公開用

Internet Week 2024

ROAキャッシュサーバハンズオン

～RPKI/ROVの普及を目指して～

2024/11/21 14:00 – 17:00

長崎県立大学　コーディネータ 岡田 雅之
講師・アシスタント 齋藤 脩愉・後藤 汰珠・井上 七星

自己紹介

- 名前：岡田 雅之
- 所属：長崎県立大学 情報セキュリティ学科
  - 教授
  - （2020年までJPNIC）
- 出身：茨城県生まれ 千葉育ち
- 趣味：
  - AS運用

自己紹介

- 名前：後藤 汰珠
- 所属：長崎県立大学 情報セキュリティ学科
  - 学部 4年
- 出身：大分県大分市
  - とり天やカボスが有名
- 趣味：
  - ゲーム(MMORPG),料理

自己紹介

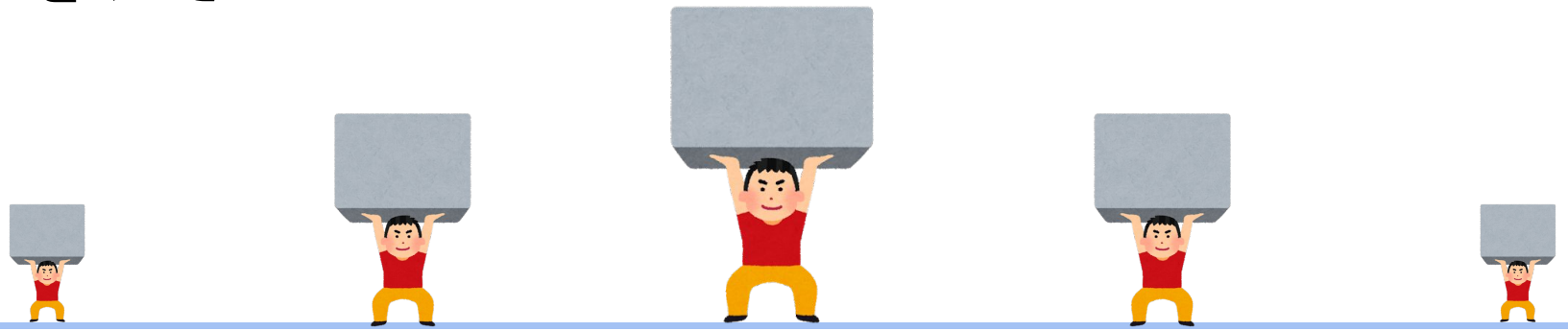
- 名前：齋藤 脩愉
- 所属：長崎県立大学 情報セキュリティ学科
  - 学部 4年
- 出身：香川県高松市
  - うどん県が最近、ヤドン県になっていた
- 趣味：
  - 小室哲哉（ファンクラブはGOLD会員）

