
	<p><i>Universidad Mayor de San Andrés</i> <i>Facultad de Ciencias Puras y Naturales</i> <i>Carrera de Informática</i></p>	
	<p>317 - PROGRAMACIÓN DISTRIBUIDA Y PARALELA</p>	
	<p>Docente: Ph. D. Moises Martín Silva Choque</p>	
	<p>Apellidos y Nombre: Jael Lany Valeriano Baltazar C.I.: 9945924 LP</p>	
	<p>Fecha: 26 - 04 - 2023</p>	
	<p>PRIMER EXAMEN PARCIAL</p>	

1. Describa cada una de las siguientes: SISD, SIMD, MISD y MIMD.

R. SISD, SIMD, MISD y MIMD, son una clasificación de sistemas paralelos, propuestas por Michael J. Flynn en 1966.

SISD (*Single Instruction/Single Data*): consiste en un flujo único de instrucciones y un único flujo de datos. De acuerdo a Von Neumann, en cualquier momento, sólo se está ejecutando una única instrucción. Una SISD posee un registro simple llamado *contador de programas* que asegura la ejecución en serie del programa.

SIMD (*Single Instruction stream, Multiple Data stream*): consiste en un flujo de instrucción simple y flujo de datos múltiple, esto significa que una única instrucción es aplicada sobre diferentes datos al mismo tiempo. En las máquinas de este tipo, varias unidades de procesamiento diferentes son invocadas por una única unidad de control.

MISD (*Multiple Instruction stream, Single Data stream*): flujo múltiple de instrucciones y único flujo de datos. Significa que varias instrucciones actúan sobre el mismo y único trozo de datos. Este tipo de máquinas se pueden interpretar de dos maneras. Una es considerar la clase de máquinas que requerirían que unidades de procesamiento diferentes recibieran instrucciones distintas operando sobre los mismos datos. Esta clase de arquitectura ha sido clasificada por numerosos arquitectos de computadores como impracticable o imposible, y en estos momentos no existen ejemplos que funcionen siguiendo este modelo. Otra forma de interpretar los MISD es como una clase de máquinas donde un mismo flujo de datos fluye a través de numerosas unidades procesadoras.

MIMD (*Multiple Instruction stream, Multiple Data stream*): flujo de instrucciones múltiple y flujo de datos múltiple. Son máquinas que poseen varias unidades procesadoras en las cuales se pueden realizar múltiples instrucciones sobre datos diferentes de forma simultánea. Las MIMD son las más complejas, pero son también las que potencialmente ofrecen una mayor eficiencia en la ejecución concurrente o paralela.

2. De la anterior describa cual se aplica en: OpenMP, MPI, Multiprocessing y Threads.

R. *MIMD*, se aplica en **OpenMP** y **MPI**, ya que ambos están diseñados para permitir que varios hilos de ejecución independientes se ejecuten simultáneamente en diferentes núcleos de procesador.

MIMD y *MISD*, se aplican en **MPI** y **Multiprocessing**, ambos están diseñados para ejecutar múltiples tareas en paralelo en diferentes procesadores y sistemas, lo que requiere la capacidad de procesar tanto múltiples instrucciones como múltiples datos al mismo tiempo.

https://github.com/Jael-Lany-Valeriano-Baltazar/PRIMER_EXAMEN_PARCIAL_INF317

3. Con MPI y Multiprocessing despliegue verdad o falso si una palabra es palíndromo.

https://github.com/Jael-Lany-Valeriano-Baltazar/PRIMER_EXAMEN_PARCIAL_INF317/tree/main/PRIMER_EXAMEN_INF317/EJERCICIO_3

4. Con Multiprocessing y MPI genere la siguiente serie 0,1,1,2,3,5,8,13,... Al menos unos 1000 elementos de la serie.

https://github.com/Jael-Lany-Valeriano-Baltazar/PRIMER_EXAMEN_PARCIAL_INF317/tree/main/PRIMER_EXAMEN_INF317/EJERCICIO_4

