

Laboratorio No. 2 - Alistamiento de Sistemas operativos y Shell y Software de apoyo en redes

Integrantes: Angi Jimenez - Paola Cuellar - Daniela Ruiz.

Objetivo

- Continuar la instalación de sistemas operativos base.
- Conocer el modo de operación de herramientas de redes.
- Conocer sobre administración de sistemas operativos usando programas en Shell

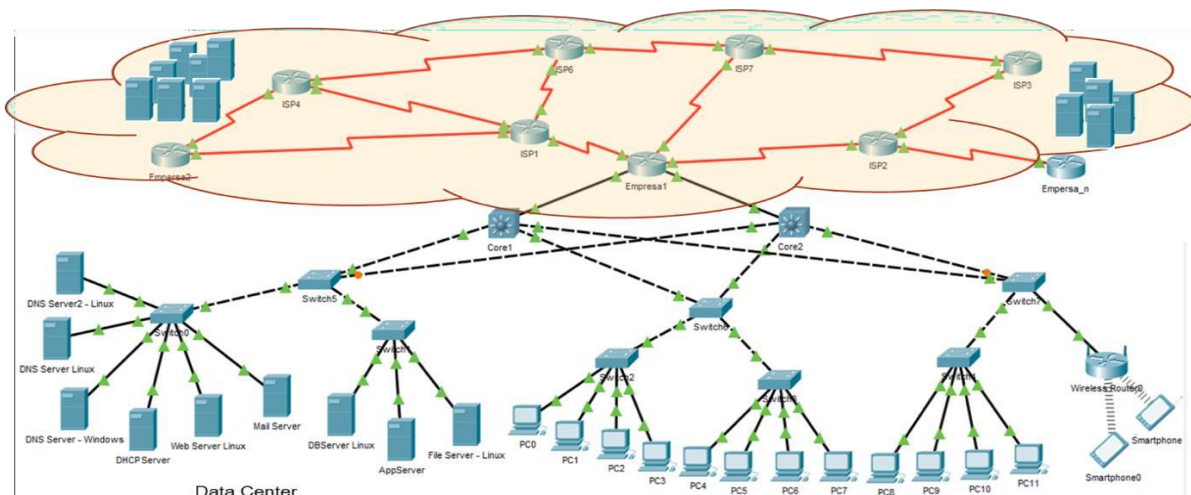
Herramientas a utilizar

- Computadores
- Acceso a Internet
- Software de virtualización
- Imágenes de Sistemas Operativos
- Packet tracer
- Wireshark

Introducción

Como ya hemos hablado, una empresa normalmente cuenta con varios servicios de infraestructura TI. En ella se encuentran estaciones de usuario alámbricas e inalámbricas y servidores (físicos y virtualizados), todos estos conectados a través de switches (capa 2 y 3), equipos inalámbricos y routers que lo conectan a Internet. También es común contar con infraestructuras en la nube desde donde se provisionan recursos según las necesidades de la organización. Dentro de los servidores se pueden encontrar servicios web, DNS, correo, base de datos, almacenamiento y aplicaciones, entre otros.

A continuación se presenta una posible configuración:



Nuestra introducción

Durante el desarrollo de este laboratorio haremos uso de virtualizadores como VMware y virtualBox que ya habían sido utilizados en el laboratorio pasado, esta vez con el fin de instalar maquinas virtuales con diferentes sistemas operativos en equipos de la universidad, accediendo a ellos de manera remota a través de openVPN. Por otro lado, conoceremos una gran herramienta que nos ofrece cisco, Packet Tracer, en la que podremos interactuar con diferentes dispositivos de red, conocer un poco a cerca de su configuración, como se ve físicamente, entre otras cosas. También analizaremos protocolos de red haciendo uso de una herramienta llamada Wireshark y por ultimo ahondaremos en el tema de shells, creando algunos programas shell en Slackware y CentOS.

Experimentos

Para construir una infraestructura tecnológica como la presentada en el dibujo anterior, se debe contar con computadores y servidores, los cuales tienen instalado un sistema operativo, también es importante conocer la operación de los mismos desde el punto de vista del administrador del sistema, así como apoyar procesos de automatización. A continuación, se plantean diferentes actividades enfocadas a conocer dicha estructura.

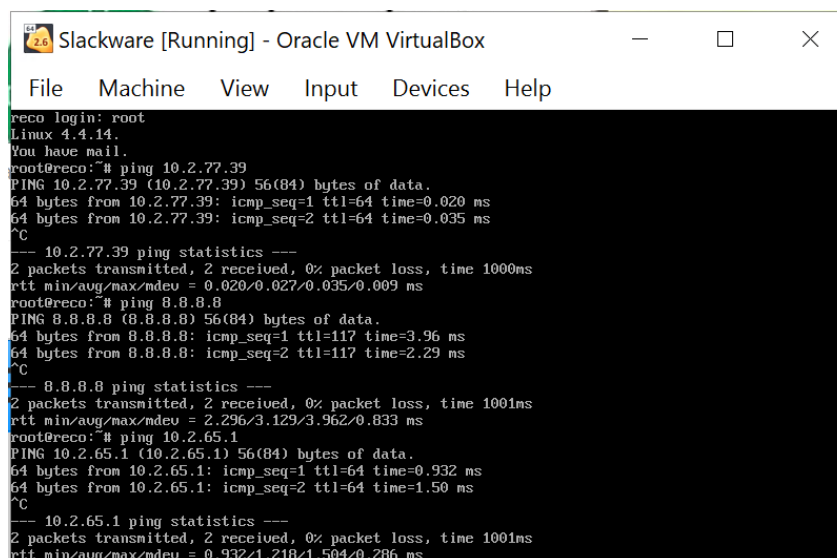
1. Pruebas de uso del Laboratorio de Informática

