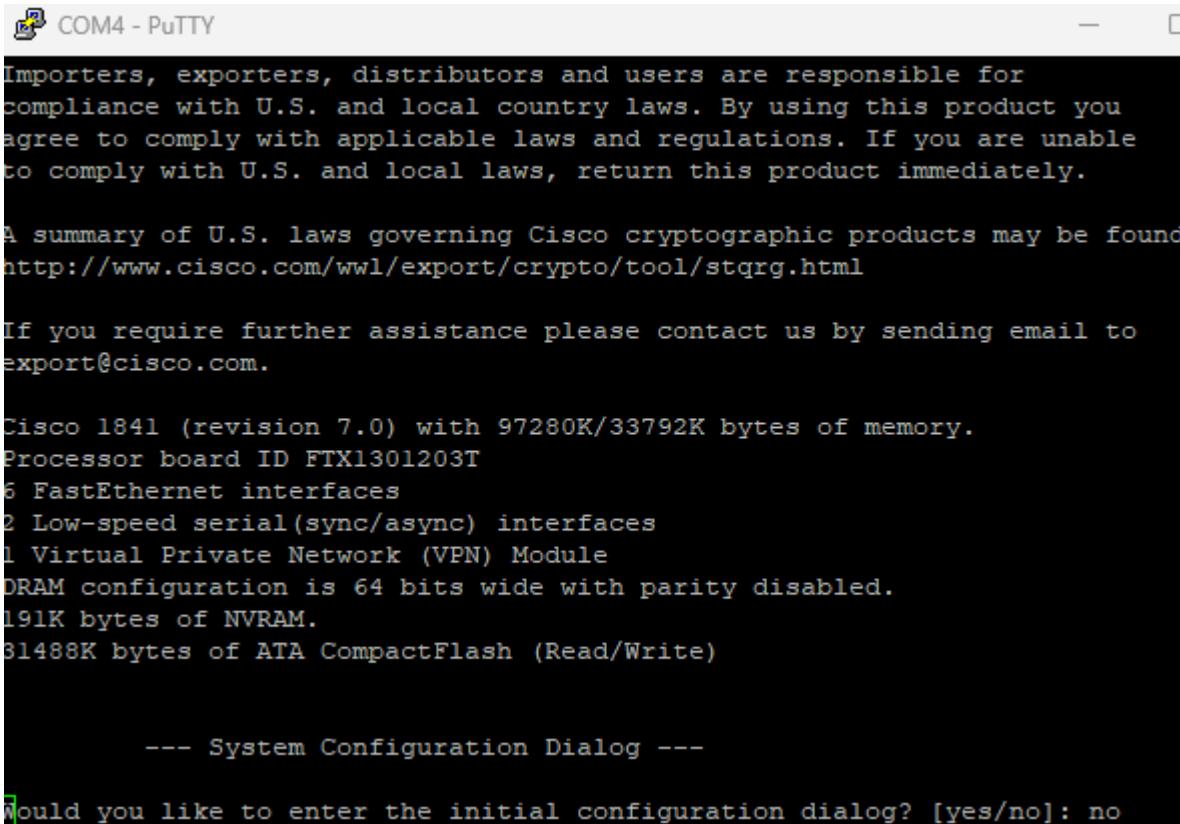


Figure 123 PuTTY configuration



```
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found
at http://www.cisco.com/ww1/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Cisco 1841 (revision 7.0) with 97280K/33792K bytes of memory.
Processor board ID FTX1301203T
6 FastEthernet interfaces
2 Low-speed serial(sync/async) interfaces
1 Virtual Private Network (VPN) Module
DRAM configuration is 64 bits wide with parity disabled.
191K bytes of NVRAM.
31488K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: no
```

Figure 124 initial router configuration

We observe that there are no restrictions when accessing the router or entering privileged mode, so we configure it according to the specifications.

Now we are going to explain the process the router goes through on boot when configured in modes 0x2142 and 0x2102.

## Configuration Register 0x2142

- **Purpose:** Password recovery and troubleshooting.
- **Process:**
  1. Performs the POST to check hardware components.
  2. Loads the bootstrap program from ROM.
  3. Checks the configuration register and detects the value 0x2142.
  4. Bypasses the startup-config in NVRAM, loading default settings.
  5. Loads the IOS from flash memory into RAM.
  6. Enters setup mode or CLI with factory default settings.

## Configuration Register 0x2102

- **Purpose:** Normal router operation.
- **Process:**
  1. Performs the POST to check hardware components.
  2. Loads the bootstrap program from ROM.
  3. Checks the configuration register and detects the value 0x2102.
  4. Loads the IOS from flash memory into RAM.
  5. Loads the startup-config from NVRAM.
  6. Completes the boot process and presents the CLI in user mode with the saved configuration.

We perform the following configuration using physical devices. Which include:

- Access keys for privileged mode, console, and remote access. The privileged mode key should be "cisco", the console key "claveC", and the remote access (telnet) key "claveT".
- Router name. Assign the router the last name of one of the students in the group.
- Console and remote access screen synchronization.
- Description of the interfaces used.
- Disable remote command server lookup.
- Message of the day.

