

Instalacion y puesta en marcha de FORT

1.7.0.experimental

Ubuntu Server 26.04 - Guia verificada paso a paso

Compilacion desde codigo fuente, validacion RPKI, servicio systemd y acceso RTR desde la red local

Autor: Nicolas Antoniello (Git: 65007)

Alcance. Esta guia consolida los pasos que fueron probados y funcionaron en Ubuntu Server 26.04 para instalar FORT 1.7.0.experimental. El paquete .deb oficial no se utilizo porque declara dependencias con nombres antiguos que Ubuntu 26.04 ya no ofrece.

1. Entorno validado

Componente	Valor validado
Sistema operativo	Ubuntu Server 26.04 (resolute)
Arquitectura	amd64 / x86_64
FORT	1.7.0.experimental
Instalacion	Compilacion desde tarball oficial
Directorio de configuracion	/etc/fort
Repositorio RPKI local	/var/lib/fort/repository
Salida de VRP/ROA	/var/lib/fort/output/validated-roas.csv
Servicio	fort.service
Puerto RTR usado	TCP 8323
Version maxima RTR	1
Zona horaria recomendada	UTC

2. Por que se compila desde codigo fuente

El paquete Debian de FORT 1.7.0.experimental fue reconocido correctamente, pero apt no pudo instalarlo porque solicita paquetes con nombres anteriores:

```
fort : Depends: libmicrohttpd12
      Depends: libxml2
```

En Ubuntu 26.04 los paquetes equivalentes son:

```
libmicrohttpd12t64
libxml2-16
```

La solucion usada y validada fue compilar FORT contra las bibliotecas nativas de Ubuntu 26.04. No se recomienda forzar dependencias ni mezclar paquetes de otras versiones de Ubuntu o Debian.

3. Instalar dependencias de compilacion

```
sudo add-apt-repository universe
sudo apt update

sudo apt install -y \
    build-essential \
    pkg-config \
    rsync \
    libjansson-dev \
    libssl-dev \
    libcurl4-openssl-dev \
    libxml2-dev \
    libmicrohttpd-dev \
    ca-certificates \
    wget \
    tar
```

Verificar que Ubuntu resolvió las bibliotecas esperadas:

```
apt-cache policy \
    libxml2-dev \
    libxml2-16 \
    libmicrohttpd-dev \
    libmicrohttpd12t64
```

4. Descargar, compilar e instalar FORT 1.7.0.experimental

```
cd /tmp

wget https://github.com/NICMx/FORT-validator/releases/download/1.7.0.experimental/
fort-1.7.0.experimental.tar.gz

tar -xzf fort-1.7.0.experimental.tar.gz
cd fort-1.7.0.experimental

./configure
make -j"$(nproc)"
sudo make install
hash -r
```

Comprobar la instalacion:

```
command -v fort
fort --version
```

La ruta esperada para una instalacion mediante make install es:

```
/usr/local/bin/fort
```

5. Verificar bibliotecas enlazadas

```
ldd "$(command -v fort)" | grep -E 'xml2|microhttpd|ssl|crypto|curl|jansson'
```

En la instalacion validada aparecieron, entre otras:

```
libjansson.so.4
libcurl.so.4
libxml2.so.16
libmicrohttpd.so.12
libcrypto.so.3
libssl.so.3
```

No debe aparecer ninguna biblioteca como not found.

6. Crear usuario y directorios

```
getent passwd fort || sudo useradd \
  --system \
  --home-dir /var/lib/fort \
  --create-home \
  --shell /usr/sbin/nologin \
  fort

sudo install -d -o root -g fort -m 0750 /etc/fort
sudo install -d -o fort -g fort -m 0750 /etc/fort/tal
sudo install -d -o fort -g fort -m 0750 /var/lib/fort/repository
sudo install -d -o fort -g fort -m 0750 /var/lib/fort/output
```

7. Inicializar los TAL

```
sudo -u fort /usr/local/bin/fort \
  --init-tals \
  --tal /etc/fort/tal

sudo find /etc/fort/tal \
  -maxdepth 1 \
  -type f \
  -name '*.tal' \
  -print
```

8. Primera validacion manual

```
sudo -u fort /usr/local/bin/fort \
  --mode standalone \
  --tal /etc/fort/tal \
  --local-repository /var/lib/fort/repository \
  --output.roa /var/lib/fort/output/validated-roas.csv
```

La primera ejecucion puede consumir los dos vCPU disponibles y descargar varios GB. Esto es normal. En la prueba realizada, el repositorio alcanzo aproximadamente 2.7 GB durante la primera sincronizacion.