Vamos a realizar pruebas de operación de los equipos del Laboratorio de Informática. Para esto, se les enviará por grupos instrucciones de acceso a una máquina del Laboratorio. Luego de que haya entrado realice las siguientes actividades

- Instale Linux Slackware (instalación básica) usando VirtualBox en dicha máquina
- Configure la dirección IP así
 - o DIR_IP: 10.2.77.n
 - o Mascara: 255.255.0.0
 - o Gateway: 10.2.65.1
- Usando el comando ping, haga las siguientes pruebas
 - o ping 10.2.77.n
 - o ping 10.2.65.1
 - o ping 8.8.8.8
 - o ping 10.2.77.n1 (máquina de otros compañeros)

COMPUTADOR 39: ip 172.20.0.162

➤ Slackware:



```
Slackware [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@reco:~# login: root
Linux 4.4.14.
You have mail.
root@reco:~# ping 10.2.77.39
PING 10.2.77.39 (10.2.77.39) 56(84) bytes of data.
64 bytes from 10.2.77.39: icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from 10.2.77.39: icmp_seq=2 ttl=64 time=0.035 ms
^C
--- 10.2.77.39 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.020/0.027/0.035/0.009 ms
root@reco:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=3.96 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=2.29 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.296/3.129/3.962/0.833 ms
root@reco:~# ping 10.2.65.1
PING 10.2.65.1 (10.2.65.1) 56(84) bytes of data.
64 bytes from 10.2.65.1: icmp_seq=1 ttl=64 time=0.932 ms
64 bytes from 10.2.65.1: icmp_seq=2 ttl=64 time=1.50 ms
^C
--- 10.2.65.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.932/1.218/1.504/0.286 ms
```

Ping con la máquina de nuestros compañeros David y Juan Manuel, la numero 180:

```
root@reco:~# ping 10.2.77.180
PING 10.2.77.180 (10.2.77.180) 56(84) bytes of data.
64 bytes from 10.2.77.180: icmp_seq=1 ttl=64 time=0.999 ms
64 bytes from 10.2.77.180: icmp_seq=2 ttl=64 time=1.22 ms
64 bytes from 10.2.77.180: icmp_seq=3 ttl=64 time=1.24 ms
^C
--- 10.2.77.180 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.999/1.156/1.245/0.114 ms
root@reco:~# _
```

2. Conociendo Packet Tracer

- Responda las siguientes preguntas
 1. ¿Qué versión de Packet Tracer se encuentra instalada en el Lab?

7.2.2
 2. A través de la plataforma de Cisco inscribise en el curso Introduction to Packet Tracer v1.1. muestre con un video hecho por el grupo un resumen del curso. Máximo 10 min.


Link del video:

https://pruebacorreoescuelaingeducomy.sharepoint.com/:v:/g/personal/angie_ruiz_mail_escuelaing_edu_co/ERe_0cXYzc9Bpq-DDg506P4B9xTc9mUKPqU1dI1IsqPwxA?e=egjNyC

3. Realice la evaluación del curso y tome un pantallazo del resultado de la evaluación.

➤ Daniela

Grades for Angie Daniela Ruiz Alfonso

 Print Grades

Course

Arrange By

Introduction to Packet

Due Date

Apply

Name	Due	Status	Score	Out of
End of Course Feedback			✓	100
Assignments			N/A	0.00 / 0.00
Course Completion			100%	100.00 / 100.00
Total			100%	

➤ Angi

Grades for Angi Paola Jimenez Pir

Print Grades

Total: 100%

Show Saved "What-If" Scores

Show All Details

Assignments are weighted by group:

Group	Weight
Assignments	0%
Course Completion	100%
Total	100%

☒ Calculate based only on graded assignments

You can view your grades based on What-If scores so that you know how grades will be affected by upcoming or resubmitted

Course: Introduction to Packet Arrange By: Due Date Apply

Name	Due	Status	Score	Out of
End of Course Feedback		✓	100	
Assignments			N/A	0.00 / 0.00
Course Completion			100%	100.00 / 100.00
Total			100%	

➤ Paola

Grades for Paola Andrea Cuellar Lopez

Print Grades

Total: 100%

Show All Details

Assignments are weighted by group:

Group	Weight
Assignments	0%
Course Completion	100%
Total	100%

☒ Calculate based only on graded assignments

You can view your grades based on What-If scores so that you know how grades will be affected by upcoming or resubmitted assignments. You can test scores for an assignment that already includes a score, or an assignment that has yet to be graded.

Course: Introduction to Packet Arrange By: Due Date Apply

Name	Due	Status	Score	Out of
End of Course Feedback		✓	100	
Assignments			N/A	0.00 / 0.00
Course Completion			100%	100.00 / 100.00
Total			100%	

- Usando Packet Tracer haga el diagrama de red que se presenta en la página siguiente.

Nota:

No tenga en cuenta los colores de los puntos que aparecen en los enlaces (los enlaces son las líneas de conexión entre dispositivos. Más adelante serán importantes los colores de dichos puntos, pero en su momento los revisaremos.

- Las conexiones o enlaces que se presentan en el diagrama son:
 - Las de color negro corresponden a cables Ethernet (Ethernet, FastEthernet o GigaEthernet).

- ¿Qué significan las conexiones negras continuas?

Las líneas negras continuas son cables *Straight Through*, un tipo de cable de par trenzado que normalmente se usa en redes de área local para conectar una computadora a un concentrador de red como un router. Este cable usa un estándar de cableado: ambos extremos usan el estándar de cableado T568A o ambos extremos usan el estándar de cableado T568B.