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Torres
Torres(config)#banner motd "configuracion router8"
Torres(config)#line console 0
Torres(config-line)#logging synchronous
Torres(config-line)#password claveC
Torres(config-line)#login
Torres(config-line)#exit
Torres(config)#line vty 0 15
```

Figure 125 basic configuration

We configure interfaces. First, we set the IPs according to the given conditions. We need a network for 4000 hosts and another for 600 hosts. Based on the classroom distribution, we will manage the network 88.0.0.0/10.

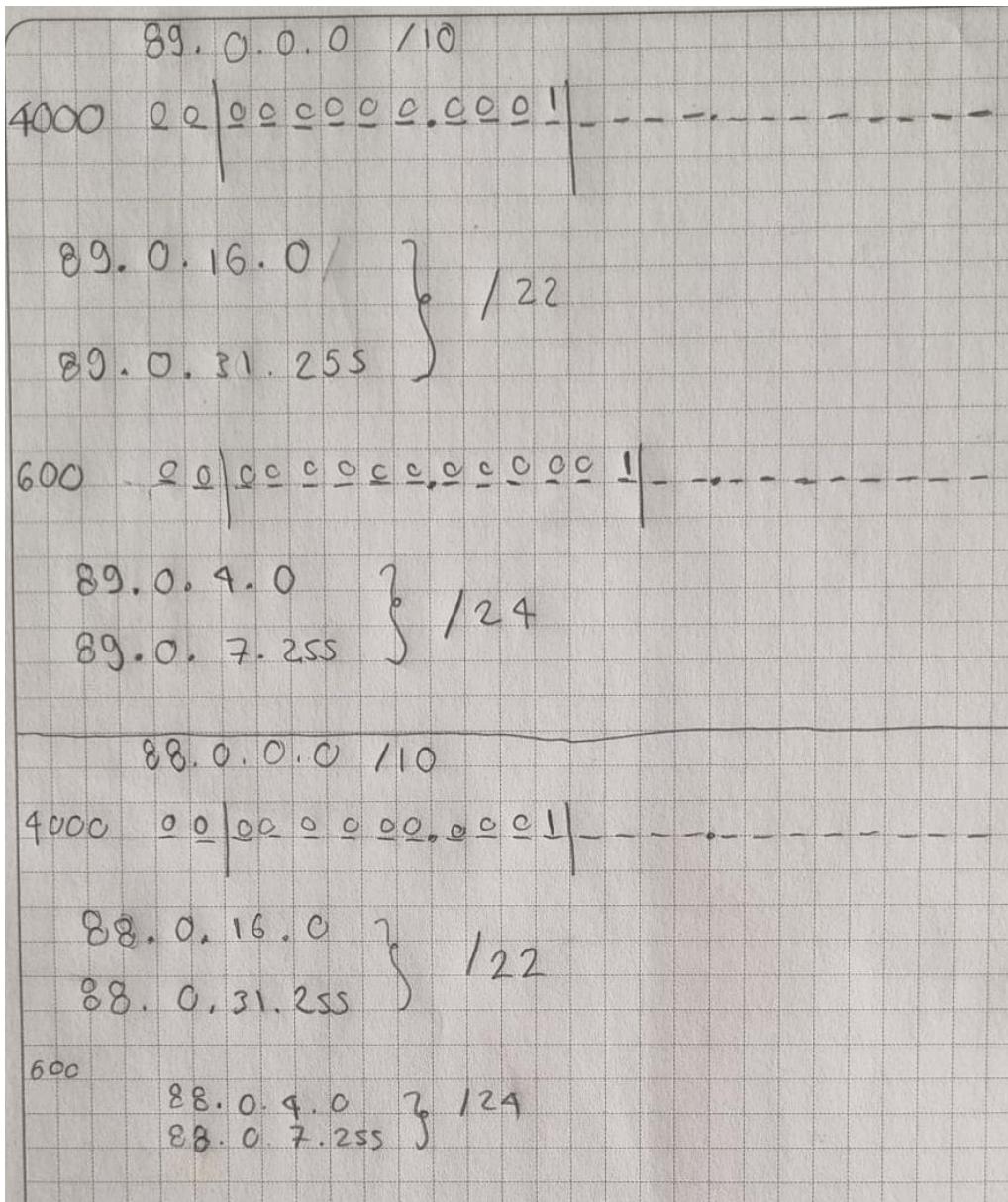
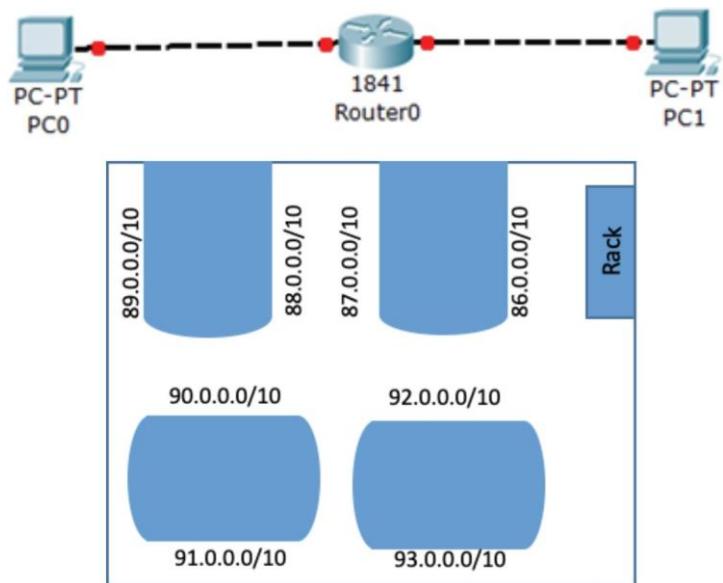