自己紹介

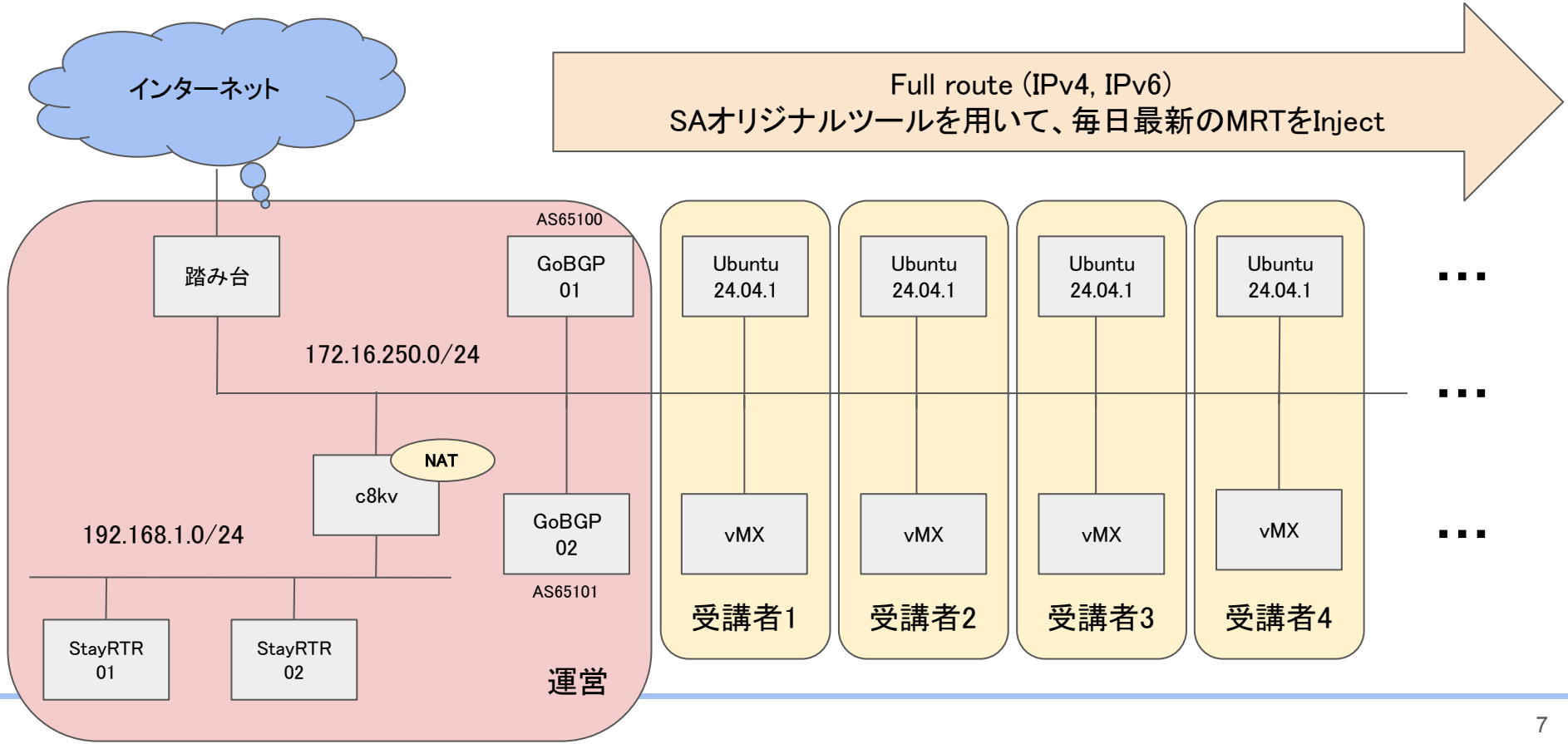
- 名前：井上 七星
- 所属：長崎県立大学 情報セキュリティ学科
  - 学部2年
- 出身：長崎県長崎市->博多->長崎
- 趣味：
  - Linuxを触ったり
  - 自宅NWの改革計画中…

本日のハンズオンは若干、ヘビーです

- RPKIをすでに理解されている方、Linuxを一般程度に操作できる方を除いて、本日のハンズオンは若干重めです
  - 講師側でシナリオ検証した際は、半日以上かかりました
- 少しでも躓いた、わからないなどあればお気軽に挙手・チャットをください