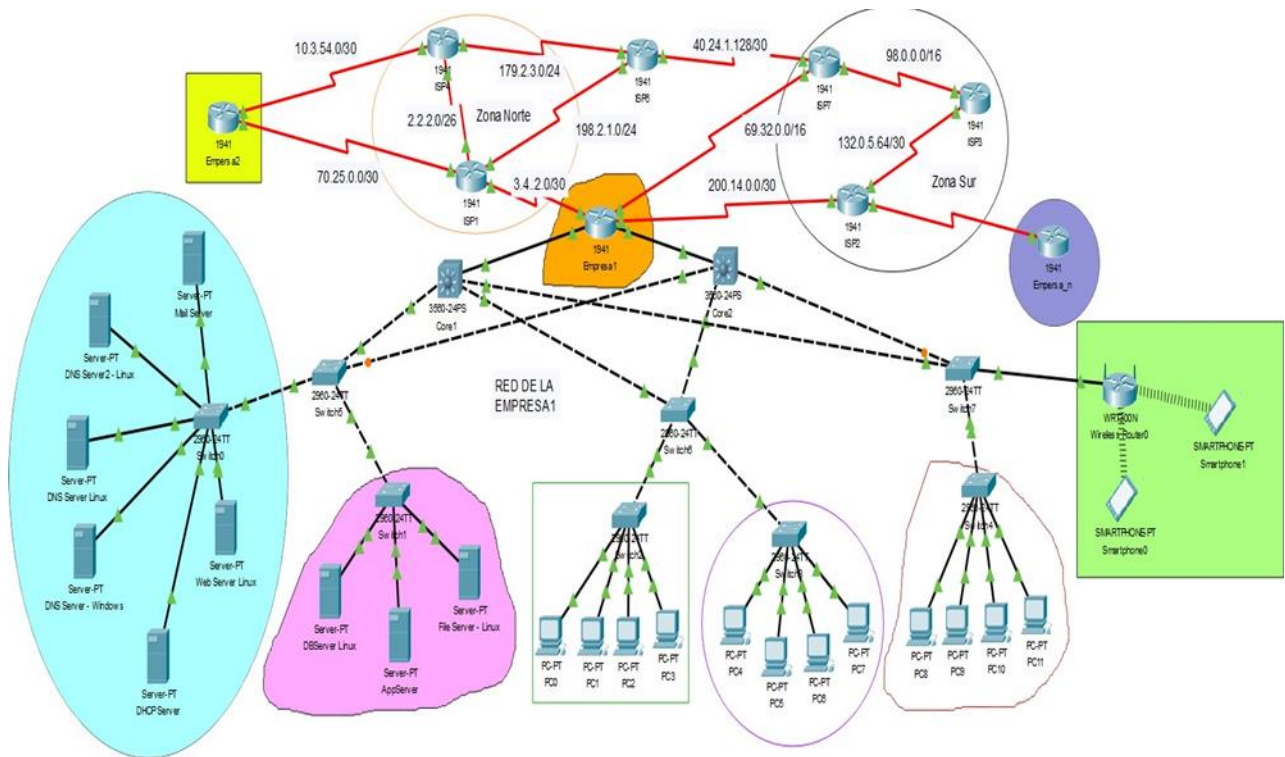
- ¿Qué significan las conexiones negras discontinuas?

Las líneas negras discontinuas son cables *Cross-Over*, un tipo de cable Ethernet utilizado para conectar dispositivos informáticos normalmente del

mismo tipo, por ejemplo: dos computadoras, dos switches, etc.

A diferencia del *Straight Through*, este cable usa dos estándares de cableado diferentes: un extremo usa el estándar de cableado T568A y el otro extremo usa el estándar de cableado T568B.

- Las de color rojo son seriales (Conexiones típicamente WAN). Al dibujarlas en packet tracer aparecerán un poco diferentes respecto al dibujo.



Realizamos la red procurando que quedara lo más parecida posible a la de la imagen, sin embargo en la parte de arriba, donde se usan cables de serie, intentamos configurar los routers de manera que la conexión funcionara (que fuera de color verde), en algunos casos funcionó, en otros no aceptaba la ip que se encuentra en la imagen, por este motivo, algunas conexiones se encuentran en rojo.

3. Siguiendo mensajes con Packet tracer

- Seleccione dos computadores ubicados en el cuadrado verde no relleno y el círculo morado no relleno. Póngales la siguiente configuración
 - PC red cuadrada (PCA)
 - IP 10.0.0.2
 - Máscara: 255.0.0.0
 - PC red círculo (PCB)
 - IP 10.0.0.3
 - Máscara 255.0.0.0
- Entre en el modo simulación con que cuenta Packet Tracer y revise los PDUs por capas (Todavía no hemos visto el significado de lo que cada uno tiene, pero vea que existen y que cada capa adiciona información a los datos de usuario). Para esto use la siguiente información como guía

Run the simulation and capture the traffic¹.

- In the far lower right of the PT interface is the toggle between Realtime and Simulation mode. Click on Simulation mode.
- Click in the Edit filters button and select only ICMP.
- Click in PCA. Choose the Desktop tab. Open the Command Prompt. Enter the command ping IP_PCB). Pressing the Enter key will initiate four ICMP echo requests. Minimize the PC configuration window. Two packets appear in the Event List, the first ICMP echo request and an ARP request needed to resolve the IP address of the server to its hardware MAC address.
- Click the Auto Capture / Play button to run the simulation and capture events. Click OK when the "No More Events" message is reached.

- Revise el contenido de los paquetes capturados. Revise el contenido del encabezado de capa 2.

Montaje real

Realice las siguientes pruebas usando la herramienta Wireshark.

