



# **Creación de syslog en un EPC 1522 PLCnext Control**

**Phoenix Contact**

**Diciembre de 2025**



## Contenido

1	Introducción .....	4
1.1	¿Qué es Syslog? .....	4
1.2	Servidor syslog .....	4
1.3	Cliente syslog .....	5
2	Rsyslog .....	6
2.1	¿Por qué Rsyslog? .....	7
2.2	¿Está Rsyslog instalado? .....	7
2.3	Servidor en contenedor Linux.....	8
2.4	Configuración del servidor Rsyslog .....	10
2.5	Activación del contenedor con syslog .....	19
2.6	Consideraciones sobre el Contenedor.....	21
3	Referencias .....	22



## Tabla de Ilustraciones

Ilustración 1.1.- Syslog	4
Ilustración 2.1.- Logo Rsyslog	6
Ilustración 2.2.- Ejemplo de aplicación de Rsyslog/syslog	9
Ilustración 2.3.- Primera parte del archivo de configuración rsyslog.conf	11
Ilustración 2.4.- Segunda parte del archivo de configuración rsyslog.conf	12
Ilustración 2.5.- Tercera parte del archivo de configuración rsyslog.conf	15
Ilustración 2.6.- Ejemplo de mensajes del cliente Syslog Switch 2212	18
Ilustración 2.7.- Carpetas y archivos generados por el servidor Rsyslog	18

## 1 Introducción

### 1.1 ¿Qué es Syslog?

Syslog, abreviatura de *System Logging Protocol*, es un protocolo estándar utilizado para enviar mensajes de registro o eventos del sistema a un servidor centralizado, conocido como servidor Syslog. Fue desarrollado inicialmente para sistemas UNIX en la década de los 80 y formalizado como estándar por la IETF en el RFC 3164 en 2001 [1]. Su propósito principal es recopilar, almacenar y analizar registros provenientes de diversos dispositivos y aplicaciones, como routers, switches, firewalls, servidores y más [1] [2].

### 1.2 Servidor syslog

Un servidor Syslog es una herramienta centralizada que recopila, almacena y gestiona mensajes de registro (logs) enviados por dispositivos y aplicaciones en una red. Estos servidores permiten a los administradores de sistemas y redes analizar eventos, solucionar problemas y cumplir con normativas de seguridad.

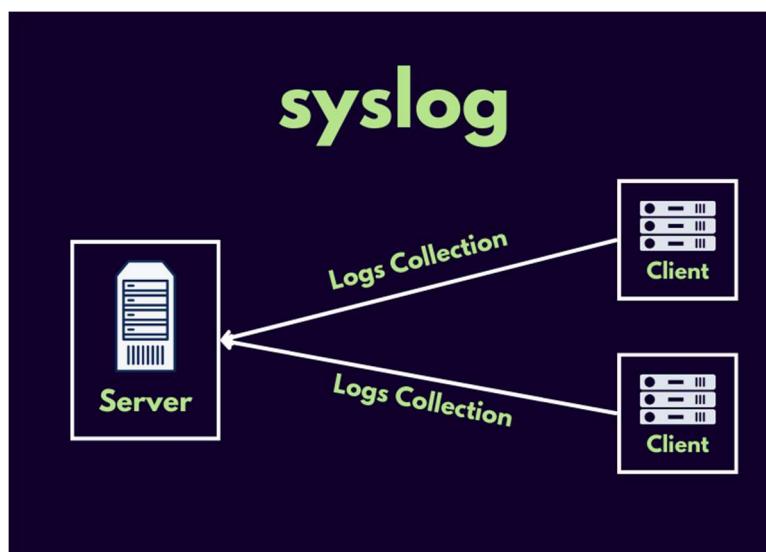


Ilustración 1.1.- Syslog



### 1.3 Cliente syslog

Un cliente Syslog es cualquier dispositivo o aplicación configurada para enviar mensajes de registro a un servidor Syslog. Esto incluye desde sistemas operativos hasta dispositivos especializados.

Los dispositivos de Phoenix Contact, como los PLCs de la familia **PLCnext** y algunos switches gestionados, incluyen soporte nativo para enviar logs mediante Syslog.