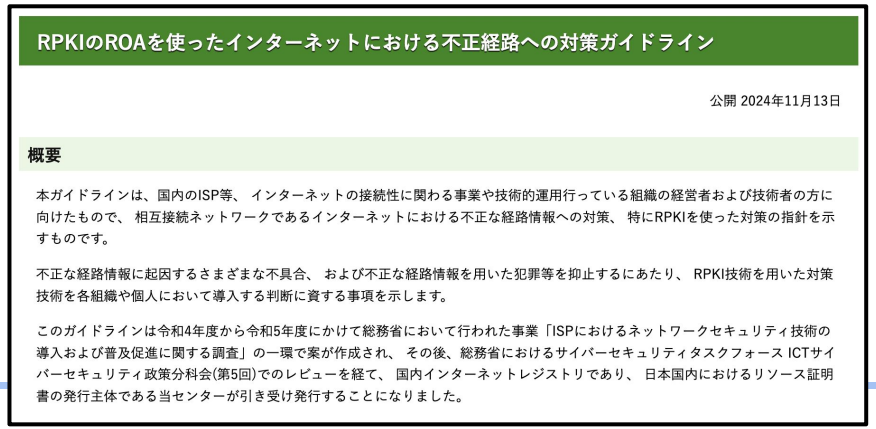


本日の環境


インターネット
Full route (IPv4, IPv6)
SAオリジナルツールを用いて、毎日最新のMRTをInject
AS65100
踏み台
GoBGP
01
172.16.250.0/24
NAT
c8kv
192.168.1.0/24
GoBGP
02
AS65101
StayRTR
01
StayRTR
02
運営
Ubuntu
24.04.1
Ubuntu
24.04.1
Ubuntu
24.04.1
Ubuntu
24.04.1
vMX
vMX
vMX
vMX
受講者1
受講者2
受講者3
受講者4

参考：ガイドライン

- 2024年11月13日 JPNICより「RPKIのROAを使ったインターネットにおける不正経路への対策ガイドライン」が公開された
  - このガイドラインには別紙として、各ベンダOSごとのROVの設定例が記載されている



RPKIのROAを使ったインターネットにおける不正経路への対策ガイドライン

公開 2024年11月13日

概要

本ガイドラインは、国内のISP等、 インターネットの接続性に関わる事業や技術的運用行っている組織の経営者および技術者の方に向けたもので、 相互接続ネットワークであるインターネットにおける不正な経路情報への対策、 特にRPKIを使った対策の指針を示すものです。

不正な経路情報に起因するさまざまな不具合、 および不正な経路情報を用いた犯罪等を抑止するにあたり、 RPKI技術を用いた対策技術を各組織や個人において導入する判断に資する事項を示します。

このガイドラインは令和4年度から令和5年度にかけて総務省において行われた事業「ISPにおけるネットワークセキュリティ技術の導入および普及促進に関する調査」の一環で案が作成され、 その後、総務省におけるサイバーセキュリティタスクフォース ICTサイバーセキュリティ政策分科会(第5回)でのレビューを経て、 国内インターネットレジストリであり、 日本国内におけるリソース証明書の発行主体である当センターが引き受け発行することになりました。

Routinatorとは？

- NLnet Labs提供のオープンソースソフトウェア
  - サポートも存在する
- rcyncを用いたROAの収穫とrtrを用いたルータへのデータ提供を同時に行うことのできるソフトウェア
  - どちらか片方のみの利用も可能
- Rust言語で書かれている
  - スレッドセーフ
  - メモリ安全
- Web GUIが存在する

Routinatorとは？

- 今回ハンズオンで使用するRoutinatorのバージョンは**0.14.0**
- Ubuntu 22.04はリポジトリを追加すれば、aptでインストールできる
  - Ubuntu 24.04はリポジトリが無いのでビルドする

ハンズオンの流れ

ハンズオンの流れ

1. 接続準備
2. Routinatorの構築・接続
3. ROVと各種パラメータの挙動確認
4. SLURMの設定・挙動確認
5. トラブルシューティング
6. Tips及びまとめ

ハンズオンの流れ

1. 接続準備
2. **Routinatorの構築・接続**
3. ROVと各種パラメータの挙動確認
4. SLURMの設定・挙動確認
5. トラブルシューティング
6. Tips及びまとめ