1. Usando Wireshark

Wireshark es una herramienta multiplataforma utilizada para realizar análisis sobre paquetes de red². La utilizaremos dentro del curso para observar, en tiempo real, lo datos que pasan por la red y la manera de operación de los diferentes protocolos que estudiaremos. Por tal razón

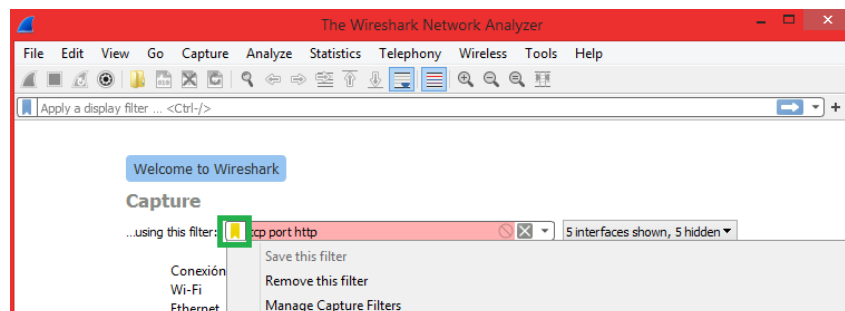
- Ejecute Wireshark en el computador en el que está trabajando
- Revise los siguientes videos
 - Wireshark Tutorial for Beginners.
<https://www.youtube.com/watch?v=TkCSr30UojM>.
 - Wireshark Tutorial for Beginners 2017 - Overview of the environment.
<https://www.youtube.com/watch?v=6LGw31TsP6E>.
 - Wireshark demo (simple http). <https://www.youtube.com/watch?v=PYoXowOCppc>.

- ¿Qué es Wireshark?

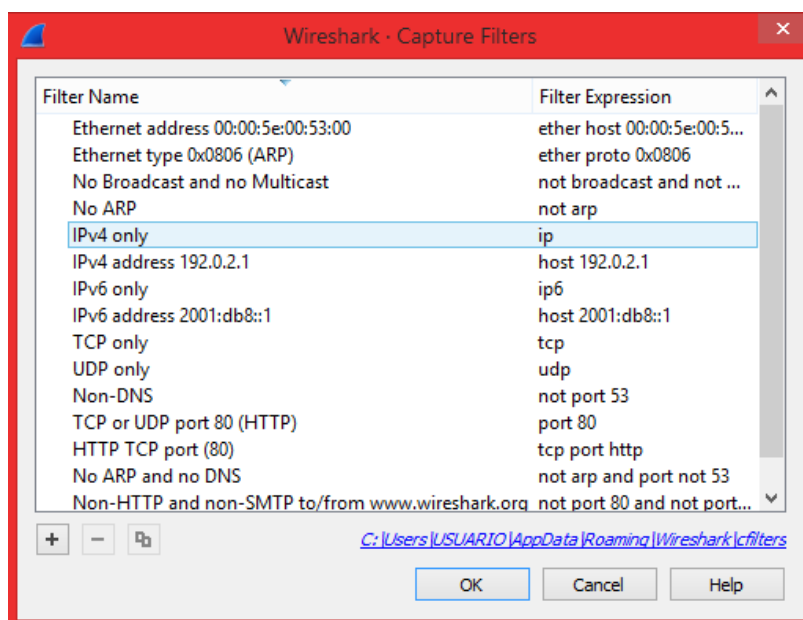
Wireshark es una herramienta multiplataforma que permite el análisis de protocolos sobre paquetes de red. Es la continuación de un proyecto iniciado por Gerald Combs en 1998.

- ¿Cómo generar filtros?

- Para generar filtros estando dentro de la herramienta de Wireshark, seleccionamos el icono amarillo al lado izquierdo de la barra de filtros y seleccionamos "Manage Capture Filters"

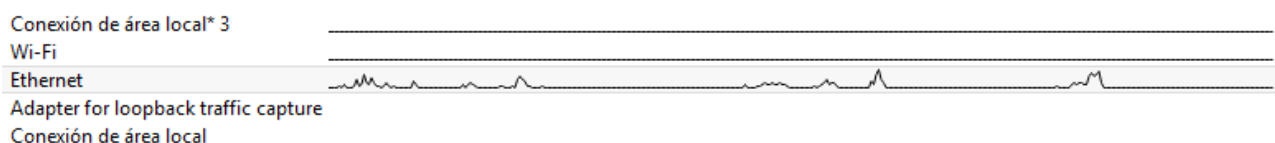


- Nos abrirá la siguiente ventana; se listan todos los nombres de filtros que existen en el momento y la expresión que tiene en cuenta para realizar el filtro, cuando agregamos uno nuevo debemos darle un nombre al filtro y una expresión que corresponda a la información que deseamos filtrar.



- ¿Para qué se usan?
Los filtros nos permiten obtener o visualizar una información específica sobre el tráfico de paquetes, así podemos hacer el seguimiento de uno de estos; sino es aplicado ningún filtro, nos mostrará la información correspondiente a todo el tráfico de paquetes en la pantalla principal.
- De unos ejemplos
 - Se debe tener en cuenta en el filtro se usa src haciendo referencia al origen y dst como referencia al destino. Se puede filtrar por host (host), puerto (port), protocolo IP (ip), protocolo Ethernet (ether) y red (net).
 - Capturar todos los paquetes con origen y destino 149.56.16.159
host 149.56.16.159
 - Capturar los paquetes con puerto origen 80 port src port 80
- Realice una consulta web al link <http://campusvirtual.escuelaing.edu.co/> y capture el tráfico generado (para eso, ingrese al browser, inicie la captura con Wireshark y visite a la página indicada, termine la captura). Finalmente, pare la captura.