## 2 Rsyslog

Rsyslog es una herramienta lanzada en 2004 de código abierto utilizada en sistemas UNIX y similares para la gestión y procesamiento de mensajes de registro (logs). Es una evolución del protocolo syslog tradicional, ofreciendo características avanzadas como filtrado basado en contenido, configuración flexible, transporte seguro mediante TCP, SSL y RELP, y compatibilidad con bases de datos como MySQL, PostgreSQL y MongoDB [3] [4]. Además, Rsyslog es conocido por su alto rendimiento y diseño modular, lo que lo convierte en una solución versátil para entornos empresariales y sistemas pequeños [5].

Se ha convertido en el Daemon syslog por defecto en distribuciones como Ubuntu o RHEL. [6]



Ilustración 2.1.- Logo Rsyslog

## 2.1 ¿Por qué Rsyslog?

- **Alto rendimiento:** Rsyslog está diseñado para manejar grandes volúmenes de datos de manera eficiente [5].
- **Compatibilidad:** Su configuración es una extensión del formato syslog original, lo que facilita la transición desde syslogd [6].
- **Flexibilidad:** Permite personalizar el flujo de mensajes mediante módulos de entrada, reglas y acciones específicas [7].
- **Soporte avanzado:** Incluye características como transporte seguro (TLS), operaciones en cola para manejar salidas fuera de línea y soporte para múltiples protocolos (RFC 5424, RFC 5425) [4].

## 2.2 ¿Está Rsyslog instalado?

En una distribución basada en Linux, para verificar que Rsyslog está instalado existen varias opciones. Algunas de ellas son:

- 1) Comprobar si el archivo de configuración de Rsyslog está instalado, normalmente estará en '`/etc/rsyslog.conf`'.
- 2) Verificar si el paquete Rsyslog ya está instalado:
  - Para sistemas basados en Debian/Ubuntu

`dpkg -l | grep Rsyslog`

- Para versiones basadas en CentOS/RHEL

`rpm -qa | grep Rsyslog`

- Para versiones basadas en Alpine:

`apk info -e Rsyslog`

- 3) Verificar si el archivo binario de Rsyslog existe:

`which rsyslogd`

## 2.3 Servidor en contenedor Linux

Una opción muy versátil es tener instalado el servidor Rsyslog en un contenedor, ya que ofrece múltiples ventajas para la gestión de logs en entornos modernos. Entre estas razones destacan [8]:

- **Portabilidad:** Al estar en un contenedor, el servidor Rsyslog puede ser fácilmente desplegado y trasladado entre diferentes entornos, asegurando consistencia en la configuración y reduciendo problemas de compatibilidad.
- **Escalabilidad:** Los contenedores permiten escalar instancias de Rsyslog según las necesidades de la infraestructura, adaptándose a picos de carga o requisitos específicos.
- **Aislamiento:** Mantener Rsyslog en un contenedor asegura un entorno separado del sistema anfitrión, lo cual mejora la seguridad y facilita la gestión de dependencias.
- **Facilidad de mantenimiento:** Los contenedores simplifican las actualizaciones y el mantenimiento al permitir la creación de imágenes preconfiguradas con versiones específicas de Rsyslog.
- **Integración con Docker:** Rsyslog puede configurarse para recolectar directamente los logs de los contenedores Docker, centralizando toda la información en un único punto para su análisis y monitoreo.

Esta documentación utiliza como base las imágenes de Ubuntu 22.04 y Alpine 3.21, que se utilizaron de manera independiente para probar Rsyslog en distintas distribuciones de Linux. Cabe destacar que en ambas imágenes, Rsyslog no viene preinstalado. Los pasos de instalación y configuración son idénticos, variando únicamente el gestor de paquetes: **apt** para Ubuntu y **apk** para Alpine.

Para la validación de esta documentación, se han probado los contenedores mencionados en el equipo de Phoenix Contact PLCnext Industrial Edge EPC 1502. Como cliente se ha empleado un switch Ethernet industrial [FL SWITCH 2212-2TC-2SFX](#).