5. Con MPI y OpenMP realice el cálculo de Pi, mediante sumas sucesivas (unos 1000 elementos).

6. Con OpenMP y MPI multiplique una matriz de 100x100.

A

1	2	3	4	5	...	100
101	102					
				...	999	10000

B

1	2	3	4	5	...	100
101	102					
				...	999	10000

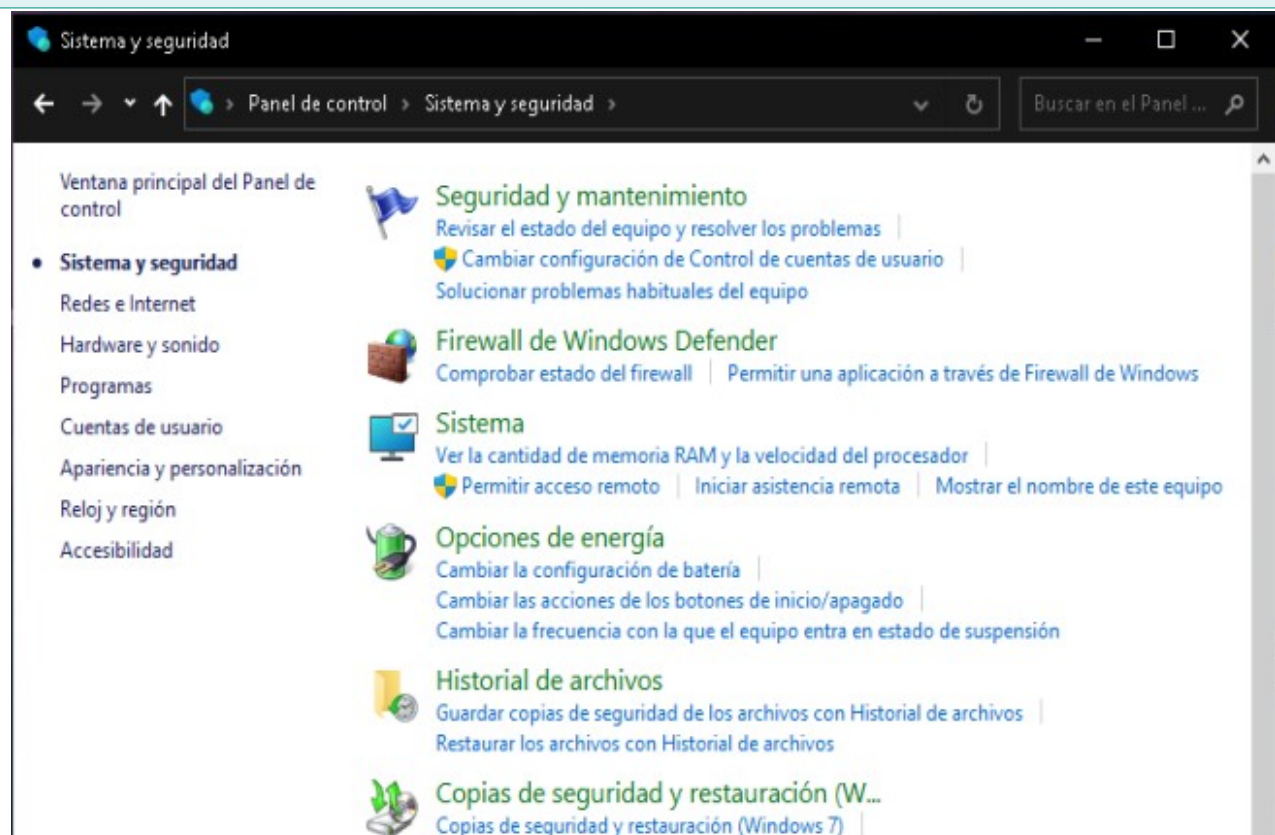
https://github.com/Jael-Lany-Valeriano-Baltazar/PRIMER_EXAMEN_PARCIAL_INF317/tree/main/PRIMER_EXAMEN_INF317/EJERCICIO_6

7. Configure una máquina virtual con Linux debían y configure otra máquina Windows, para que se consulte a ambas mediante ssh (manual de como se hace).

