Exercício 8a - Routinator

Objetivo: instalar o validador RPKI Routinator para verificar os anúncios RPKI da Internet. Nesse caso em específico vamos validar apenas os anúncios feitos no laboratório.

Preparo máquina Linux:

1. Entre no equipamento LinuxKrill
   login: ceptro
   senha: ceptro
2. Abra o Terminal do LinuxKrill
3. Ligue a interface ens4:

```
#sudo ip link set dev ens4 up
```

4. Adicione os seguintes endereços na interface ens4:

```
#sudo ip address add 172.16.0.10/24 dev ens4
#sudo ip address add 2001:db8::10/64 dev ens4
```

5. Verifique se os endereços foram adicionados com sucesso:

```
#ip address show ens4
```

Passo a passo de instalação do Routinator:

1. Entre no equipamento LinuxKrill
   login: ceptro
   senha: ceptro
2. Abra o Terminal do Linux QTerminal

3. Instale o Routinator

```
#sudo -i
#apt install routinator -y
```

4. O Routinator já vem com as TALs referentes aos RIRs (exceto do ARIN). Dessa forma basta adicionar o TAL do ARIN e já é possível validar todas as ROAs do RPKI. No entanto, para esse lab não queremos validar toda a Internet, mas sim as ROAs anunciadas dentro do laboratório através do beta do Registro. Para isso precisamos remover os TALs originais e substituir pelo TAL do beta.

```
#mkdir /var/lib/routinator/tals
#cd /var/lib/routinator/tals
#wget https://rpki-test-ta.beta.registro.br/ta/ta.tal
```

5. Execute o Routinator e verifique as ROAs recebidas

```
#routinator -q --no-rir-tals --extra-tals-dir="/var/lib/routinator/tals" vrps
```

6. Suba o routinator no modo RTR (Router to RPKI)

```
#routinator -q --no-rir-tals --extra-tals-dir="/var/lib/routinator/tals"
server --rtr 172.16.0.10:3323 --rtr [2001:db8::10]:3323
```

Exercício 8b - Validando pelo Roteador

Objetivo: Configurar o Mikrotik (possível apenas na versão 7 do RouterOS) para conversar com o Routinator via protocolo RTR

1. Entre no equipamento Mikrotik7
   login: admin
   senha: **<sem senha>**

2. Adicione os endereços IPv4 e IPv6 no Mikrotik

```
/ip/address/add address=172.16.0.20/24 interface=ether1
/ipv6/address/add address=2001:db8::20/64 interface=ether1
```

3. Coloque as informações do validador Routinator no Mikrotik

```
/routing/rpki
add group=RPKIGrupoXX address=172.16.0.10 port=3323
refresh-interval=20
```

4. Verifique se a sessão foi estabelecida

```
/routing/rpki print
```

5. Para verificar se determinada rota é válida pelo RPKI utilize o comando

```
/routing/rpki rpki-check group=RPKIGrupoXX prefix=<prefixo>
origin-as=<ASN>
```

Alguns exemplos para verificar:

```
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.XX.0.0/21
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.XX.0.0/22
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.XX.0.0/23
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.XX.2.0/23
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.XX.0.0/24
origin-as=655XX
```

```
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:XX::/31
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:XX::/32
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:XX::/33
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:XX:8000:/33
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:XX::/34
origin-as=655XX
```

Obs: verifique a validade dos demais grupos também trocando YY pelo número dos outros grupos!

```
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.YY.0.0/21
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.YY.0.0/22
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.YY.0.0/23
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.YY.2.0/23
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=10.YY.0.0/24
origin-as=655YY

/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:YY::/31
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:YY::/32
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:YY::/33
origin-as=655YY
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:YY:8000:/33
origin-as=655XX
/routing/rpki rpki-check group=RPKIGrupoXX prefix=3fff:YY::/34
origin-as=655YY
```

6. Aplique os filtros para filtrar as rotas **inválidas**

```
/routing/filter/rule
add chain=bgp_in rule="rpki-verify RPKIGrupoXX"
add chain=bgp_in rule="if (rpki invalid) { reject } else { accept }"
```