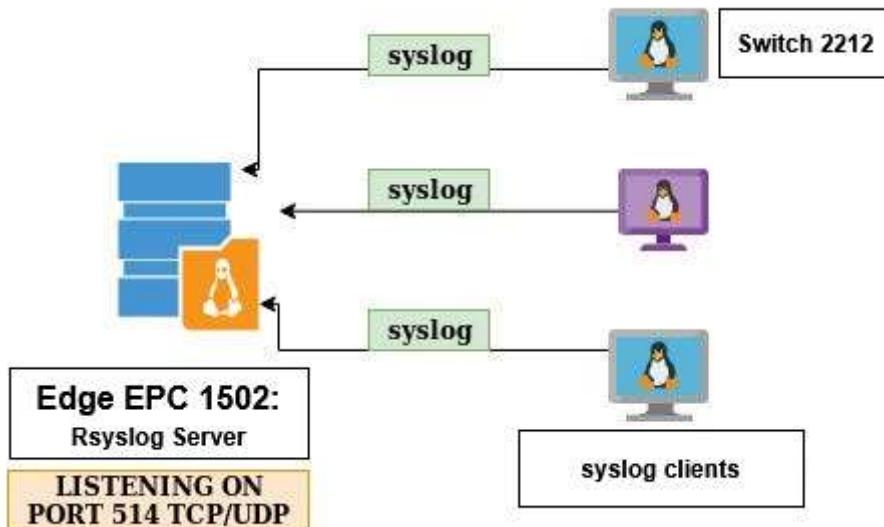


Ilustración 2.2.- Ejemplo de aplicación de Rsyslog/syslog

### i. Instalación de Rsyslog

Primero se actualizan los repositorios y se instala Rsyslog:

```
apt update && apt install Rsyslog
```

Si se ejecuta uno de los comandos de verificación de la instalación:

```
dpkg -l | grep rsyslog # Debian/Ubuntu
```

El resultado debería ser similar a:

```
ii rsyslog 8.2112.0-2ubuntu2.2 amd64 reliable system and kernel logging Daemon
```

Esto confirma que Rsyslog está correctamente instalado.

## 2.4 Configuración del servidor Rsyslog

### Introducción

Para configurar Rsyslog en modo servidor dentro de un contenedor, se recomienda montar el archivo de configuración desde el host. Esta práctica ofrece dos ventajas principales:

1. Permite editar la configuración directamente desde el host sin necesidad de reconstruir la imagen o acceder al contenedor
2. Preserva la configuración incluso si el contenedor es eliminado, facilitando su reutilización

### Obtención de archivo de configuración

Para obtener el archivo de configuración de Rsyslog, se puede extraer el que viene por defecto en la instalación de Rsyslog, en la carpeta '`/etc/rsyslog.conf`'.

### Configuración de Rsyslog como servidor

Dentro del archivo de configuración:

#### Habilitación de protocolos

Para habilitar los protocolos UDP y TCP, son necesarias las siguientes líneas:

`module(load="imudp") # Para cargar el módulo UDP`

`module(load="imtcp") # Para cargar el módulo TCP`

#### Configuración de puertos

Una vez habilitados los protocolos, se configuran los puertos para recibir los logs:

`input(type="imudp" port="514") # Puerto 514 para recibir logs mediante UDP`

`input(type="imtcp" port="514") # Puerto 514 para recibir logs mediante TCP`

#### Control de acceso (Opcional)

Si desea limitar el acceso desde subredes, IPs o dominios específicos, puede utilizar la directiva '`$AllowedSender`' que funciona como lista blanca: Un ejemplo sería:

`$AllowedSender TCP, 127.0.0.1, 192.168.1.0/24, *.example.com`

Esta configuración:

- Aplica solo para conexiones TCP.
- Permite conexiones desde localhost (127.0.0.1).
- 192.168.10.0/24: Permite conexiones desde toda la subred 192.168.1.x (es decir, todas las IPs desde 192.168.1.0 hasta 192.168.1.255)
- \*.example.com: Permite conexiones desde cualquier host en el dominio example.com

```
C:\> Users > rsyslog.conf
1  # /etc/rsyslog.conf configuration file for rsyslog
2  #
3  # For more information install rsyslog-doc and see
4  # /usr/share/doc/rsyslog-doc/html/configuration/index.html
5  #
6  # Default logging rules can be found in /etc/rsyslog.d/50-default.conf
7
8
9 #####
10 ##### MODULES #####
11 #####
12
13 module(load="imuxsock") # provides support for local system logging
14 #module(load="immark") # provides --MARK-- message capability
15
16 # provides UDP syslog reception
17 module(load="imudp")
18 input(type="imudp" port="514")
19
20 # provides TCP syslog reception
21 module(load="imtcp")
22 input(type="imtcp" port="514")
23
24 # $AllowedSender
25 $AllowedSender TCP, 127.0.0.1, 192.168.1.0/24, *.example.com
```