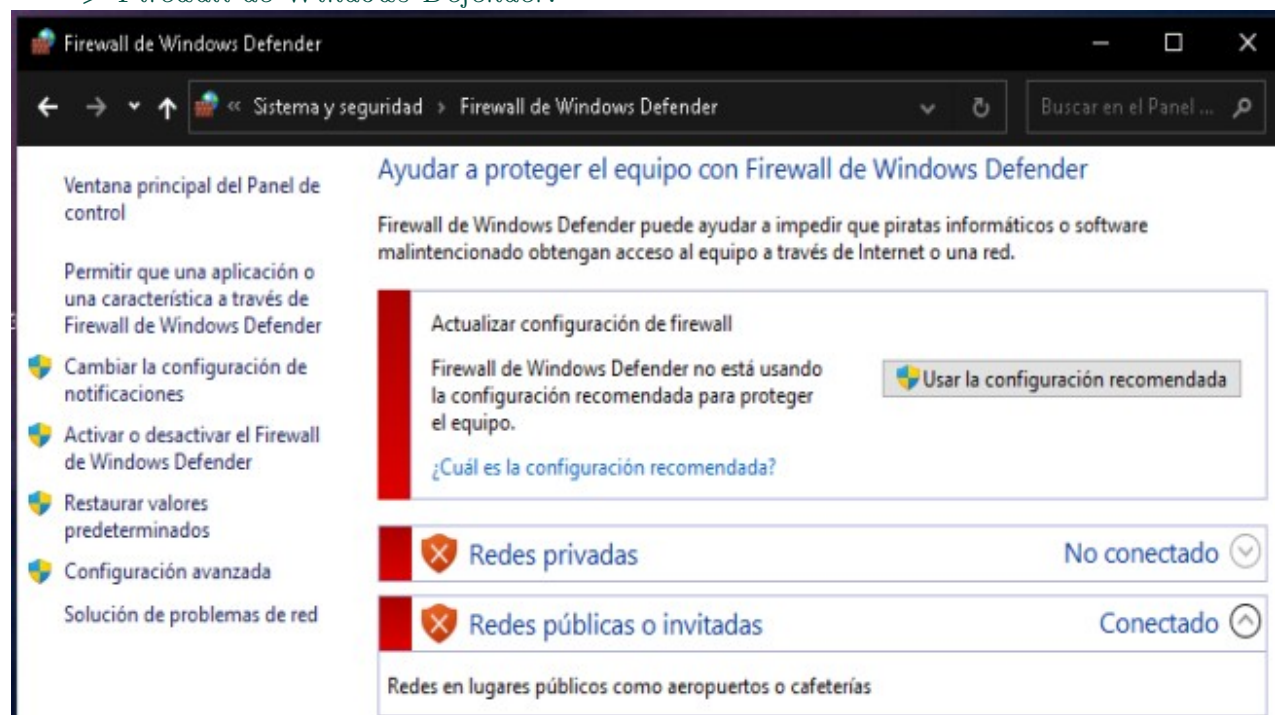
R.

CONFIGURACIÓN DE WINDOWS:

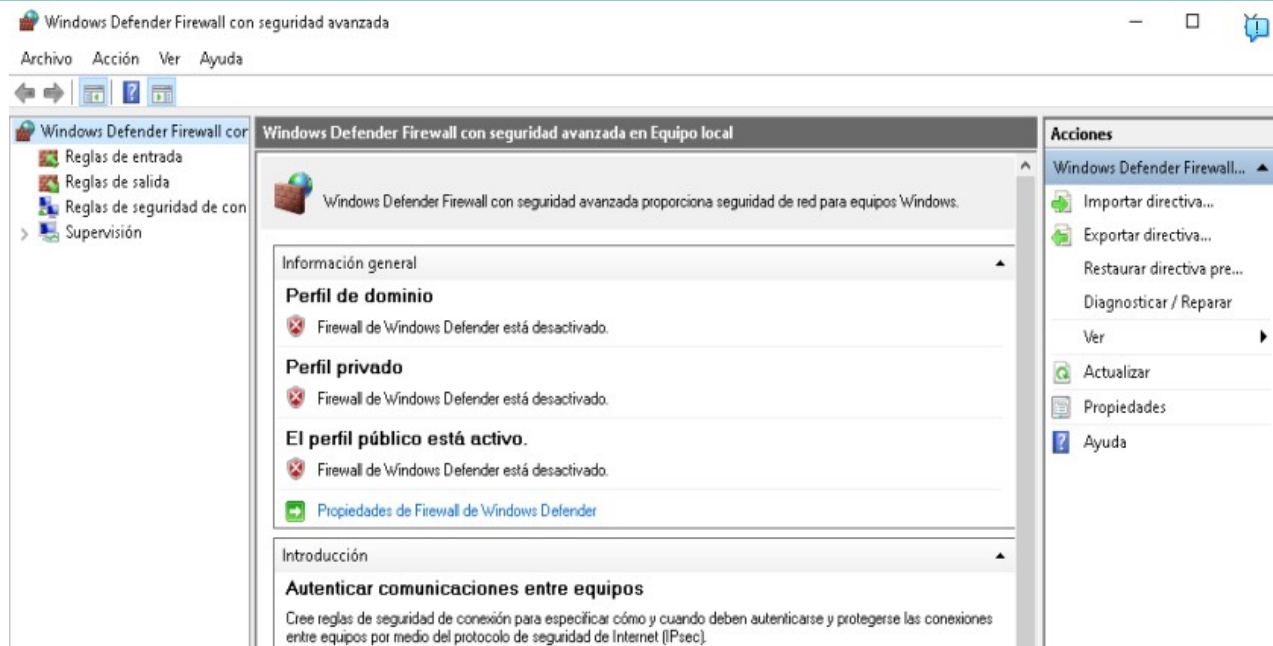
1. Abrir el Panel de control de Windows y seleccionar "Sistema y seguridad".