Figure 126 subnetting configuration

When performing the subnetting process, the subnets we will manage are as follows:  
 88.0.4.1/22 y 88.0.16.1/24



Now we configure the physical settings on the router to enable communication between the two computers.

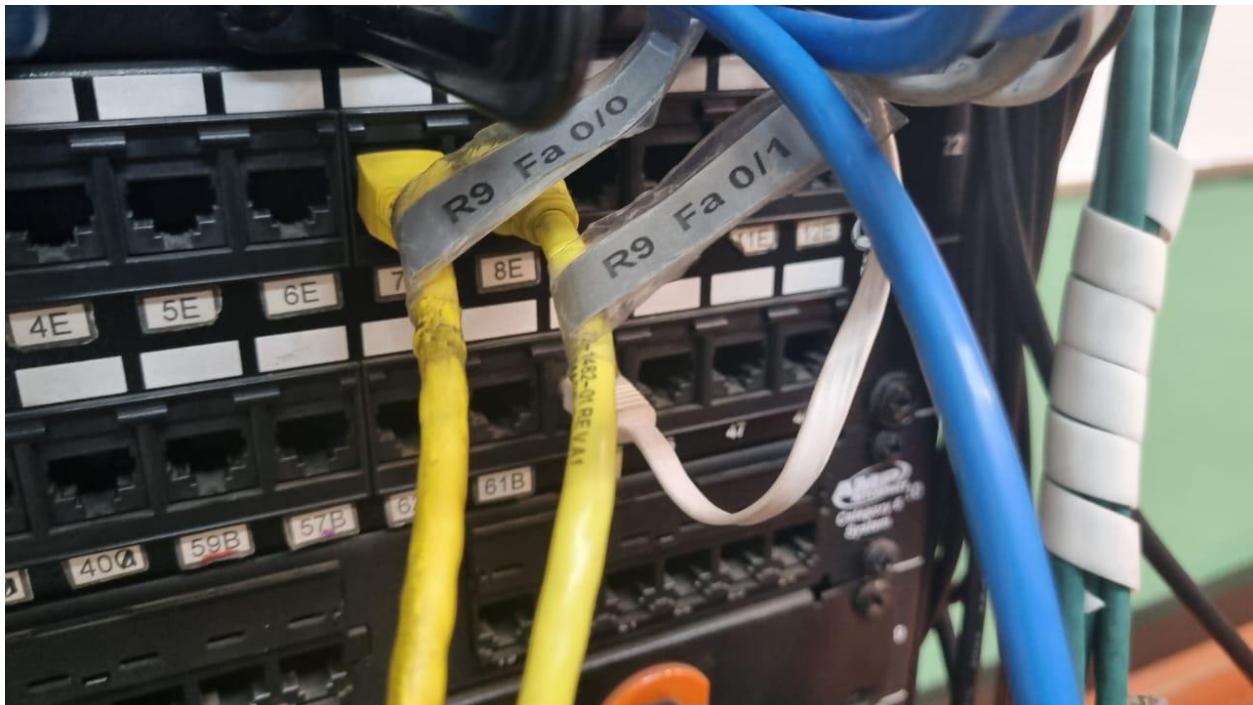


Figure 127 connection of devices



Figure 128 connection between devices and router

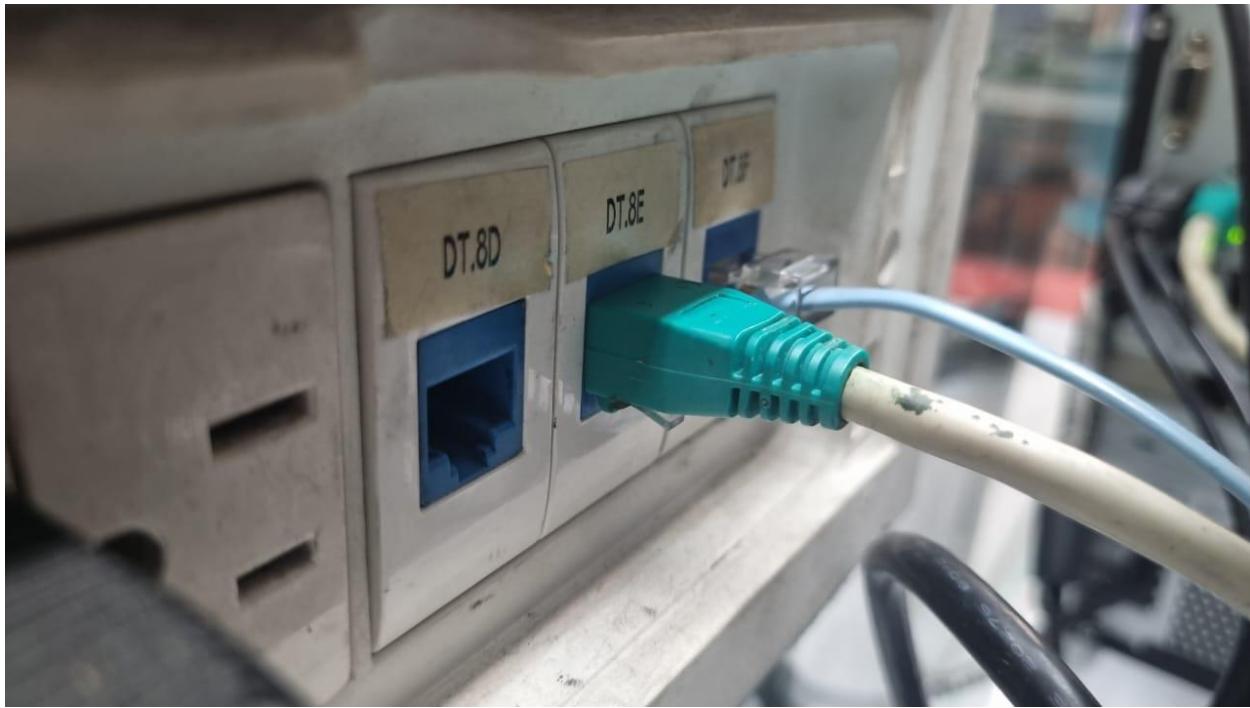
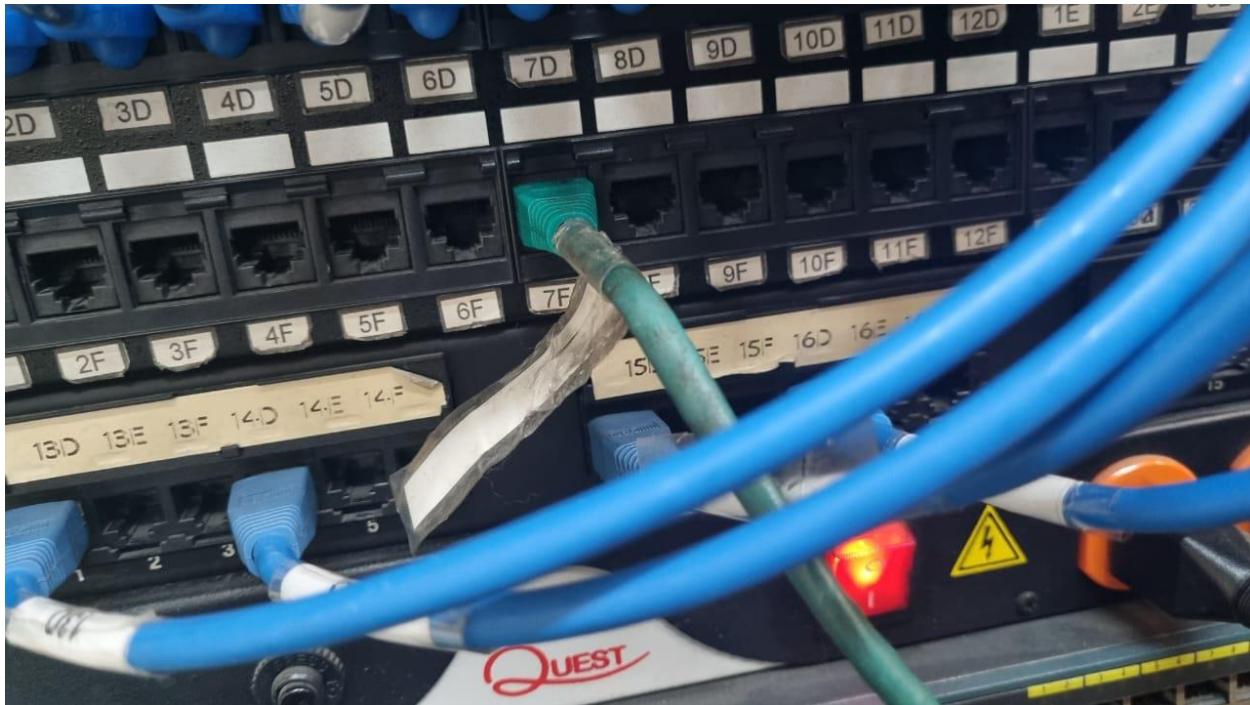


Figure 129 connection between devices and router



*Figure 130 connection to console cable*

Now we configure the logical settings on the router to enable communication between the two computers.