Monitoreo recomendado desde otra sesion:

```
pgrep -a fort
ps -o pid,stat,etime,%cpu,%mem,cmd -p "$(pgrep -n fort)"
sudo ss -tpn | grep fort
sudo du -sh /var/lib/fort/repository /var/lib/fort/output
vmstat 1
```

Un uso de CPU superior a 100% indica que el proceso utiliza mas de un nucleo. Con 2 vCPU, el maximo teorico es aproximadamente 200%.

9. Verificar el resultado de la validacion

```
echo $?
sudo ls -lh /var/lib/fort/output/validated-roas.csv
sudo head /var/lib/fort/output/validated-roas.csv
sudo wc -l /var/lib/fort/output/validated-roas.csv
sudo stat /var/lib/fort/output/validated-roas.csv
```

Resultado observado en la validacion realizada:

```
Exit status: 0
File size: approximately 24 MB
Total lines: 887046
Header: ASN,Prefix,Max prefix length
```

10. Crear la configuracion permanente

Crear el archivo:

```
sudo nano /etc/fort/config.json
```

Contenido final validado:

```
{
  "mode": "server",
  "tal": "/etc/fort/tal",
  "local-repository": "/var/lib/fort/repository",

  "output": {
    "roa": "/var/lib/fort/output/validated-roas.csv"
  },

  "server": {
    "address": [
      "127.0.0.1",
      "::1",
      "10.0.1.15"
    ],
    "port": "8323",
    "max-rtr-version": 1,
    "interval": {
      "validation": 3600,
      "refresh": 3600,
      "retry": 600,
      "expire": 7200
    }
  },

  "log": {
    "output": "console",
    "level": "info"
  }
}
```

Reemplazar 10.0.1.15 si la VM utiliza otra direccion privada.

```
sudo chown root:fort /etc/fort/config.json
sudo chmod 0640 /etc/fort/config.json
```

Validar sintaxis y permisos:

```
sudo python3 -m json.tool /etc/fort/config.json > /dev/null && \
  echo "JSON syntax is valid."

sudo -u fort test -r /etc/fort/config.json && \
  echo "FORT user can read the configuration."
```

11. Probar la configuracion fuera de systemd

```
sudo -u fort /usr/local/bin/fort \
  --configuration-file /etc/fort/config.json
```

La salida correcta debe confirmar, entre otros valores:

FORT 1.7.0.experimental en Ubuntu Server 26.04 - Guia verificada

```
mode: server
server.address:
  127.0.0.1
  ::1
  10.0.1.15
server.port: 8323
server.max-rtr-version: 1
output.roa: /var/lib/fort/output/validated-roas.csv
```

Y debe mostrar que los sockets se abrieron correctamente. Detener esta prueba con Ctrl+C antes de iniciar el servicio systemd.

12. Crear el servicio systemd

Crear el archivo de unidad:

```
sudo nano /etc/systemd/system/fort.service
```

Contenido del archivo /etc/systemd/system/fort.service:

```
[Unit]
Description=FORT RPKI Validator
Documentation=https://nicmx.github.io/FORT-validator/
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
User=fort
Group=fort
Environment="MALLOC_ARENA_MAX=2"
ExecStart=/usr/local/bin/fort --configuration-file /etc/fort/config.json
Restart=on-failure
RestartSec=10
NoNewPrivileges=true
PrivateTmp=true
ProtectHome=true

[Install]
WantedBy=multi-user.target
```

