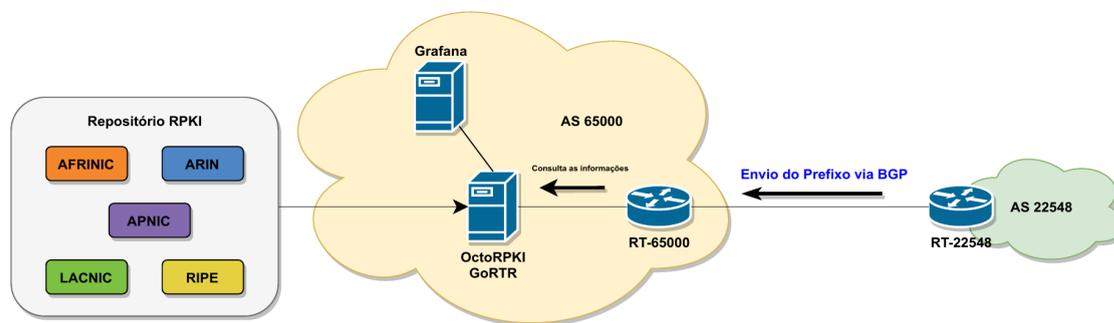


Laboratório OctoRPKI - Semana de Capacitação 2021 [27/09 à 01/10]

Nesse documento está contido o passo-a-passo da instalação e configuração do OctoRPKI e o GoRTR, bem como os scripts de configuração dos roteadores, utilizados no tutorial apresentado na Semana de Capacitação 2021 do NIC.br.

Segue a topologia utilizada para o laboratório.



Instalação e Configuração do OctoRPKI

Aqui está um passo-a-passo para a instalação e configuração básica do OctoRPKI. Lembrando que no laboratório utilizamos o sistema operacional Ubuntu Mate 20.04.

1. Criar uma pasta chamada 'octorpki', onde vamos baixar e instalar os serviços.

```
mkdir octorpki
cd octorpki
```

2. Baixar e instalar o OctoRPKI do [repositório oficial da Cloudflare](https://github.com/cloudflare/cfrpki/releases/download/v1.3.0/octorpki_1.3.0_amd64.deb), no momento do tutorial a versão mais recente do OctoRPKI é a [v1.3.0](https://github.com/cloudflare/cfrpki/releases/download/v1.3.0/octorpki_1.3.0_amd64.deb).

```
wget \
https://github.com/cloudflare/cfrpki/releases/download/v1.3.0/
octorpki_1.3.0_amd64.deb

dpkg -i octorpki_1.3.0_amd64.deb
```



3. Criar uma pasta e baixar os arquivos TAL (Trust Anchor Locator) e voltar para a pasta 'octorpki'.

```
mkdir tals
cd tals

wget \
https://raw.githubusercontent.com/cloudflare/cfrpki/master/cmd
/octorpki/tals/afrinic.tal

wget \
https://raw.githubusercontent.com/cloudflare/cfrpki/master/cmd
/octorpki/tals/apnic.tal

wget \
https://raw.githubusercontent.com/cloudflare/cfrpki/master/cmd
/octorpki/tals/lacnic.tal

wget \
https://raw.githubusercontent.com/cloudflare/cfrpki/master/cmd
/octorpki/tals/ripe.tal

wget \
https://www.arin.net/resources/manage/rpki/arin-rfc7730.tal -O
arin.tal

cd..
```

4. Iniciar o serviço em segundo plano na porta 8081.

```
nohup octorpki -output.sign=false > out 2> err &
```

5. Gera o arquivo JSON com as informações das ROAs

```
curl localhost:8081/output.json
```

Caso ocorra o erro ***File not ready yet***, aguarde de 5-10 minutos para que o arquivo ser gerado, após isso ele ficará disponível no link <http://localhost:8081/output.json>.

Instalação e Configuração do GoRTR

Aqui está um passo-a-passo para a instalação e configuração básica do **GoRTR**. Lembrando que no laboratório utilizamos o sistema operacional Ubuntu Mate 20.04.