```
Torres(config)#interface fa0/0
Torres(config-if)#ip address 88.0.16.1 255.255.252.0
Torres(config-if)#description "router 8 fa0/0"
Torres(config-if)#no shutdown
Torres(config-if)#exit
Torres(config)#exit
```

*Figure 131 configuration interface Fa0/0*

```
Torres(config)#interface fa0/1
Torres(config-if)#ip address 88.0.4.1 255.255.255.0
Torres(config-if)#description "router 8 fa0/1"
Torres(config-if)#no shutdown
Torres(config-if)#

```

*Figure 132 configuration interface fa0/1*

We see that the interfaces have been configured correctly with the command `show ip interface brief`.

```

Torres(config)#exit
Torres#configure terminal
*Jan 1 00:09:04.975: %SYS-5-CONFIG_I: Configured from console by console
Torres#show ip interface brief
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 88.0.16.1 YES manual up up
FastEthernet0/1 88.0.4.1 YES manual up up
Serial0/0/0 unassigned YES unset administratively down down
Serial0/0/1 unassigned YES unset administratively down down
Serial0/1/0 unassigned YES unset administratively down down
Serial0/1/1 unassigned YES unset administratively down down
Torres#

```

Figure 133 Verification of the configuration of the interfaces

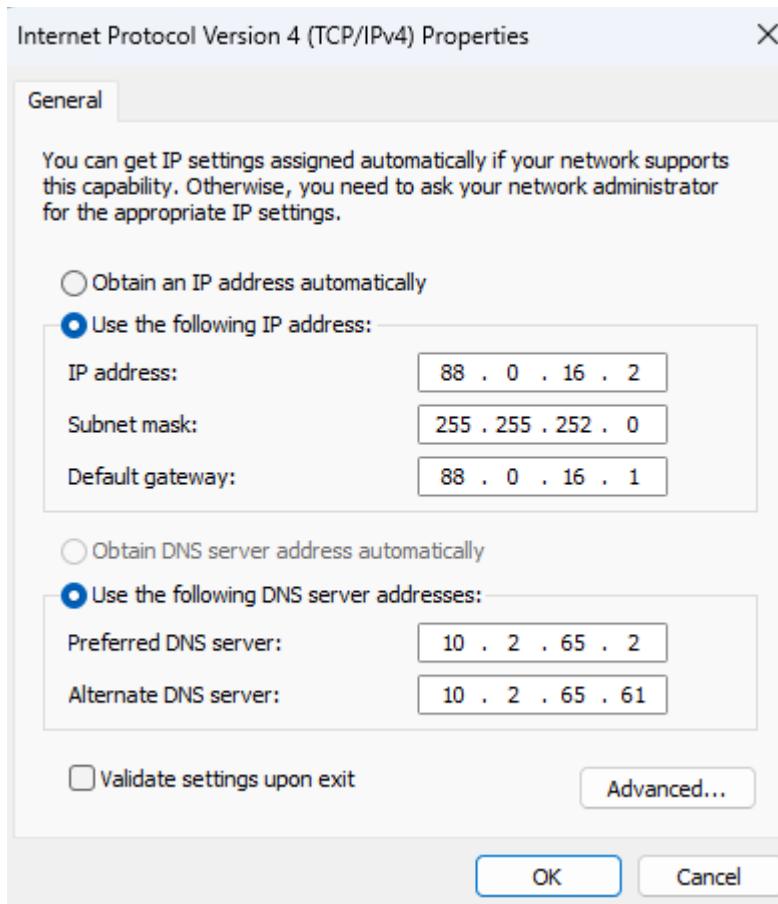


Figure 134 Configuration of IP and gateway on one of the devices

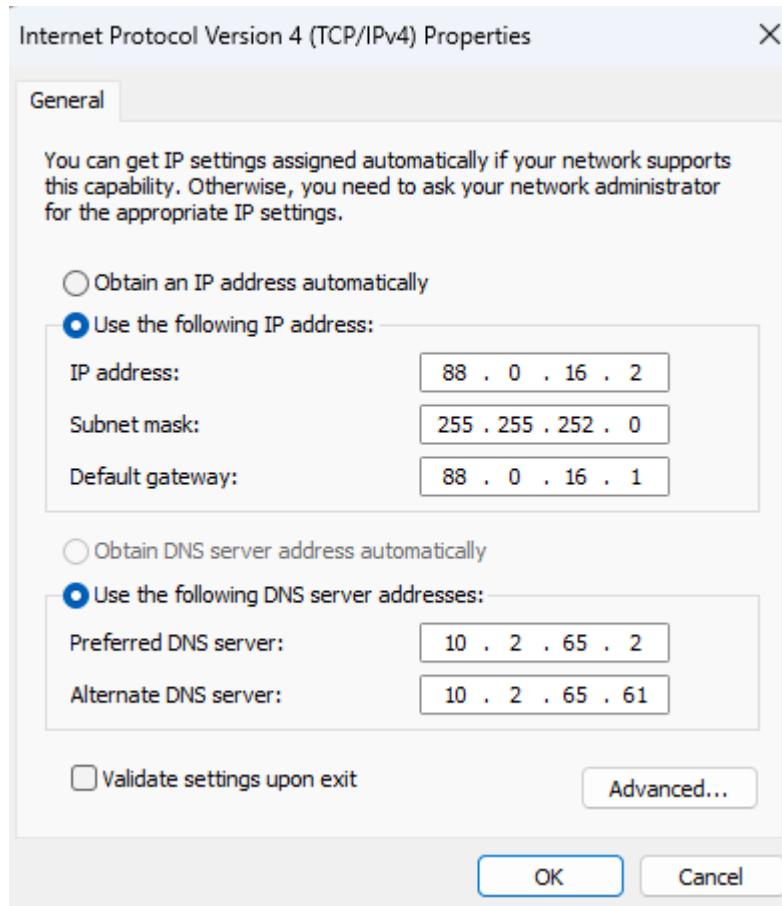


Figure 135 Configuration of IP and gateway on one of the devices

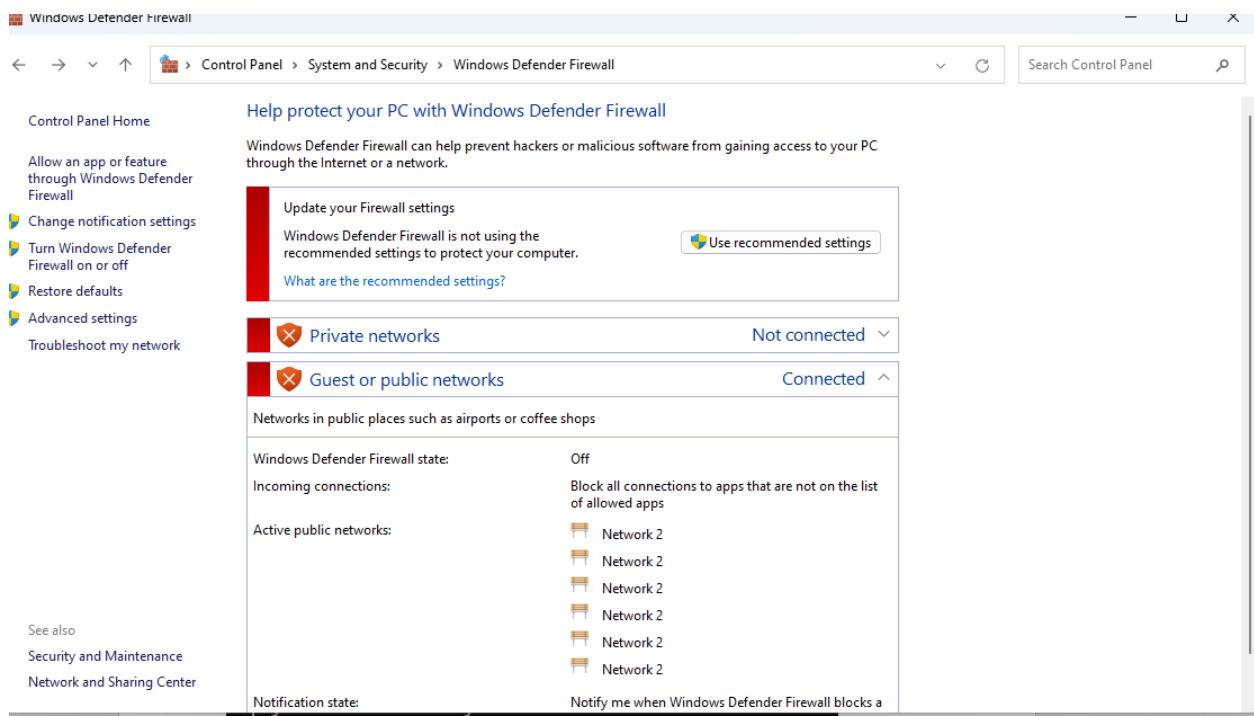


Figure 136 Disabling the firewall

We perform ping tests between the computers and the router's network interfaces.