Ilustración 2.3.- Primera parte del archivo de configuración rsyslog.conf



## Gestión de Logs del Kernel

Al usar Rsyslog en contenedores, podría aparecer el siguiente error:

```
rsyslogd: imklog: cannot open kernel log (/proc/kmsg): Operation not permitted.  
rsyslogd: activation of module imklog failed [v8.2112.0 try  
https://www.rsyslog.com/e/2145]
```

Esto ocurre porque Rsyslog intenta leer los logs del kernel, pero Docker no lo permite por razones de seguridad. Hay dos soluciones:

### Opción 1: Ejecutar con privilegios

Si necesita acceder a los mensajes del kernel, ejecute el contenedor con el flag ‘**--privileged**’:

```
docker run --privileged [otras_opciones] imagen_rsyslog
```

Los mensajes del kernel se enviarán a la dirección localhost (127.0.0.1), algunos ejemplos de estos mensajes escritos en el archivo de ‘syslog.log’ que corresponde a los mensajes locales son:

```
Apr 8 10:47:15 4df736643b79 kernel: wls7: 80 MHz not supported, disabling VHT  
Apr 8 10:47:15 4df736643b79 kernel: wls7: send auth to 2a:5b:0e:e7:94:75 (try 1/3)  
Apr 8 10:47:15 4df736643b79 kernel: wls7: authenticated  
Apr 8 10:47:15 4df736643b79 kernel: wls7: RX ReassocResp from  
2a:5b:0e:e7:94:75 (capab=0x1421 status=0 aid=1)  
Apr 8 10:47:15 4df736643b79 kernel: wls7: associated  
Apr 8 10:47:15 4df736643b79 kernel: ath: EEPROM regdomain: 0x833a
```

### Opción 2: Deshabilitar la captura de logs del kernel

Si no necesita los mensajes del kernel, comentar o eliminar la siguiente línea:

```
module(load="imklog" permitnonkernelfacility="on")
```

```
24 # provides kernel logging support and enable non-kernel klog messages  
25 module(load="imklog" permitnonkernelfacility="on")
```

Ilustración 2.4.- Segunda parte del archivo de configuración rsyslog.conf



## Plantillas para almacenamiento de Logs

Se pueden utilizar dos tipos de sintaxis para crear plantillas que definan cómo almacenar los mensajes syslog:

Sintaxis 'legacy' o 'tradicional':

```
$template remote-incoming-logs, "/var/log/%HOSTNAME%/%PROGRAMNAME%.log"  
.* ?remote-incoming-logs  
& ~
```

Sintaxis moderna, basada en RainerScript:

- Para logs de equipos remotos:

```
template(name="RemoteLogTemplate" type="string"  
string="/var/log/remote/%HOSTNAME%/syslog.log")  
if $fromhost-ip != '127.0.0.1' then {  
    action(type="omfile" dynaFile="RemoteLogTemplate")  
    stop # Evita que el mensaje sea procesado por reglas posteriores  
}
```

Esta configuración guarda los mensajes syslog en un archivo llamado 'syslog.log' dentro de una carpeta cuyo nombre será la dirección IP del cliente syslog.