La siguiente imagen visualiza las redes que se encuentran disponibles en el dispositivo; Para el ejercicio, se captura el tráfico generado en la red de Ethernet. Y a continuación se encuentra una imagen que presenta la captura total de datos filtradas por http.



No.	Time	Source	Destination	Protocol	Length	Info
59	3.628687	192.168.1.6	149.56.16.159	HTTP	561	GET /moodle/ HTTP/1.1
124	4.146022	192.168.1.6	149.56.16.159	HTTP	565	GET /moodle/theme/styles.php/essential/1592190001/all HTTP/1.1
132	4.153942	149.56.16.159	192.168.1.6	HTTP	1028	HTTP/1.1 200 OK (text/html)
291	4.749269	149.56.16.159	192.168.1.6	HTTP	1396	HTTP/1.1 200 OK (text/css)
296	4.881098	192.168.1.6	149.56.16.159	HTTP	548	GET /theme/essential/style/fonts.css HTTP/1.1
297	4.881318	192.168.1.6	149.56.16.159	HTTP	550	GET /moodle/theme/jquery.php/core/jquery-3.1.0.min.js HTTP/1.1
298	4.881373	192.168.1.6	149.56.16.159	HTTP	597	GET /moodle/theme/yui_combo.php?rollup/3.17.2/yui-moodlesimple-min.js&rollup/1592190001/mcore-min.js HTTP/1.1
302	4.896983	192.168.1.6	149.56.16.159	HTTP	555	GET /moodle/theme/jquery.php/theme_essential/custom_1.0.js HTTP/1.1
303	4.896983	192.168.1.6	149.56.16.159	HTTP	556	GET /moodle/theme/jquery.php/theme_essential/cslider_1.0.js HTTP/1.1
307	5.004166	149.56.16.159	192.168.1.6	HTTP	1158	HTTP/1.1 200 OK (text/css)
336	5.056874	149.56.16.159	192.168.1.6	HTTP	1006	HTTP/1.1 200 OK (application/javascript)
340	5.057481	149.56.16.159	192.168.1.6	HTTP	812	HTTP/1.1 200 OK (application/javascript)
349	5.062506	192.168.1.6	149.56.16.159	HTTP	574	GET /moodle/theme/jquery.php/theme_essential/bootstrap_plugins/alert_2.3.2.js HTTP/1.1
371	5.149673	192.168.1.6	149.56.16.159	HTTP	647	GET /theme/essential/style/fonts/ATKpV8nLYAKUvexo8iqqrg.woff2 HTTP/1.1
388	5.183012	149.56.16.159	192.168.1.6	HTTP	379	HTTP/1.1 200 OK (application/javascript)
414	5.197827	149.56.16.159	192.168.1.6	HTTP	230	HTTP/1.1 200 OK (application/javascript)
436	5.208642	192.168.1.6	149.56.16.159	HTTP	577	GET /moodle/theme/jquery.php/theme_essential/bootstrap_plugins/carousel_2.3.2.js HTTP/1.1
440	5.232219	192.168.1.6	149.56.16.159	HTTP	577	GET /moodle/theme/jquery.php/theme_essential/bootstrap_plugins/collapse_2.3.2.js HTTP/1.1
509	5.332171	149.56.16.159	192.168.1.6	HTTP	211	HTTP/1.1 200 OK (application/javascript)
515	5.343845	149.56.16.159	192.168.1.6	HTTP	1285	HTTP/1.1 200 OK (application/javascript)
523	5.362572	192.168.1.6	149.56.16.159	HTTP	578	GET /moodle/theme/jquery.php/theme_essential/bootstrap_plugins/scrollspy_2.3.2.js HTTP/1.1

▶ Frame 349: 574 bytes on wire (4592 bits), 574 bytes captured (4592 bits) on interface \Device\NPF_{E2B8085B-485B-4284-9716-1FEE9C0A108}, id 0
 ▶ Ethernet II, Src: QuantaCo_75:f3:d2 (08:9e:01:75:f3:d2), Dst: MitraSta_ab:42:78 (cc:d4:a1:ab:42:78)
 ▶ Internet Protocol Version 4, Src: 192.168.1.6, Dst: 149.56.16.159
 ▶ Transmission Control Protocol, Src Port: 63545, Dst Port: 80, Seq: 1, Ack: 1, Len: 520
 ▶ Hypertext Transfer Protocol

```

0000  cc d4 a1 ab 42 78 08 9e 01 75 f3 d2 08 00 45 00  ....Bx...u....E
0010  02 30 28 04 40 00 06 00 00 c0 a8 01 06 95 38  0(@.....8
0020  10 9f f8 39 00 50 21 a5 b4 32 a6 33 fc 76 50 18  ...9Pl...23vP
0030  fd 30 69 a8 00 00 47 a5 54 20 2f 6d 6f 64 6c    i-GET/moodl
0040  65 2f 74 68 65 6d 65 2f 6a 71 75 65 72 79 2e 70  e/theme/jquery.p
0050  68 70 2f 74 68 65 6d 65 5f 65 73 73 65 6e 74 69  hp/theme_essenti
0060  61 6c 2f 62 6f 6f 74 73 74 72 61 70 5f 70 6c 75  al/boots trap plu
  
```

- Analice los datos encontrados en uno de los paquetes capturados. Mire el encapsulamiento y presente capturas del mismo (Use el paquete que contiene una de las solicitudes GET que se realizan).
- **En la pestaña de “frame”:** el paquete seleccionado corresponde al número 298, en la información podemos evidenciar información como tipo de encapsulamiento: Ethernet, La fecha y lugar donde se solicita el paquete, el tiempo de captura y envío del paquete, el tamaño del paquete que corresponde a 597 bytes (4776 bits), adicionalmente dos banderas que nos dicen que no fue marcado ni ignorado el frame, el protocolo de este y el nombre de la regla que corresponde a http.