```
Pinging 88.0.4.1 with 32 bytes of data:
Reply from 88.0.4.1: bytes=32 time=1ms TTL=255
Reply from 88.0.4.1: bytes=32 time=1ms TTL=255
Reply from 88.0.4.1: bytes=32 time=1ms TTL=255

Ping statistics for 88.0.4.1:
 Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
 Approximate round trip times in milli-seconds:
 Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Figure 137 test to interface 88.0.4.1

```
C:\Users\Redes>ping 88.0.4.2

Pinging 88.0.4.2 with 32 bytes of data:
Reply from 88.0.4.2: bytes=32 time=1ms TTL=127

Ping statistics for 88.0.4.2:
 Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
 Approximate round trip times in milli-seconds:
 Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Figure 138 test to device with ip 88.0.4.2

```
C:\Users\Redes>ping 88.0.16.2

Pinging 88.0.16.2 with 32 bytes of data:
Reply from 88.0.16.2: bytes=32 time<1ms TTL=128

Ping statistics for 88.0.16.2:
 Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
 Approximate round trip times in milli-seconds:
 Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figure 139 test to device with ip 88.0.16.0.2

#### 4. Setup - Serial Interconnection

- - What is a null modem?

- 
- A **null modem** is a cable that allows direct communication between two devices, such as two computers or routers, without the need for a traditional modem. It is used to connect two serial ports by crossing the transmission and reception pins, allowing bidirectional data transmission between the devices. This type of connection is commonly used in networking to connect routers or devices that do not have Ethernet network ports available.
- 
- **What is the clock rate command used for in routers, and why is it needed?**

The **clock rate** command is used in routers to set the clock speed on serial interfaces, especially for **DCE (Data Communications Equipment)** connections. This command is necessary because, in a serial connection between two devices, one of them must generate the "clock" that controls the data transmission rate. The device configured as DCE is responsible for generating this clock, while the DTE (Data Terminal Equipment) device simply receives it. Without a clock rate, data transmission cannot occur properly.

- **What does DTE and DCE mean? What is the relationship with the routers in the Network Laboratory?**

- **DTE (Data Terminal Equipment):** Refers to a device that receives data or connects to the equipment generating the data. In terms of routers, the **DTE** device would be the router or computer that does not generate the clock signal and connects to a DCE device.
- **DCE (Data Communications Equipment):** The device that generates the clock signal in a serial connection. In the network lab, the **DCE** router is responsible for providing the clock signal via the serial interface.

We connect our router with the network device 89.0.0.0/10. We ensure that all devices connected to the router are on the 89.0.0.0 network.