Routinatorの導入 -下準備-

- ユーザroutinatorを作成
- 今回は/opt/routinator下に設定などを置きます

```
sudo apt install -y curl rsync build-essential
sudo useradd --system -d /opt/routinator routinator
sudo install -d -o routinator /opt/routinator
sudo install -d -o routinator /var/lib/routinator/
sudo -u routinator bash
cd ~
```

Routinatorの導入

- Ubuntu上で次のコマンドを実行
- 1行目rustupの導入時に質問されますが、**そのままEnter**

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
. "$HOME/.cargo/env"
cargo install --locked routinator
```

Routinatorの導入

```
nano /opt/routinator/routinator.conf
```

```
repository-dir = "/var/lib/routinator/rpki-cache"
rtr-listen = ["0.0.0.0:3323"]
http-listen = ["0.0.0.0:80"]
```

```
(Ctrl + oで保存, Ctrl + xでexit)
exit
```

Routinatorの導入

```
cd /usr/lib/systemd/system/
curl
https://raw.githubusercontent.com/NLnetLabs/routinator/refs/heads/main/pkg/com
mon/routinator.routinator.service | sudo tee routinator.service
```

curl ~ .serviceまでは一続き！
途中に改行はありません

Routinatorの導入

- ExecStart=から始まる行を次のように書き換え、保存してください

```
sudo nano routinator.service
```

```
ExecStart=/opt/routinator/.cargo/bin/routinator
--config=/opt/routinator/routinator.conf --syslog
--exceptions=/opt/routinator/slurm.json server
```

--exceptionでSLURMファイルを指定
次ページで空のSLURMを生成します

Routinatorの導入 -空のSLURMを生成 -

- 後述するツールを用いて、空のSLURMを生成する

```
cd ~

sudo python3 control-slurm.py
```

```
Create base SLURM file successful. PATH: /opt/routinator/slurm.json
```

と表示されれば成功

SLURMの読み込み

- Routinatorの起動オプションに
  - `--exceptions <SLURM記述ファイル>`
- を追加すればSLURMがRoutinatorで使用可
- 検証の度にファイルが読み込まれる
  - SLURMの変更時にいちいちRoutinatorの再起動をしなくても良い
- ここでは、空のSLURMを適用しておき、適宜加工していく

Routinatorの導入

- Routinatorを起動する

```
sudo systemctl daemon-reload
sudo systemctl enable --now routinator
sudo systemctl status routinator
```

```
user@iw24-user28-server:~$ sudo systemctl status routinator
● routinator.service - Routinator 3000
     Loaded: loaded (/usr/lib/systemd/system/routinator.service; enabled; preset: enabled)
     Active: active (running) since Mon 2024-11-18 16:43:46 JST; 8s ago
       Docs: man:routinator(1)
   Main PID: 7796 (routinator)
      Tasks: 15 (limit: 9443)
     Memory: 5.5M (peak: 6.2M)
        CPU: 144ms
     CGroup: /system.slice/routinator.service
             └─7796 /opt/routinator/.cargo/bin/routinator --config=/opt/routinator/routinator.conf --syslog --exceptions=/opt/routinator/slurm.json server

Nov 18 16:43:46 iw24-user28-server systemd[1]: Starting routinator.service - Routinator 3000...
Nov 18 16:43:46 iw24-user28-server systemd[1]: Started routinator.service - Routinator 3000.
Nov 18 16:43:46 iw24-user28-server routinator[7796]: Using config file /opt/routinator/routinator.conf.
```

Routinatorの導入

- ROAを取得している様子を確認してみる
  - Ctrl + cでjournalctlを終了できる

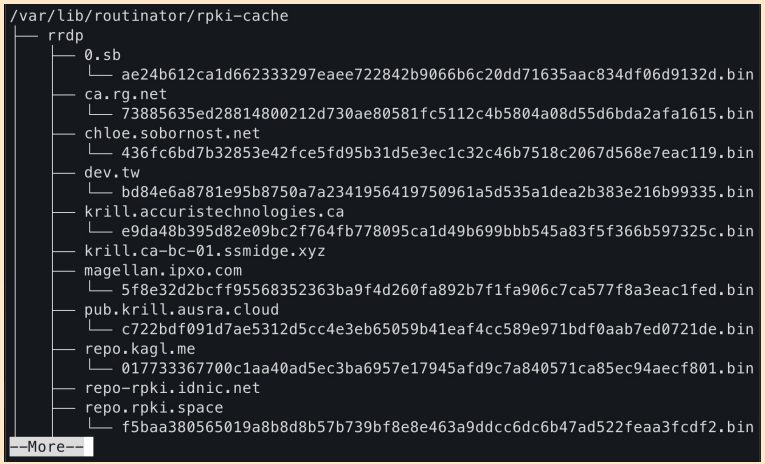
```
sudo journalctl -xeu routinator.service -f
```