No.	Time	Source	Destination	Protocol	Length	Info
298	4.881373	192.168.1.6	149.56.16.159	HTTP	597	GET /moodle/theme/yui_combo.php?rollup/3.17.2/yui-moodlesimple-min.js&rollup/1592190001/mcore-min.js HTTP/1.1

▶ Frame 298: 597 bytes on wire (4776 bits), 597 bytes captured (4776 bits) on interface \Device\NPF_{E2B8085B-485B-4284-9716-1FEE9C0A108}, id 0
 ▶ Interface id: 0 (\Device\NPF_{E2B8085B-485B-4284-9716-1FEE9C0A108})
 Encapsulation type: Ethernet (1)
 Arrival Time: Jun 14, 2020 23:02:37.694653000 Hora est. Pacífico, Sudamérica
 [Time shift for this packet: 0.000000000 seconds]
 Epoch Time: 1592193757.694653000 seconds
 [Time delta from previous captured frame: 0.000055000 seconds]
 [Time delta from previous displayed frame: 0.000055000 seconds]
 [Time since reference or first frame: 4.881373000 seconds]
 Frame Number: 298
 Frame Length: 597 bytes (4776 bits)
 Capture Length: 597 bytes (4776 bits)
 [Frame is marked: False]
 [Frame is ignored: False]
 [Protocols in frame: eth:ethertype:ip:tcp:http]
 [Coloring Rule Name: HTTP]
 [Coloring Rule String: http || tcp.port == 80 || http2]

- En la pestaña de “Ethernet” donde nos especifica el destino, el origen y el tipo de conexión (IPv4), y adicionalmente aparecen las pestañas de “Internet Protocol Version 4”, “Transmission Control Protocol” e “Hypertext Transfer Protocol” en el que nos informa el host con el que podemos confirmar que fue en la página de campusvirtual.escuelaing.edu.co/moodle/

No.	Time	Source	Destination	Protocol	Length	Info
298	4.881373	192.168.1.6	149.56.16.159	HTTP	597	GET /moodle/theme/yui_combo.php?rollup/3.17.2/yui-moodlesimple-min.js&rollup/1592190001/mcore-min.js HTTP/...
Ethernet II, Src: QuantaCo_75:f3:d2 (08:9e:01:75:f3:d2), Dst: MitraSta_ab:42:78 (cc:d4:a1:ab:42:78) <ul style="list-style-type: none"> Destination: MitraSta_ab:42:78 (cc:d4:a1:ab:42:78) Source: QuantaCo_75:f3:d2 (08:9e:01:75:f3:d2) Type: IPv4 (0x0800) 						
Internet Protocol Version 4, Src: 192.168.1.6, Dst: 149.56.16.159						
Transmission Control Protocol, Src Port: 63541, Dst Port: 80, Seq: 1, Ack: 1, Len: 543						
Hypertext Transfer Protocol						
GET /moodle/theme/yui_combo.php?rollup/3.17.2/yui-moodlesimple-min.js&rollup/1592190001/mcore-min.js HTTP/1.1\r\n Host: campusvirtual.escuelaing.edu.co\r\n Connection: keep-alive\r\n User-Agent: Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.97 Safari/537.36\r\n Accept: */*\r\n Referer: http://campusvirtual.escuelaing.edu.co/moodle/\r\n Accept-Encoding: gzip, deflate\r\n Accept-Language: es-US,es;q=0.9,es-419;q=0.8,en;q=0.7\r\n Cookie: _ga=GAl.4.1754274999.1591649555; MoodleSession=s26e7vpcv468o5sihuirbf4a3\r\n \r\n [Full request URI: http://campusvirtual.escuelaing.edu.co/moodle/theme/yui_combo.php?rollup/3.17.2/yui-moodlesimple-min.js&rollup/1592190001/mcore-min.js] [HTTP request 1/1] [Response in frame: 509]						

Software Base

Dentro de la infraestructura también se requiere contar con programas que apoyen la administración de diferentes actividades del sistema operativo. Vamos a realizar actividades que les ayuden a entender un poco el sistema operativo y la gestión de usuarios.

Bourne Shell programming- Unix

Usando las máquinas virtuales Linux que instaló para el curso, desarrolle las siguientes aplicaciones (recuerde documentar su código). Revise que operen en ambas distribuciones de Linux.

1. Escriba un programa Shell que:

- Limpie la pantalla

➤ Slackware:

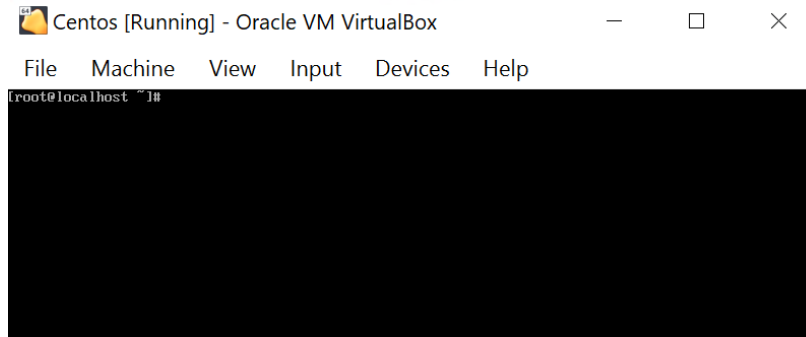
Iniciaremos la máquina virtual, posteriormente ingresaremos el comando **vi limpiar.sh**, presionamos la tecla **i** y escribiremos el comando para limpiar la terminal el cual es **clear**, luego presionamos la tecla **esc** e ingresamos el comando **:wq** para guardar el archivo y salir del editor. Ahora debemos ingresar el comando **chmod +x limpiar.sh** para dar permisos de ejecución del archivo y luego digitaremos **./limpiar.sh** para ejecutar el programa y veremos que se limpia la terminal.