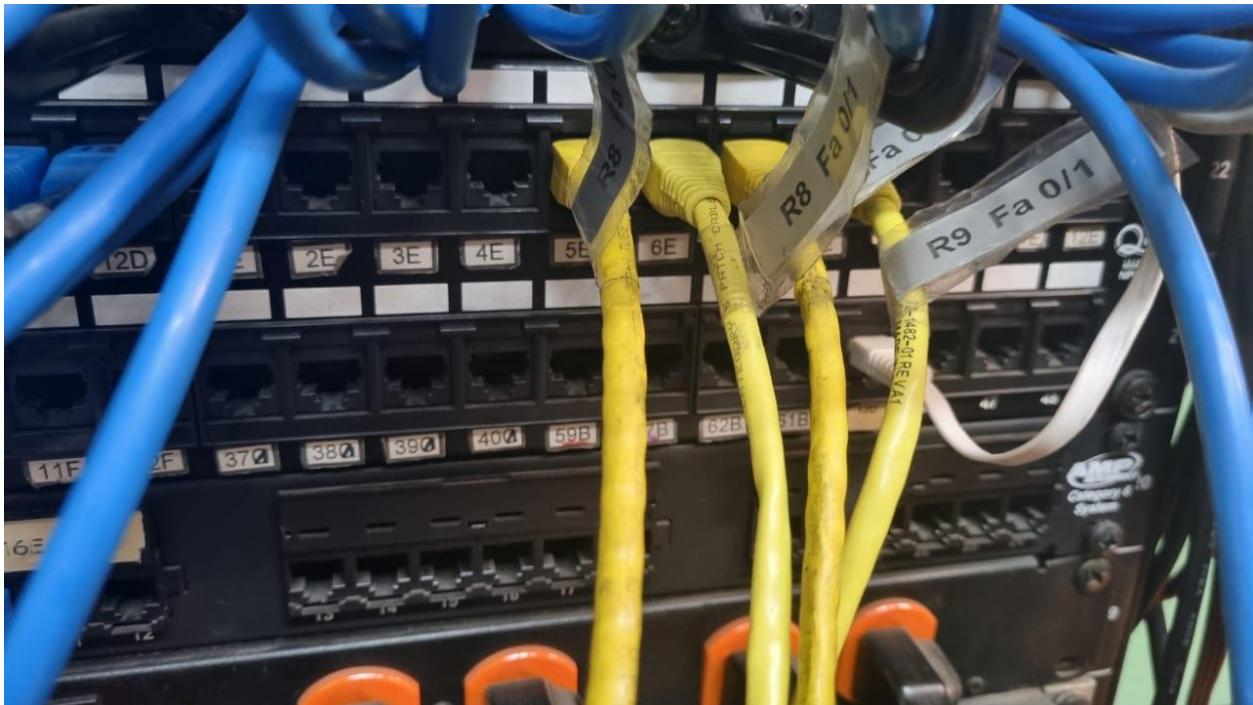


Figure 140 connection between devices

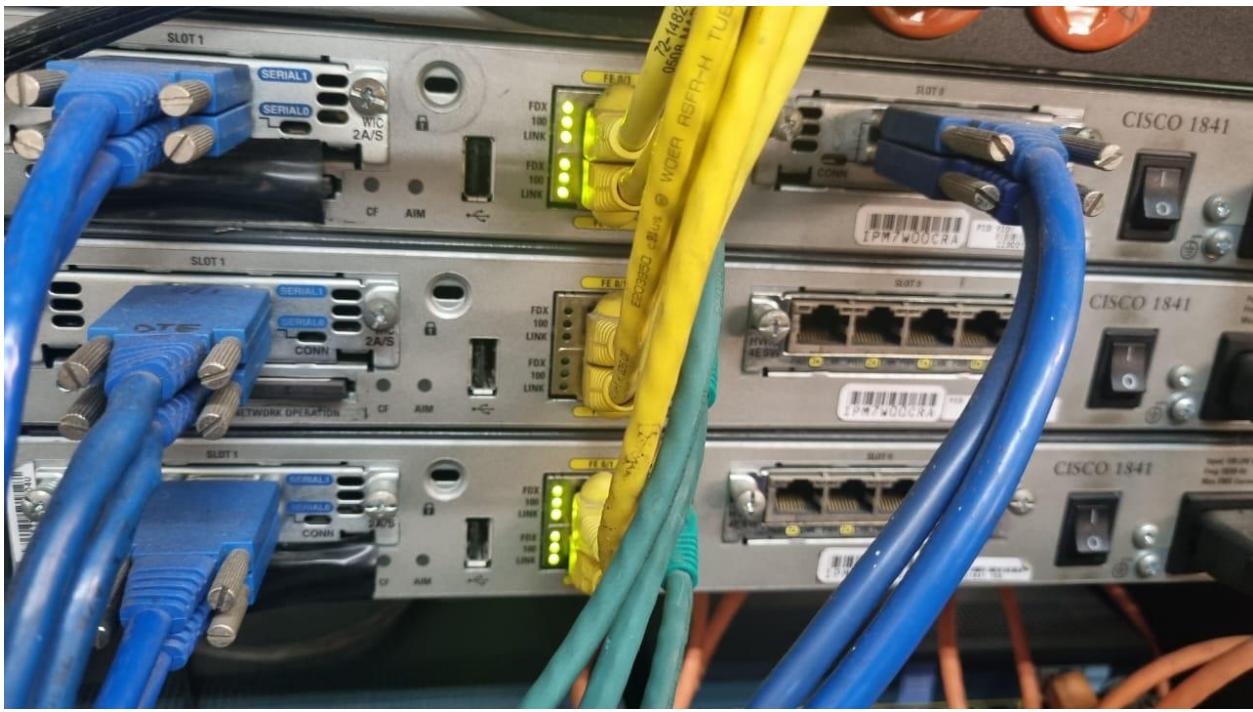


Figure 141 connection between devices an router