Guardar el archivo y ejecutar los siguientes comandos por separado:

```
sudo systemctl daemon-reload
sudo systemctl enable --now fort
sudo systemctl status fort
sudo journalctl -u fort -n 100 --no-pager
```

13. Verificar RTR y acceso desde la red local

```
sudo ss -ltnp | grep ':8323'
```

Se esperan listeners en localhost y en la direccion privada de la VM:

```
127.0.0.1:8323
[::1]:8323
10.0.1.15:8323
```

Desde otro equipo o router de la misma red:

```
nc -vz 10.0.1.15 8323
```

Cuando el router se conecte:

```
sudo ss -tnp | grep ':8323'
```

Parametros a configurar en el router:

```
RTR server: 10.0.1.15
RTR port: 8323
RTR version: 1 or automatic negotiation
```

14. Firewall

En la instalacion validada UFW estaba deshabilitado. Aun asi, pueden existir filtros en Proxmox, el nodo, el datacenter, el router o entre VLAN/subredes.

```
sudo ufw status
sudo nft list ruleset
sudo iptables -S
```

Si se habilita UFW, permitir preferentemente solo la IP del router:

```
sudo ufw allow from ROUTER_IP_ADDRESS to 10.0.1.15 port 8323 proto tcp
```

Para acceso desde otra red, debe existir enrutamiento de ida y vuelta. Si el acceso cruza Internet o NAT, se recomienda una VPN en lugar de publicar RTR directamente.

15. Hora y sincronizacion

RPKI depende de fechas de certificados, manifests y CRLs. Mantener la VM sincronizada es obligatorio.

```
timedatectl status
timedatectl timesync-status
```

Valores esperados:

```
System clock synchronized: yes
NTP service: active
```

Para un servidor de infraestructura se recomienda mantener UTC:

```
sudo timedatectl set-timezone UTC
```

Para consultar la hora local de Uruguay sin cambiar la zona del servidor:

```
TZ=America/Montevideo date
```

16. Recursos y comportamiento esperado

- La primera sincronizacion es la mas costosa; las siguientes reutilizan el repositorio local.
- Con 2 vCPU, FORT puede acercarse a 200% de CPU durante etapas intensivas.
- 4 GB de RAM funcionaron para la prueba realizada; se observaron pocos MB de swap, sin presion sostenida.
- Un periodo breve con CPU e I/O en cero antes de finalizar puede ser normal.
- No interrumpir la primera validacion mientras exista actividad de CPU, red o crecimiento del repositorio.

17. Problemas encontrados y solucion

El paquete .deb no instala

```
Unsatisfied dependencies:
fort : Depends: libmicrohttpd12
        Depends: libxml2
```

Solucion: compilar desde el tarball contra libmicrohttpd-dev y libxml2-dev de Ubuntu 26.04.

El servicio entra en auto-restart

```
unable to open /etc/fort/config.json: No such file or directory
```

Solucion: crear /etc/fort/config.json antes de iniciar fort.service.

Permission denied al abrir [::]:323

```
[::]:323: Unable to bind the socket: Permission denied
```

Causa: FORT uso el puerto privilegiado 323. Solucion validada: usar la estructura JSON anidada y server.port 8323.

Python no puede leer config.json

```
PermissionError: [Errno 13] Permission denied: '/etc/fort/config.json'
```

Causa: el archivo tiene permisos 0640 root:fort. Validarlo con sudo o como usuario fort.

```
sudo python3 -m json.tool /etc/fort/config.json > /dev/null
```

18. Checklist final

- fort --version muestra 1.7.0.experimental.
- ldd no muestra bibliotecas not found.
- Los TAL existen en /etc/fort/tal.
- El repositorio local existe y pertenece a fort:fort.
- validated-roas.csv existe y contiene datos.
- /etc/fort/config.json es JSON valido y legible por el usuario fort.
- fort.service esta active (running) y enabled.
- FORT escucha en 10.0.1.15:8323.
- El router puede abrir TCP hacia 10.0.1.15:8323.
- La hora esta sincronizada y la zona del servidor es UTC.