En el caso de Slackware no reconoce el comando dentro del ejecutable de **clear**, ni **cls**, ni **clear-screen**, ni **reset**, ni **tput clear**, entonces suponemos por consultas web sobre el tema que es algo que no está en la instalación mínima y por ende no se puede realizar y no queremos hacer uso de nano.

```
root@slackware:~/prograns# _
```

➤ CentOS:

Se inicia la maquina y cuando ingresemos con nuestro usuario root, crearemos en el editor VI un archivo ejecutable llamado limpiar usando el comando vi **limpiar.sh**, ahora escribiremos **clear**, ahora presionamos e para que se “escriba” o sea nos guarde los cambios en el archivo usando el comando y presionamos **i** para insertar esta palabra, luego para entrar al modo comando de VI presionamos la tecla **esc** y presionamos **shift+zz** para que nos regrese a la consola de la máquina. Se le asignan los permisos de ejecución al archivo con el comando **chmod +x limpiar.sh** para que ejecute el archivo usando el comando **./limpiar.sh**



- Permita, con un menú, hacer una de las siguientes actividades
 1. Muestre las últimas 20 líneas del archivo messages
El comando **tail -f messages** muestra las primeras 10 líneas del archivo messages
 2. Muestre las últimas 20 líneas del archivo messages que contengan una palabra particular
El comando **tail -f messages | grep error** muestra las primeras 10 líneas del archivo messages que tengan la palabra error
 3. Muestre las últimas 20 líneas del archivo syslog
El comando **tail -f syslog** muestra las primeras 10 líneas del archivo syslog
 4. Muestre las últimas 20 líneas del archivo syslog que contengan una palabra particularE
El comando **tail -f syslog | grep error** muestra las primeras 10 líneas del archivo syslog que tengan la palabra error

➤ Slackware:

Seguimos el mismo procedimiento que realizamos para crear el programa que limpia la pantalla, una vez estemos dentro del editor vi no escribiremos **clear** sino que empezamos a escribir el programa que cumple con el objetivo establecido en el enunciado, para ello haremos uso de los comandos consultados; una vez escrito el programa daremos permisos de ejecución con el comando **chmod +x menu.sh** y ejecutaremos con **./menu.sh**.

1.

```
root@slackware:~/programs# ./menu.sh
-----
MENU
-----
1. Muestre las ultimas 20 lineas del archivo messages
2. Muestre las ultimas 20 lineas del archivo messages
   que contengan una palabra particular
3. Muestre las ultimas 20 lineas del archivo syslog
4. Muestre las ultimas 20 lineas del archivo syslog
   que contengan una palabra particular
-----

Seleccione una opcion:2
Ingrese la palabra:
drm
Resultado:
[ 7.431303] [drm] height 400
[ 7.431772] [drm] bpp 32
[ 7.432720] [drm] Fifo max 0x00200000 min 0x00001000 cap 0x00000355
[ 7.436254] [drm] DX: no.
[ 7.484994] fbcon: sugadrmfb (fb0) is primary device
[ 7.520699] [drm] Initialized vmwgfx 2.9.0 20150810 for 0000:00:02.0 on minor 0
root@slackware:~/programs#
```

2. Las funciones 3 y 4 del menú no funcionan ya que no encontramos la ubicación del archivo syslog, por lo tanto no pudimos poner la ruta.

➤ CentOS:

El proceso para la creación del archivo es muy parecida a la de slackware

```
=====
MENU
=====
1. Muestre las ultimas 20 lineas del archivo messages
2. Muestre las ultimas 20 lineas del archivo messages
   que contengan una palabra particular
3. Muestre las ultimas 20 lineas del archivo syslog
4. Muestre las ultimas 20 lineas del archivo syslog
   que contengan una palabra particular
Seleccione una opcion:1
Resultado:
Jun 17 00:28:25 localhost systemd: Starting OpenSSH server daemon...
Jun 17 00:28:25 localhost systemd: Starting Crash recovery kernel arming...
Jun 17 00:28:25 localhost kdumpctl: No memory reserved for crash kernel
Jun 17 00:28:25 localhost kdumpctl: Starting kdump: [FAILED]
Jun 17 00:28:25 localhost systemd: kdump.service: main process exited, code=exited, status=1/FAILURE
Jun 17 00:28:25 localhost systemd: Failed to start Crash recovery kernel arming.
Jun 17 00:28:25 localhost systemd: Unit kdump.service entered failed state.
Jun 17 00:28:25 localhost systemd: kdump.service failed.
Jun 17 00:28:26 localhost systemd: Started OpenSSH server daemon.
Jun 17 00:28:26 localhost rsyslogd: [origin software="rsyslogd" swVersion="8.24.0-34.el7" x-pid="697
9" x-info="http://www.rsyslog.com"] start
Jun 17 00:28:26 localhost systemd: Started System Logging Service.
Jun 17 00:28:27 localhost systemd: Started Postfix Mail Transport Agent.
Jun 17 00:28:29 localhost systemd: Started Dynamic System Tuning Daemon.
Jun 17 00:28:29 localhost systemd: Reached target Multi-User System.
Jun 17 00:28:29 localhost systemd: Starting Update UTMP about System Runlevel Changes...
Jun 17 00:28:29 localhost systemd: Started Update UTMP about System Runlevel Changes.
Jun 17 00:28:29 localhost systemd: Startup finished in 1.127s (kernel) + 7.269s (initrd) + 29.247s (
userspace) = 37.644s.
Jun 17 00:28:39 localhost systemd: Created slice User Slice of root.
Jun 17 00:28:39 localhost systemd: Started Session 1 of user root.
Jun 17 00:28:39 localhost systemd-logind: New session 1 of user root.
[root@localhost lab2]#
```