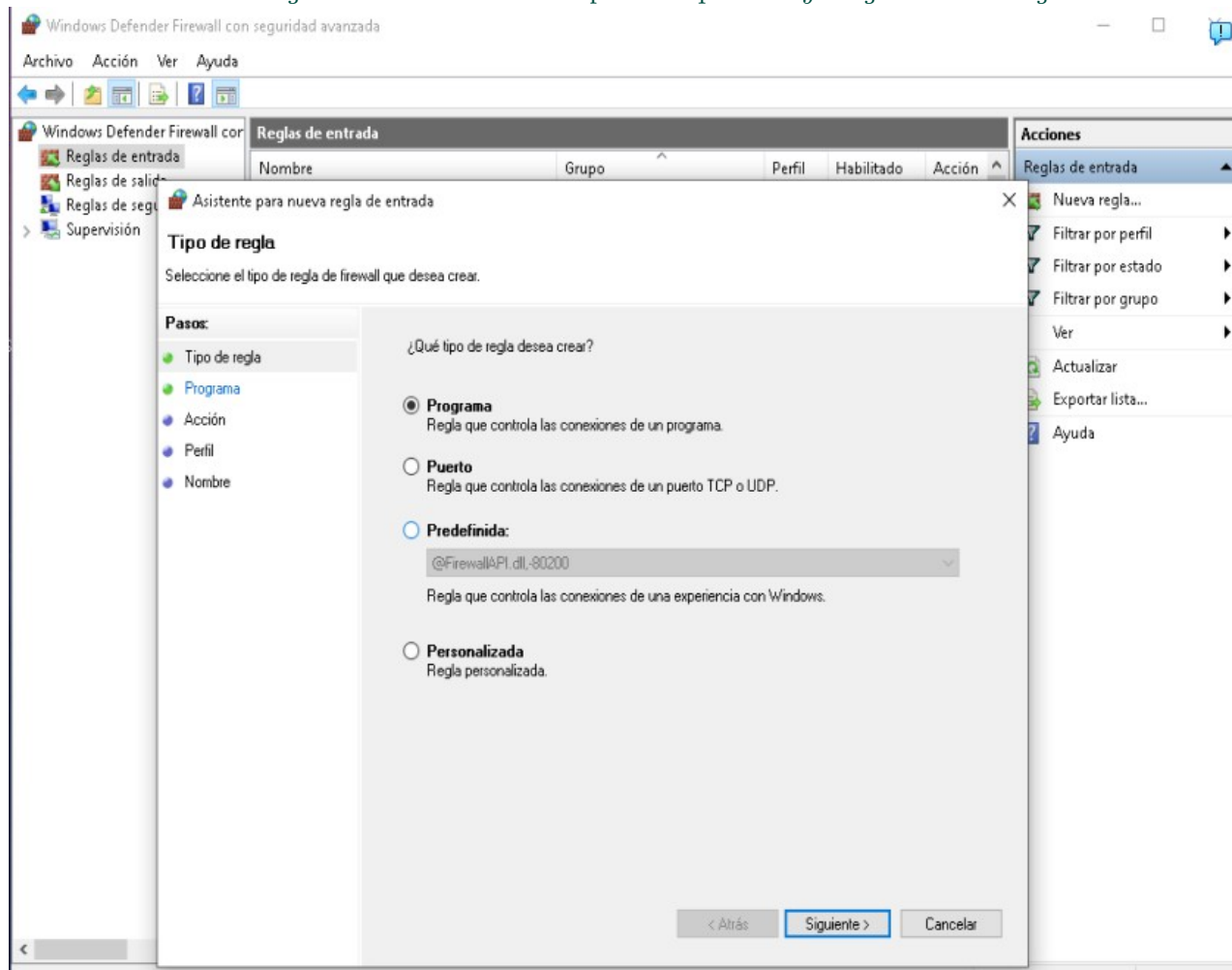
2. Seleccionar "Firewall de Windows" y luego "Configuración avanzada".
> Firewall de Windows Defender:



- > Configuración Avanzada:



3. Seleccionar "Reglas de entrada" en el panel izquierdo y luego "Nueva regla".



4. Seleccionar "Puerto" como tipo de regla y hacer clic en "Siguiente".
5. Seleccionar "TCP" como el tipo de protocolo y especificar el número de puerto utilizado para SSH (el puerto predeterminado es 22). Hacer clic en "Siguiente".

Asistente para nueva regla de entrada

Protocolo y puertos

Especifique los puertos y protocolos a los que se aplica esta regla.

Pasos:

- Tipo de regla
- Protocolo y puertos**
- Acción
- Perfil
- Nombre

¿Se aplica esta regla a TCP o UDP?

☒ TCP

☐ UDP

¿Se aplica esta regla a todos los puertos locales o a unos puertos locales específicos?

☐ Todos los puertos locales

☒ Puertos locales específicos:

Ejemplo: 80, 443, 5000-5010

< Atrás Siguiente > Cancelar

6. Seleccionar "Permitir la conexión" y hacer clic en "Siguiente".

Acción

Especifique la acción que debe llevarse a cabo cuando una conexión coincide con las condiciones especificadas en la regla.

Pasos:

- Tipo de regla
- Protocolo y puertos
- **Acción**
- Perfil
- Nombre

¿Qué medida debe tomarse si una conexión coincide con las condiciones especificadas?

☒ **Permitir la conexión**

Esto incluye las conexiones protegidas mediante IPsec y las que no lo están.

☐ **Permitir la conexión si es segura**

Esto incluye solamente las conexiones autenticadas mediante IPsec. Éstas se protegerán mediante la configuración de reglas y propiedades de IPsec del nodo Regla de seguridad de conexión.

Personalizar...