- Para logs del propio servidor:

```
template(name="LocalLogTemplate" type="string"  
string="/var/log/local/syslog.log")  
if $fromhost-ip == '127.0.0.1' then {  
    action(type="omfile" dynaFile="LocalLogTemplate")  
    stop # stop evita que el mensaje sea procesado por reglas posteriores  
}
```

- Para todos los logs (locales y remotos):

```

template(name="AllLogTemplate" type="string"
string="/var/log/%HOSTNAME%/%PROGRAMNAME%.log")

action(type="omfile" dynaFile="AllLogTemplate")

```

```

63 ######
64 ### TEMPLATES
65 #####
66
67 template(name="RemoteLogTemplate" type="string"
68 |           string="/var/log/remote/%HOSTNAME%/syslog.log")
69
70 template(name="LocalLogTemplate" type="string"
71 |           string="/var/log/local/syslog.log")
72
73
74 if $fromhost-ip != '127.0.0.1' then {
75     action(type="omfile" dynaFile="RemoteLogTemplate"
76           dirCreateMode="0755"
77           fileCreateMode="0644")
78     stop
79 }
80
81
82 if $fromhost-ip == '127.0.0.1' then {
83     action(type="omfile" dynaFile="LocalLogTemplate"
84           dirCreateMode="0755"
85           fileCreateMode="0644")
86     stop
87 }
88
89
90 #action(type="omfile" dynaFile="RemoteLogTemplate"
91 #           dirCreateMode="0755"
92 #           fileCreateMode="0644")
93

```

Ilustración 2.5.- Tercera parte del archivo de configuración rsyslog.conf

## Configuración de Permisos

Al usar la sintaxis moderna (basada en RainerScript), las directivas de permisos como '**\$FileCreateMode 0640**' o '**\$DirCreateMode 0755**' dejan de tener efecto, deben especificarse dentro de la acción:

```

action(type="omfile" dynaFile="RemoteLogTemplate"
dirCreateMode="0755"
fileCreateMode="0644")

```



La directiva '**\$Umask 0022**' limita los permisos máximos. Por ejemplo, aunque se especifique 777 para creación de directorios, el máximo efectivo será 755. Si se establece '**\$Umask 0000**', se permitirá crear carpetas con permisos 777.

### **Reinicio tras la configuración**

Tras la modificación de la configuración, para que surtan efecto los cambios, se debe reiniciar Rsyslog, para hacerlo se debe tener en cuenta lo siguiente.

#### **Control del Servicio Rsyslog**

Como los contenedores generalmente no ejecutan un sistema init completo (systemd, Upstart, SysVinit), no se puede usar, por ejemplo, systemctl restart rsyslog o service rsyslog restart.

#### **Iniciar el servicio**

Para iniciar Rsyslog se debe usar: '**rsyslogd**'

#### **Parar el servicio**

Para parar Rsyslog se debe usar: '**pkill rsyslogd**'

#### **Reiniciar el servicio**

Para reiniciar Rsyslog se debe usar: '**pkill rsyslogd && rsyslogd**'

#### **Verificación del Funcionamiento**

##### **Verificar que el servicio Rsyslog está en ejecución:**

***ps aux | grep Rsyslog***

Resultado esperado (debe aparecer al menos dos líneas, una del proceso grep y otra del propio Rsyslog):

```
syslog 24 0.6 0.0 308556 5364 ? Ssl 11:11 0:00 rsyslogd
root 34 0.0 0.0 3472 1684 pts/0 S+ 11:12 0:00 grep --color=auto Rsyslog
```

O por ejemplo en Alpine:

```
1 root 0:00 rsyslogd -n
28 root 0:00 grep Rsyslog
```

##### **Si aparece un error sobre el usuario 'syslog':**

***ID for user 'syslog' could not be found or error***



Esto ocurre porque Rsyslog por defecto intenta ejecutarse como el usuario 'syslog' por seguridad, pero este usuario no está creado automáticamente en Alpine. Para solucionarlo:

```
addgroup -S syslog && adduser -S -G syslog syslog
```

### Verificar que Rsyslog está escuchando en los puertos configurados:

```
ss -tunelp | grep 514
```

Dará algo similar a:

```
udp UNCONN 0 0 0.0.0.0:514 0.0.0.0:* users:(("rsyslogd",pid=1,fd=5))
ino:18517 sk:1001 cgroup:/ <->

udp UNCONN 0 0 [:]:514 [:]* users:(("rsyslogd",pid=1,fd=6))
ino:18518 sk:1002 cgroup:/ v6only:1 <->

tcp LISTEN 0 25 0.0.0.0:514 0.0.0.0:* users:(("rsyslogd",pid=1,fd=7))
ino:18521 sk:1003 cgroup:/ <->

tcp LISTEN 0 25 [:]:514 [:]* users:(("rsyslogd",pid=1,fd=8))
ino:18522 sk:1004 cgroup:/ v6only:1 <->
```

Nota: Para utilizar el comando ‘**ss**’, debe instalar el paquete ‘**iproute2**’ en el contenedor.

