Instalação e lançamento do FORT 1.7.0.experimental

Ubuntu Server 26.04 - Guia verificado passo a passo

Compilação a partir do código-fonte, validação RPKI, serviço *systemd* e acesso RTR da rede local.

Autor: Nicolas Antoniello (Git: 65007)

Abrangência Este guia consolida os passos que foram testados com sucesso no Ubuntu Server 26.04 para instalar o FORT 1.7.0.experimental.

1. Ambiente validado

Componente	Valor validado
Sistema operacional	Ubuntu Server 26.04 (Resolute)
Arquitetura	amd64 / x86_64
FORT	1.7.0.experimental
Método de instalação	Compilação do *tarball* oficial.
Diretório de configuração	/etc/fort
Repositório RPKI local	/var/lib/fort/repository
Saída VRP/ROA	/var/lib/fort/output/validated-roas.csv
Serviço	fort.service
Porta RTR	Neste guia, a porta TCP 8323 é a padrão; a porta TCP 323 é o padrão opcional mediante *capabilities* do *systemd*
Versão RTR máxima	1
Fuso horário recomendado	UTC

2. Por que o FORT é compilado a partir do código-fonte

O pacote oficial .deb não foi usado porque, embora seja uma opção que reduz significativamente as etapas de instalação e configuração, acredito que compilar o FORT oferece maiores possibilidades em diferentes sistemas operacionais para os quais o pacote pré-compilado poderia ainda não estar disponível.

A solução comprovada foi compilar o FORT usando as bibliotecas nativas do Ubuntu 26.04. Não force dependências nem misture pacotes de versões anteriores do Ubuntu ou Debian.

3. Instalar dependências de compilação

```
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
```

Verifique se o Ubuntu resolveu os pacotes esperados:

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

4. Baixar, compilar e instalar o 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
```

Verifique a instalação:

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

A localização esperada do binário após `make install` é:

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

5. Verificar as bibliotecas associadas

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

A instalação verificada incluiu, entre outras:

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

Nenhuma biblioteca deve aparecer como *not found*.

6. Criar a conta de serviço e os diretórios.

```
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 os arquivos 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. Executar a primeira validação 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
```

A primeira execução pode usar os dois vCPU disponíveis e baixar vários gigabytes. Isso é esperado. Durante o teste verificado, o repositório atingiu cerca de 2,7 GB durante a sincronização inicial.

Monitoramento recomendado de outra sessão:

```
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
```

O uso da CPU acima de 100% significa que o processo está utilizando mais de um núcleo. Com 2 vCPU, o máximo teórico é cerca de 200%.

9. Verificar o resultado da validação

```
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 durante a validação verificada:

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

10. Criar a configuração permanente

Crie o arquivo:

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

Conteúdo final verificado:

```
{
  "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"
  }
}
```

Substitua 10.0.1.15 se a VM usar um endereço privado diferente.

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

Valide a sintaxe JSON e as permissões de acesso:

```
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. Testar a configuração fora do *systemd*

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

A saída deve confirmar pelo menos os seguintes valores:

```
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
```

Deve também informar que todos os *sockets* configurados foram abertos corretamente. Interrompa este teste em primeiro plano com Ctrl+C antes de iniciar o serviço *systemd*.

12. Criar o serviço *systemd*

Crie o arquivo de unidade do serviço:

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

Insira o seguinte conteúdo no arquivo:

```
[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
```

Salve o arquivo, reinicie o *systemd* e habilite o serviço:

```
sudo systemctl daemon-reload
sudo systemctl enable --now fort
```

Verifique o estado do serviço e os registros recentes:

```
sudo systemctl status fort
sudo journalctl -u fort -n 100 --no-pager
```

13. Opcional: usar a porta RTR padrão 323 com *capabilities* de *systemd*

A porta TCP 323 é a porta padrão designada a RPKI-RTR. Como as portas inferiores a 1024 são privilegiadas no Linux, a conta de serviço não privilegiada *fort* não pode se vincular à porta 323, a menos que o *systemd* conceda a *capability* específica necessária para essa operação.

Este método mantém o FORT em execução como o usuário dedicado *fort* e não requer a execução do serviço como *root*.

Edite a unidade do serviço:

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

Adicione as duas diretivas a seguir dentro da seção [Service] existente:

```
AmbientCapabilities=CAP_NET_BIND_SERVICE
CapabilityBoundingSet=CAP_NET_BIND_SERVICE
```

A seção [Service] completa deve ficar da seguinte forma:

```
[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
AmbientCapabilities=CAP_NET_BIND_SERVICE
CapabilityBoundingSet=CAP_NET_BIND_SERVICE
```

Edite a configuração de FORT:

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

Troque a porta RTR de:

```
"port": "8323"
```

a:

```
"port": "323"
```

Valide o JSON, recarregue *systemd* e reinicie o FORT:

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

sudo systemctl daemon-reload
sudo systemctl restart fort
sudo systemctl status fort
```

Verificar se o FORT está escutando na porta TCP 323:

```
sudo ss -lntp | grep ':323'
```

Os *listeners* esperados incluem os endereços locais e LAN configurados, por exemplo:

```
127.0.0.1:323
[::1]:323
10.0.1.15:323
```

A partir de outro *host* ou roteador da mesma rede, teste a conectividade TCP:

```
nc -vz 10.0.1.15 323
```

Configure o roteador com:

```
RTR server: 10.0.1.15
RTR port: 323
RTR version: 1 or automatic negotiation
```

Se o UFW for ativado posteriormente, permitir a porta TCP 323 somente a partir do endereço do roteador, sempre que possível:

```
sudo ufw allow from ROUTER_IP_ADDRESS to 10.0.1.15 port 323 proto tcp
```

14. Verificar o RTR e o acesso a partir da rede local

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

Esperam-se *listeners* no *localhost* e no endereço privado da VM:

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

A partir de outro *host* ou roteador da mesma rede:

```
nc -vz 10.0.1.15 8323
```

Quando o roteador for conectado, verifique a sessão estabelecida no validador.

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

Configure os seguintes valores no roteador:

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

15. Considerações de *firewall*

O UFW estava desativado na instalação verificada. Ainda pode existir filtragem no nível do centro de dados, o nó ou a VM de *Proxmox*, o roteador ou entre VLAN e sub-redes.

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

Se o UFW estiver ativado, de preferência permita apenas o endereço do roteador:

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

O acesso a partir de outra rede roteada requer rotas de ida e volta. Se o acesso for feito através da Internet pública ou NAT, use uma VPN em vez de expor o RTR diretamente.

16. Hora e sincronização

A validação RPKI depende dos registros de tempo de certificados, manifestos e CRL. O relógio da VM deve permanecer sincronizado.

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

Valores esperados:

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

Recomenda-se UTC para servidores de infraestrutura:

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

Para exibir a hora local do Uruguai sem alterar o fuso horário do servidor:

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

17. Uso de recursos e comportamento esperado

- A primeira sincronização é a mais custosa; os ciclos subsequentes reutilizam o repositório local.
- Com 2 vCPU, o FORT pode se aproximar a 200% da CPU durante as fases intensivas de processamento.
- 4 GB de RAM foram suficientes no teste verificado; observou-se apenas uma pequena quantidade de *swap*, sem pressão de memória sustentada.
- Um breve período com CPU e E/S em zero imediatamente antes de finalizar pode ser normal.
- Não interrompa a primeira validação enquanto houver atividade da CPU, atividade de rede ou crescimento do repositório.

18. Resolução dos problemas encontrados

O pacote `.deb` não pode ser instalado

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

Solução: os autores corrigiram esse problema, portanto agora é possível instalar e executar o FORT usando diretamente o pacote oficial.

O serviço entra repetidamente em modo de reinicialização automática

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

Solução: crie /etc/fort/config.json antes de iniciar fort.service.

Permissão negada ao vincular `[::]:323`

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

Motivo: FORT usou a porta privilegiada 323 sem a *capability* necessária para vincular a uma porta inferior a 1024.
Solução: use a porta não privilegiada 8323, como na configuração principal, ou conceda a permissão CAP_NET_BIND_SERVICE mediante *systemd* e configure a porta padrão 323 conforme descrito na seção opcional.

Python não pode ler `config.json`

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

Motivo: as permissões do arquivo são 0640 e o proprietário é root:fort. Valide com *sudo* ou como o usuário fort:

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

19. Lista de verificação final

- [ ] fort --version informa 1.7.0.experimental.
- [ ] ldd não informa bibliotecas faltantes.
- [ ] Os arquivos TAL existem em /etc/fort/tal.
- [ ] O repositório local existe e pertence a fort:fort.
- [ ] validated-roas.csv existe e contém dados.
- [ ] /etc/fort/config.json é JSON válido e pode ser lido pela conta fort.
- [ ] fort.service está ativo (running) e habilitado.
- [ ] FORT escuta em 10.0.1.15:8323, ou em 10.0.1.15:323 quando se usa a configuração opcional de *capabilities* de *systemd*.
- [ ] O roteador pode estabelecer conectividade TCP com a porta RTR configurada.
- [ ] O relógio do sistema está sincronizado e o fuso horário do servidor é UTC.