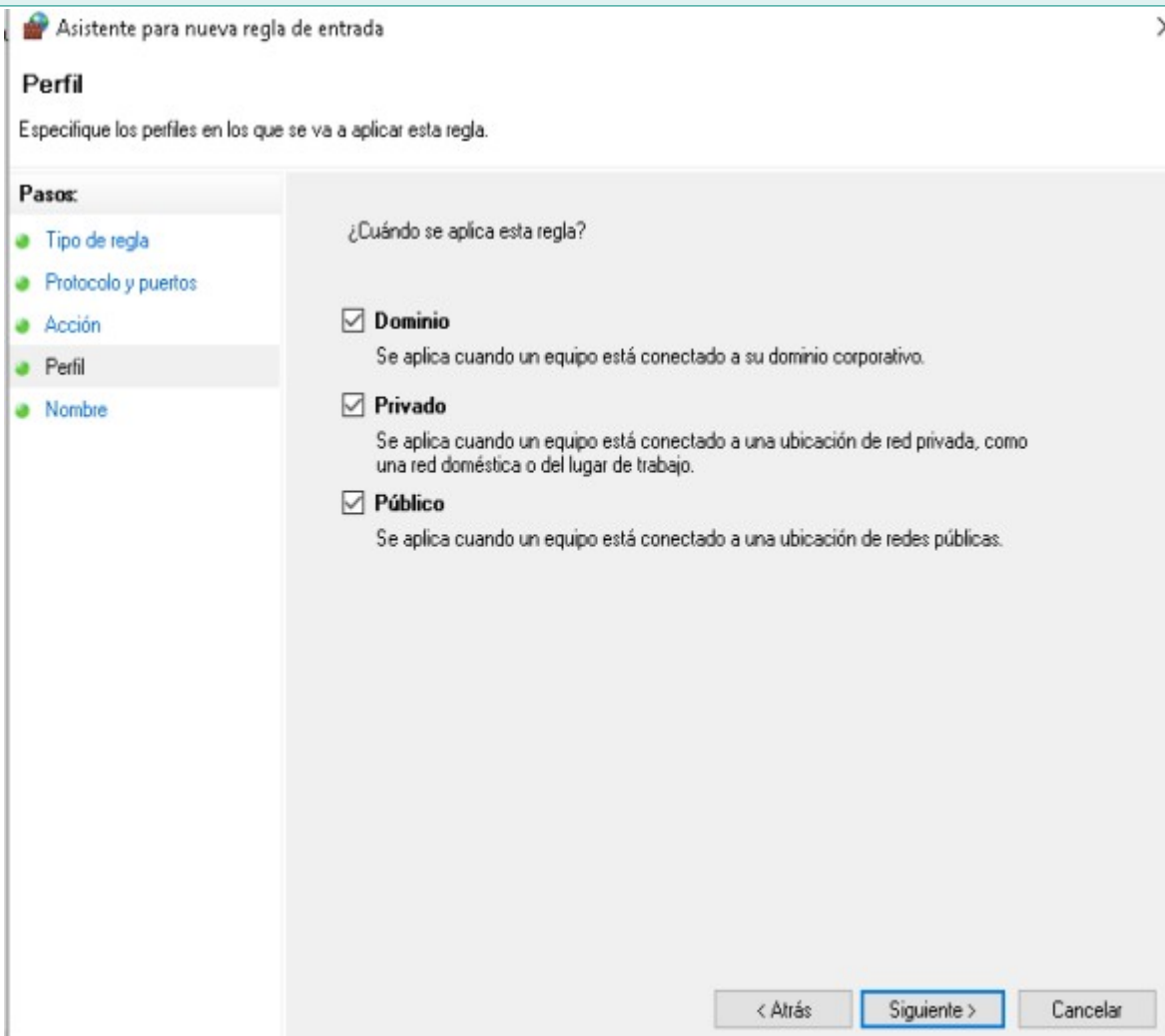
☐ **Bloquear la conexión**

< Atrás

Siguiente >

Cancelar

7. Seleccionar los perfiles de red para los que desea aplicar esta regla (por ejemplo, "Dominio", "Privado" y "Público") y hacer clic en "Siguiente".



8. Asignar un nombre a la regla (por ejemplo, "SSH entrante") y hacer clic en "Finalizar".

Nombre

Especifique el nombre y la descripción de esta regla.

Pasos:

- Tipo de regla
- Protocolo y puertos
- Acción
- Perfil
- Nombre

Nombre:

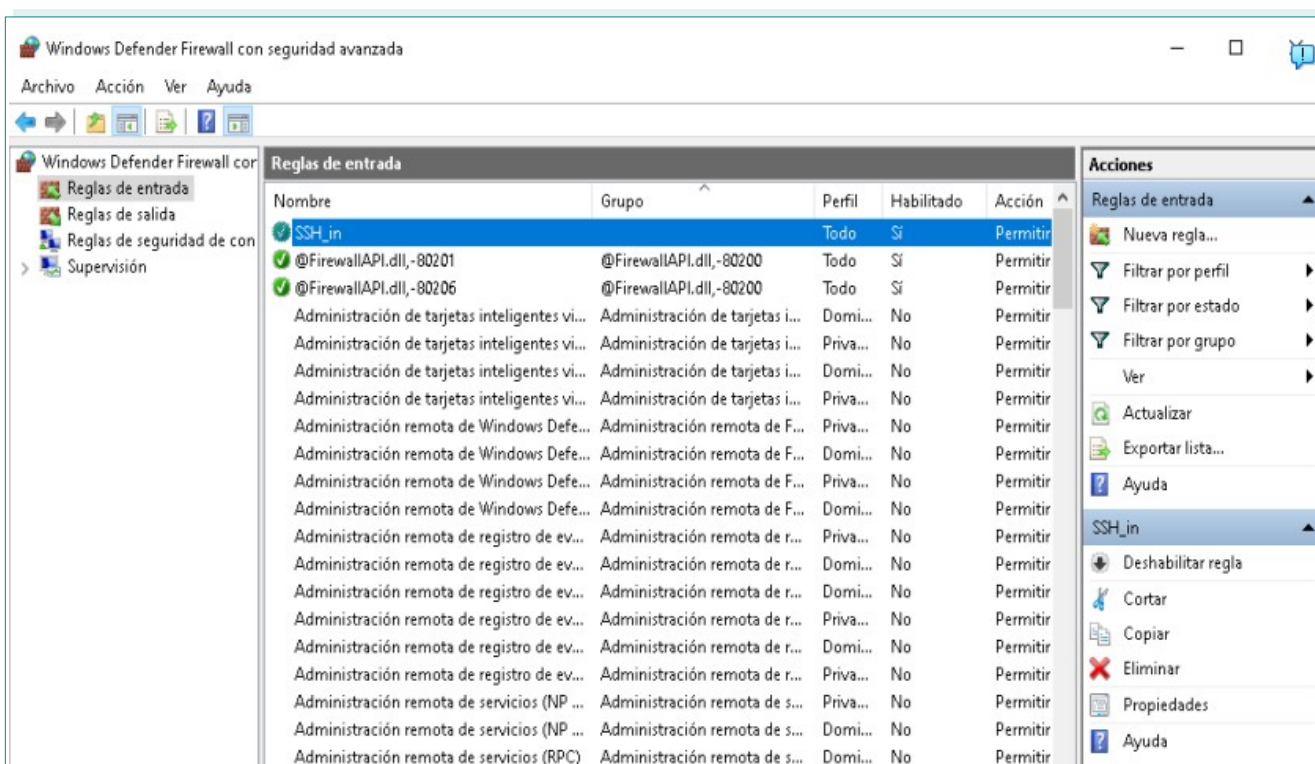
Descripción (opcional):

< Atrás

Finalizar

Cancelar

La regla SSH_in fue agregada correctamente:



INSTALACIÓN DE SSH EN LINUX:

1. Actualizar e instalar con los siguientes comandos:
`sudo apt-get remove --purge openssh-server`
`sudo apt-get install openssh-server`

```

Parrot Terminal
Archivo Editar Ver Buscar Terminal Ayuda
[~] $ sudo apt-get install openssh-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho

```

2. Después de la instalación. Se inicia el SSH:

```

[~] $ sudo systemctl start sshd

```

3. Se habilita el SSH:

```

└─$ sudo systemctl enable sshd
Failed to enable unit: Refusing to operate on alias name or linked unit
file: sshd.service
└─[x]─[lany@parrot]─[~]
└─$ sudo ls -l /etc/systemd/system/sshd.service
lrwxrwxrwx 1 root root 31 abr 26 15:47 /etc/systemd/system/sshd.service
-> /lib/systemd/system/ssh.service
└─[lany@parrot]─[~]
└─$ sudo systemctl enable /lib/systemd/system/ssh.service
└─[lany@parrot]─[~]

```

Si sale el error en letras rojas, se verifica si existe un archivo de unidad vinculado, con el siguiente comando:

```
sudo ls -l /etc/systemd/system/sshd.service
```

Luego, se activa el archivo:

```
sudo systemctl enable <nombre del archivo de unidad vinculado>
```

Previamente identificados los nombres de usuario de las máquinas e IP's, se realiza la siguiente conexión:

→ Para realizar la conexión de Windows a Linux, se ejecuta el siguiente comando:

```

C:\Users\Zia OS>ssh lany@192.168.0.13
ssh: connect to host 192.168.0.13 port 22: Connection

```

→ Para realizar la conexión de Linux a Windows, se ejecuta el siguiente comando:

```

└─[lany@parrot]─[~]
└─$ ssh "zia os"@192.168.0.12
ssh: connect to host 192.168.0.12 port 22: Connection

```