Now we connect the two routers using a serial cable.

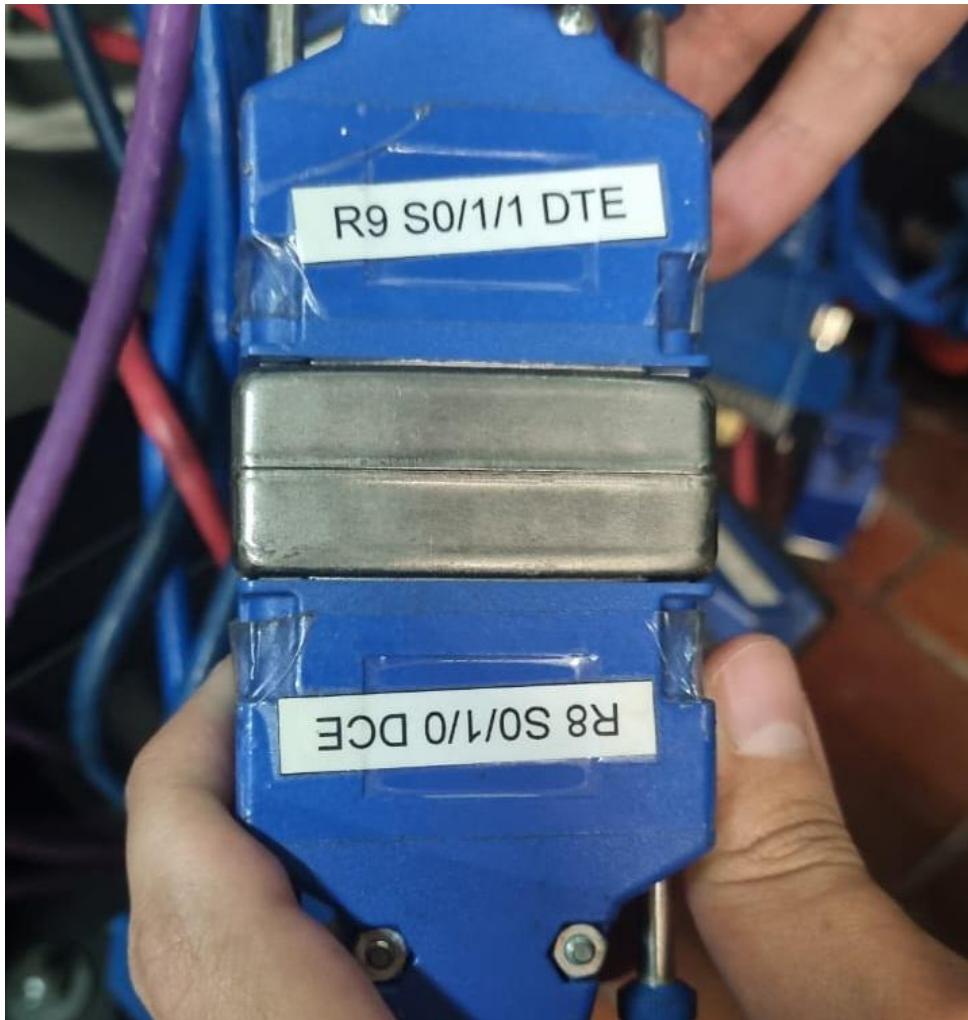


Figure 142 serial connection

We configure the network interfaces on each router.