/home/ima_user/compartido_iec62443/syslog/var/log/					
Nombre	Tamaño	Modificado	Permisos	Propietario	
..		23/03/2025 17:29:12	rwxrwxr-x	ima_user	
local		10/04/2025 12:12:20	rwxrwxr-x	ima_user	
remote		08/04/2025 12:54:34	rwxrwxrwx	dhcpcd	
/home/ima_user/compartido_iec62443/syslog/var/log/local/					
Nombre	Tamaño	Modificado	Permisos	Propietario	
..		24/03/2025 12:46:50	rwxrwxr-x	ima_user	
syslog.log	2 KB	14/04/2025 8:21:21	rw-r--r--	root	
/home/ima_user/compartido_iec62443/syslog/var/log/remote/					
Nombre	Tamaño	Modificado	Permisos	Propietario	
..		24/03/2025 12:46:50	rwxrwxr-x	ima_user	
192.168.1.110		08/04/2025 12:54:34	rwxr-xr-x	root	
/home/ima_user/compartido_iec62443/syslog/var/log/remote/192.168.1.110/					
Nombre	Tamaño	Modificado	Permisos	Propietario	
..		08/04/2025 12:54:34	rwxrwxrwx	dhcpcd	
syslog.log	6 KB	14/04/2025 8:57:46	rw-r--r--	root	

Ilustración 2.6.- Carpetas y archivos generados por el servidor Rsyslog

```

7 Apr 8 12:12:49 192.168.1.110 Link down on port 14.
8 Apr 8 12:12:52 192.168.1.110 Link up on port 14.
9 Apr 8 12:13:08 192.168.1.110 Link down on port 14.
10 Apr 8 12:13:23 192.168.1.110 Link down on port 1.
11 Apr 8 12:13:27 192.168.1.110 Link down on port 2.
12 Apr 8 12:13:27 192.168.1.110 LLDP neighbour lost on Port 2.
13 Apr 9 09:25:16 192.168.1.110 Successful user login.
14 Apr 10 10:12:39 192.168.1.110 Configuration saved successfully.
15 Apr 10 10:42:49 192.168.1.110 Automatic user logout via Web-based management.
16 Apr 10 10:42:57 192.168.1.110 Successful user login.
17 Apr 10 10:43:18 192.168.1.110 Ldap Server IP address changed
18 Apr 10 10:43:18 192.168.1.110 Parameter has been changed by the user.
19 Apr 10 10:43:18 192.168.1.110 Changed device configuration.
20 Apr 10 10:43:11 192.168.1.110 Configuration saved successfully.
21 Apr 10 10:43:31 192.168.1.110 Ldap Server IP address changed
22 Apr 10 10:43:31 192.168.1.110 Parameter has been changed by the user.
23 Apr 10 10:43:31 192.168.1.110 Changed device configuration.
24 Apr 10 10:43:32 192.168.1.110 Configuration saved successfully.
25 Apr 10 10:43:36 192.168.1.110 message repeated 2 times: [ Configuration saved successfully.]
26 Apr 10 10:49:14 192.168.1.110 Link down on port 15.
27 Apr 10 10:49:32 192.168.1.110 Link down on port 14.
```

Ilustración 2.7.- Ejemplo de mensajes del cliente Syslog Switch 2212

## 2.5 Activación del contenedor con syslog

Para lanzar el contenedor con el servicio Rsyslog sería, por ejemplo:

```
docker run --name c_syslog --restart=always -d  
-v /home/ima_user/compartido_iec62443/syslog/var/log/local:/var/log/local  
-v /home/ima_user/compartido_iec62443/syslog/var/log/remote:/var/log/remote  
-v  
/home/ima_user/compartido_iec62443/syslog/var/spool/rsyslog:/var/spool/Rsysl  
og  
-v  
/home/ima_user/compartido_iec62443/syslog/etc/rsyslog.conf:/etc/rsyslog.conf  
-p 514:514/udp -p 514:514/tcp  
pxc_syslog:1.0 rsyslogd -n
```

- **--name c\_syslog:** Asigna el nombre 'c\_syslog' al contenedor para facilitar su gestión.
- **--restart=always:** Configura el contenedor para reiniciarse automáticamente si se detiene por cualquier razón.
- **-d:** Ejecuta el contenedor en modo "detached" (segundo plano).