2. Escriba un programa Shell que implemente el trabajo que hizo de creación de usuarios, grupos y permisos del laboratorio anterior

```

[root@localhost lab2]# ./script2
[root@localhost lab2]# cat script2
#creacion de grupos
    groupadd desarrollo
    groupadd soporte

#creacion de usuarios

useradd angie -c "primeera integrante" -g soporte -d /usr/angie -s "/bin/bash"
useradd andrea -c "segunda nombre de Paola" -g soporte -d /usr/andrea -s "/bin/bash"
useradd daniela -c "segunda integrante" -g desarrollo -d /usr/daniela -s "/usr/bin/bash"
useradd paola -c "tercera integrante" -g desarrollo -d /usr/paola -s "/usr/bin/sh"

```

3. Dada una palabra y una ruta de arranque, busque todos los archivos que contengan dicha palabra. La salida debe ser de la siguiente forma

```

*****
*          BUSQUEDA DE ARCHIVOS PARABRA palabra_buscada          *
***** DIRECTORIO 1:

                path
ARCHIVOS: archivo1; archivo2; archivo3; ... ; archivon
DIRECTORIO 2:    path
ARCHIVOS: archivo1; archivo2; archivo3; ... ; archivon
DIRECTORIO 3:    path
ARCHIVOS: archivo1; archivo2; archivo3; ... ; archivon
.
.
.
DIRECTORIO n:    path
ARCHIVOS: archivo1; archivo2; archivo3; ... ; archivon

```

```

[root@localhost lab2]# ./script3
Ingrese su palabra: localhost
Ingrese su ruta de arranque: /var/log/messages
*****
*          BUSQUEDA DE ARCHIVOS PALABRA localhost          *
*****
1:Jun 16 21:21:02 localhost rsyslogd: [origin software="rsyslogd" swVersion="8.24.0-34.el7" x-pid="6991" x-info="http://www.rsyslog.com"] rsyslogd was HUPed
2:Jun 16 21:33:25 localhost systemd: Got automount request for /proc/sys/fs/binfmt_misc, triggered by 7340 (find)
3:Jun 16 21:33:25 localhost systemd: Mounting Arbitrary Executable File Formats File System...
4:Jun 16 21:33:25 localhost systemd: Mounted Arbitrary Executable File Formats File System.
5:Jun 16 22:01:01 localhost systemd: Started Session 3 of user root.
6:Jun 16 23:01:02 localhost systemd: Started Session 4 of user root.
7:Jun 16 23:08:38 localhost su: (to paola) root on tty1
8:Jun 16 23:22:19 localhost su: (to angie) root on tty1
9:Jun 16 23:29:24 localhost su: (to angie) root on tty1
10:Jun 16 23:30:45 localhost su: (to angie) root on tty1
11:Jun 16 23:38:11 localhost su: (to andrea) root on tty1
12:Jun 16 23:40:20 localhost su: (to angie) root on tty1
13:Jun 16 23:40:20 localhost su: (to paola) root on tty1
14:Jun 16 23:40:20 localhost su: (to daniela) root on tty1
15:Jun 16 23:40:21 localhost su: (to andrea) root on tty1

```

NOTA: Muestre a su profesor la ejecución de sus programas.

Referencias:

Usos y aplicaciones

En este laboratorio tratamos varios temas que tienen aplicaciones en la vida real, tal vez el más fácil de ver es el que tiene que ver con Packet Tracer ya que esta herramienta se encarga principalmente de simular montajes de red y ya tiene una gran aplicación que es en la educación, pero además esto se puede usar para simular redes que se van a construir en realidad; por otro lado, los programas de shells pueden ser usados a la hora de administrar archivos, estos podrían hacerse con el fin de facilitarnos tareas que podrían resultar tediosas a la hora de realizarlas manualmente. Con Wireshark podemos realizar análisis en tiempo real sobre paquetes (generales o específicos) de redes que tenemos implementadas en el mundo real.

Conclusiones

Aprendimos a conectarnos mediante VPN a un servidor remoto en este caso a los computadores físicos de la escuela desde casa, estudiamos y entendimos el manejo básico de las herramientas Packet Tracer y Wireshark para el tema del funcionamiento de la red, además aprendimos a realizar programas ejecutables básicos en shell en Linux usando el editor VI el cual es muy importante en el mundo laboral, adicionalmente en el transcurso que realizábamos dichos programas aprendimos un poco más sobre el sistema operativo Linux, la forma como se estructuran sus paquetes.

Referencias

- Alfon. (24 de Marzo de 2008). *Análisis de red con Wireshark. Filtros de captura y visualización*. Obtenido de <https://seguridadyredes.wordpress.com/2008/03/24/analisis-de-red-con-wireshark-filtros-de-captura-y-visualizacia/>
- Cables-Solutions.com. (21 de Diciembre de 2016). *Fiber Optic Cabling Solutions*. Obtenido de <http://www.cables-solutions.com/difference-between-straight-through-and-crossover-cable.html>
- Catoira, F. (28 de Enero de 2013). *Uso de filtros en Wireshark para detectar actividad maliciosa*. Obtenido de <https://www.welivesecurity.com/la-es/2013/01/28/uso-filtros-wireshark-para-detectar-actividad-maliciosa/>
- Wireshark. (s.f.). *Learn Wireshark*. Obtenido de <https://www.wireshark.org/>