1. Baixar e instalar o GoRTR do [repositório oficial da Cloudflare](#), no momento do tutorial a versão mais recente do OctoRPKI é a [v0.14.7](#).

```
wget \
https://github.com/cloudflare/gortr/releases/download/v0.14.7/
gortr_0.14.7_amd64.deb

dpkg -i gortr_0.14.7_amd64.deb
```

2. Iniciar o GoRTR, lembrando que o serviço do RTR roda na porta 8282 e as metrics (que podem ser utilizadas no Grafana) rodam na porta 8080.

```
nohup gortr -bind=:8282 -metrics.addr=:8080 -verify=false
-cache=http://localhost:8081/output.json > out 2> err &
```



Script RT-65000

```
!Script de Configuração Inicial do RT-65000
!Desenvolvido para o laboratório de OctoRPKI realizado para a
Semana de Capacitação do NIC.br (27/09 à 01/10 de 2021).

!Entra no modo de configuração e alterar o nome do roteador
enable
conf t
hostname RT-65000

!Ativa o Roteamento IPv6 no Roteador
ipv6 unicast-routing

!Configura os endereços IPv6 nas interface GigabitEthernet1 e
GigabitEthernet 2
interface g1
ipv6 address 2001:db8:faca::1/64
ipv6 address fe80::1 link-local
no shutdown
interface g2
ipv6 address 2001:db8:cafe::2/64
ipv6 address fe80::2 link-local
no shutdown

!Realiza a configuração da sessão BGP com o RT-22548 e publica
o prefixo do AS65000
router bgp 65000
no bgp default ipv4-unicast
bgp router-id 2.2.2.2
neighbor 2001:db8:cafe::1 remote-as 22548
address-family ipv6
neighbor 2001:db8:cafe::1 activate
network 2001:db8:faca::/64
exit

!Realizar a conexão com o servidor onde está instalado o
```



```
GoRTR, o roteador puxará a tabela de VRP (Verified ROA  
Payloads)  
bgp rpki server tcp 2001:db8:faca::10 port 8282 refresh 30  
  
!Salva as configurações realizadas  
do wr
```

Para verificar se a conexão do roteador com o servidor funcionou digite o comando.

```
#show bgp ipv4 unicast rpki servers
```

Para verificar a tabela de prefixos registrados digite.

```
#show bgp ipv6 unicast rpki tables
```

Para verificar se as rotas foram validadas corretamente digite o comando:

```
#show bgp ipv6 unicast
```



Script RT-22548

```
!Script de Configuração do RT-22548
!Desenvolvido para o laboratório de OctoRPKI realizado para a
Semana de Capacitação do NIC.br (27/09 à 01/10 de 2021).

!Entra no modo de configuração e altera o nome do roteador
enable
conf t
hostname RT-22548

!Ativa o Roteamento IPv6 no Roteador
ipv6 unicast-routing

!Configura os endereços IPv6 nas interface GigabitEthernet1 e
GigabitEthernet 2
interface g1
ipv6 address 2001:12ff::1/32
ipv6 address fe80::1 link-local
no shutdown
interface g2
ipv6 address 2001:db8:cafe::1/64
ipv6 address fe80::1 link-local
no shutdown

!Realiza a configuração da sessão BGP com o RT-65000 e publica
o prefixo do AS22548
router bgp 22548
no bgp default ipv4-unicast
bgp router-id 1.1.1.1
neighbor 2001:db8:cafe::2 remote-as 65000
address-family ipv6
neighbor 2001:db8:cafe::2 activate
network 2001:12ff::/32

!Salva as configurações realizadas
do wr
```

Monitoramento com Prometheus e o Grafana

Aqui está um passo-a-passo para a instalação e configuração básica do **Prometheus** e do **Grafana**. Lembrando que no laboratório utilizamos o sistema operacional Ubuntu Mate 20.04.

1. Baixar a pasta **compose** do [repositório oficial da Cloudflare](#). A pasta **compose** possui arquivos de configurações que podem ser utilizados para dar início ao monitoramento do **OctoRPKI** e do **GoRTR**.
2. Certifique-se que o docker e docker-compose estejam instalados em sua máquina.
3. Edite o arquivo de configuração do Prometheus, [prometheus.yml](#), adicionando os *endpoints* das métricas do OctoRPKI e do GoRTR. O servidor OctoRPKI e GoRTR está sendo executado na máquina com IP 192.168.0.114, no nosso exemplo.

```
global:
  scrape_interval:     15s
  evaluation_interval: 15s
alerting:
  alertmanagers:
  - static_configs:
    - targets:
rule_files:
scrape_configs:
  - job_name: 'prometheus'
    static_configs:
    - targets: ['localhost:9090']
  - job_name: 'gortr'
    static_configs:
    - targets: ['192.168.0.114:8080']
  - job_name: 'octorpki'
    static_configs:
    - targets: [192.168.0.114:8081']
```



4. Inicie o Prometheus e o Grafana.

```
docker-compose up -d
```

O Grafana fica disponível em <http://localhost:3000>, usuário: admin, senha: admin.

Após fazer login no Grafana, você pode utilizar o **OctoRPKI - GoRTR Dashboard** para visualizar o status da validação do RPKI, o health do servidor RTR e outras métricas.