### Volúmenes:

- **-v /home/ima\_user/compartido\_iec62443/syslog/var/log/local:/var/log/local:** Mapea la carpeta de logs locales del host al contenedor, garantizando la persistencia de los logs.
- **-v /home/ima\_user/compartido\_iec62443/syslog/var/log/remote:/var/log/remote:** Mapea la carpeta de logs remotos para mantener los registros de clientes remotos.
- **-v /home/ima\_user/compartido\_iec62443/syslog/var/spool/rsyslog:/var/spool/rsy  
slog:** Mapea el directorio de almacenamiento temporal donde Rsyslog guarda los mensajes pendientes de procesamiento.
- **-v /home/ima\_user/compartido\_iec62443/syslog/etc/rsyslog.conf:/etc/rsyslog.co**



***nf***: Mapea el archivo de configuración desde el host, permitiendo modificar la configuración sin reconstruir la imagen.

## Puertos

- **-p 514:514/udp -p 514:514/tcp:** Expone el puerto 514 (estándar para syslog) tanto para conexiones UDP como TCP, mapeándolos al mismo puerto dentro del contenedor.
- **pxc\_syslog:1.0:** Especifica la imagen Docker a utilizar con su etiqueta.
- **rsyslogd -n:** Comando que se ejecutará dentro del contenedor, iniciando el demonio rsyslog en modo "no-fork" para que se ejecute en primer plano.

## 2.6 Consideraciones sobre el Contenedor

El contenedor no funcionará correctamente si se ejecuta en entornos Docker para Windows debido a que las direcciones IP de los clientes syslog no aparecerán; serán reemplazadas por la dirección IP del host Docker.

Esta limitación se debe a que Docker en Windows utiliza una capa adicional de virtualización, haciendo que los contenedores se conecten a la red del sistema virtualizado y no directamente al host Windows.

Por esta misma razón, las opciones de red avanzadas de Docker como los drivers host, ipvlan y macvlan tampoco funcionan correctamente en Windows.

### 3 Referencias

- [1] “syslog: Funcionamiento y Configuración - CCNA desde Cero.” Accessed: Apr. 08, 2025. [Online]. Available: <https://ccnadesdecero.es/syslog-funcionamiento-y-configuracion/>
- [2] “¿Qué es Syslog?” Accessed: Apr. 08, 2025. [Online]. Available: <https://pandorafms.com/es/it-topics/que-es-syslog-una-introduccion-al-protocolo-de-registro-del-sistema/>
- [3] “Rsyslog - Debian Wiki.” Accessed: Apr. 08, 2025. [Online]. Available: <https://wiki.debian.org/Rsyslog>
- [4] “Rsyslog - Wikipedia.” Accessed: Apr. 08, 2025. [Online]. Available: <https://en.wikipedia.org/wiki/Rsyslog>
- [5] “Resúmenes - Documentación Rsyslog.” Accessed: Apr. 08, 2025. [Online]. Available: [https://www.rsyslog.com/doc/proposals/big\\_restructuring/book/overview.html](https://www.rsyslog.com/doc/proposals/big_restructuring/book/overview.html)
- [6] “Understanding Differences in Syslog Daemons - Trunc Help.” Accessed: Apr. 08, 2025. [Online]. Available: <https://help.trunc.org/knowledge-base/understanding-differences-in-syslog-daemons/>
- [7] “Basic Structure — Rsyslog documentation.” Accessed: Apr. 08, 2025. [Online]. Available: [https://www.rsyslog.com/doc/configuration/basic\\_structure.html](https://www.rsyslog.com/doc/configuration/basic_structure.html)
- [8] “6 ventajas de usar contenedores Docker en aplicaciones.” Accessed: Apr. 08, 2025. [Online]. Available: <https://keepcoding.io/blog/ventajas-de-usar-contenedores-docker/>