Configuration of Router 8.

```
Torres#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Torres(config)#interface serial0/1/0
Torres(config-if)#ip address 100.0.0.1 255.255.255.0
Torres(config-if)#description "configuracion Router8"
Torres(config-if)#no shutdown
Torres(config-if)#End
```

Configuration of Router 9

```
vargas(config)#interface s0/0/1
vargas(config-if)#ip address 100.0.0.2 255.255.255.0
vargas(config-if)#description "Configuración entre Router5 y Router4"
vargas(config-if)#no shutdown
```

Figure 143 configuration interface s0/0/1

```
vargas#show ip interface brief
Serial0/0/1 100.0.0.2 YES SLARP up up
```

Figure 144 verification connection interfaces

```
vargas#ping 100.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Figure 145 test connection between routersStatic Routing

We configure static routes on each router so that when sending packets, the router knows which network to send them to.

```
Torres(config)#ip route 89.0.16.0 255.255.252.0 100.0.0.2
Torres(config)#ip route 89.0.4.0 255.255.255.0 100.0.0.2
```

Figure 146 static routes router 8

```
jorge(config)#ip route 88.0.4.0 255.255.255.0 100.0.0.1
jorge(config)#ip route 88.0.16.0 255.255.255.0 100.0.0.1
```

```
C:\Users\Redes>tracert 88.0.16.2
Tracing route to 88.0.16.2 over a maximum of 30 hops

 1 <1 ms <1 ms <1 ms 89.0.4.1
 2 1 ms 1 ms 1 ms 100.0.0.1
 3 2 ms 2 ms 2 ms 88.0.16.2

Trace complete.
```

Figure 147 trace route between devices

## 5. Closure

```
vargas#erase startup-config
Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
vargas#
*Nov 12 16:25:58.581: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
% Ambiguous command: "c"
vargas#write erase
Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
vargas#
*Nov 12 16:26:25.261: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvramreload

System configuration has been modified. Save? [yes/no]: yes
Building configuration...
[OK]
Proceed with reload? [confirm]

*Nov 12 16:26:45.505: %SYS-5-RELOAD: Reload requested by console. Reload Reason: Reload Command.
System Bootstrap, Version 15.0(1r)M16, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2012 by cisco Systems, Inc.

Total memory size = 512 MB - On-board = 512 MB, DIMM0 = 0 MB
CISCO2911/K9 platform with 524288 Kbytes of main memory
Main memory is configured to 72/-1(On-board/DIMM0) bit mode with ECC enabled

Readonly ROMMON initialized
program load complete, entry point: 0x80803000, size: 0xb340
program load complete, entry point: 0x80803000, size: 0xb340
```

## Conclusions

---

- **Network Management and Monitoring:** During this lab, we learned the importance of network monitoring in a corporate environment. We installed monitoring tools such as SNMP and configured network devices to track performance metrics such as CPU usage, disk space, and memory. Through this, we understood how these tools allow us to maintain network health and ensure efficient resource usage.
- **Router Configuration and Static Routing:** Configuring routers and implementing static routes was one of the key points. By doing this, we understood how networks that are not directly connected can communicate with each other. We used commands like ICMP and traceroute to check connectivity, which helped us visualize how data moves between devices and networks and how to ensure that the configured routes are working correctly.
- **ICMP Messages and Traceroute:** By experimenting with tools like traceroute and ICMP, we learned how to diagnose network routes and troubleshoot connectivity issues. These tools are very useful for network administrators, as they allow tracing the path of packets across the internet and local networks, making it easier to detect connectivity issues.
- **Router and Network Device Configuration:** By configuring routers and connecting network devices, we were able to understand the router boot process, the different types of memory, and the distinction between startup and running configuration files. This gave us a deeper understanding of how routers manage their configurations and how to maintain an optimally functioning network.
- **Serial Interconnection and Troubleshooting:** In this part of the lab, we learned how to interconnect routers using serial connections. We became familiar with commands like clock rate and understood the difference between DTE and DCE, which is essential for establishing network connections between distant devices. Additionally, we learned how to troubleshoot common issues that arise in these links.
- **Application Insights and Cloud Monitoring:** We worked with Azure and the Application Insights tool, which allowed us to explore how cloud monitoring platforms can be used to observe the performance of web applications and services. By enabling real-time monitoring, we were able to see how usage metrics and data can help identify potential issues and maintain good performance for cloud applications.

## References

---

- 3 installation from sources. (n.d.). Zabbix.com. Retrieved November 13, 2024, from <https://www.zabbix.com/documentation/current/en/manual/installation/install>
- AaronMaxwell. (n.d.-a). Application Insights overview. Microsoft.com. Retrieved November 13, 2024, from <https://learn.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview>
- AaronMaxwell. (n.d.-b). Diagnose with live metrics - Application Insights - Azure Monitor. Microsoft.com. Retrieved November 13, 2024, from <https://learn.microsoft.com/en-us/azure/azure-monitor/app/live-stream?tabs=otel>
- Al equipo, C. (2024, June 24). ¿Qué es un cable de módem nulo? Assured Systems. <https://www.assured-systems.com/es/faq/what-is-a-null-modem-cable/>
- Building and deploying Nagios on Oracle Solaris 11. (n.d.). Oracle.com. Retrieved November 13, 2024, from <https://www.oracle.com/technical-resources/articles/solaris/build-deploy-nagios-s11.html>
- Download Zabbix sources. (n.d.). Zabbix.com. Retrieved November 13, 2024, from [https://www.zabbix.com/download\\_sources](https://www.zabbix.com/download_sources)
- freeCodeCamp. (2020, December 21). The OSI model – the 7 layers of networking explained in plain English. Freecodecamp.org. <https://www.freecodecamp.org/news/osi-model-networking-layers-explained-in-plain-english/>
- Raza, M. (n.d.). OSI Model: The 7 layers of network architecture. BMC Blogs. Retrieved November 13, 2024, from <https://www.bmc.com/blogs/osi-model-7-layers/>