```
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rsync.paas.rpki.ripe.net/repository/311e47c4-02f0-4657-9022-d2b83cff8755/1/EEB9FB9A329DDB5B94FC5E2424E572ECB9EEDE52.mft
: certificate has expired.
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rsync.paas.rpki.ripe.net/repository/84d51810-1987-4701-8f1f-84251111964f4/0/EEB9FB9A329DDB5B94FC5E2424E572ECB9EEDE52.cer
: no valid manifest rsync://rsync.paas.rpki.ripe.net/repository/311e47c4-02f0-4657-9022-d2b83cff8755/1/EEB9FB9A329DDB5B94FC5E2424E572ECB9EEDE52.mft found.
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rpki.ripe.net/repository/DEFAULT/xVV8l9e7_0fKIq1fufBYn6rxWb0.cer: no valid manifest rsync://rsync.paas.rpki.ripe.net/re
pository/32bfd357-d83b-400a-8c46-4fbd1119f4a3/2/C5557C97D7BBFF47CA22AD5FB9F0589FAAF159BD.mft found.
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rpki.ripe.net/repository/DEFAULT/xZR-zIaDrQ282VqPMy8MqhNXR5A.cer: no valid manifest rsync://rpki.netiface.net/repo/Civi
lized/0/C5947ECC8683AD0DBCD95A8F332F0CAA13574790.mft found.
Nov 18 23:28:32 iw-r77 routinator[90964]: RRDP https://rpki01.hel-fi.rpki.win/rrdp/notification.xml: HTTP status server error (502 Bad Gateway) for url (https://rpki01.h
el-fi.rpki.win/rrdp/notification.xml)
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rpki01.hel-fi.rpki.win:44595/repo/: Dubious host name. Skipping update.
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://rpki.ripe.net/repository/DEFAULT/yJsxCB1b3QjRj7zY-r7oHE-wUUY.cer: no valid manifest rsync://rpki01.hel-fi.rpki.win:4459
5/repo/as60900/0/C89B31081D5BDD08D18FBCD8FABEE81C4FB05146.mft found.
Nov 18 23:28:32 iw-r77 routinator[90964]: rsync://chloe.sobornost.net/rpki/RIPE-nljobsnijders/9X0AhXWTJDl8lJhfOwvnac-42CA.spl: unknown object type.
Nov 18 23:28:34 iw-r77 routinator[90964]: rsync://rpki.luys.cloud/repo/LY-RPKI/1/47717A8A2E301D872DAC129F2CBAC587F01A0813.mft: certificate has expired.
Nov 18 23:28:34 iw-r77 routinator[90964]: rsync://sakuya.nat.moe/repo/NATOCA/1/47717A8A2E301D872DAC129F2CBAC587F01A0813.cer: no valid manifest rsync://rpki.luys.cloud/re
po/LY-RPKI/1/47717A8A2E301D872DAC129F2CBAC587F01A0813.mft found.
Nov 18 23:28:35 iw-r77 routinator[90964]: RRDP https://rpki.miralium.net/rrdp/notification.xml: error sending request for url (https://rpki.miralium.net/rrdp/notificatio
n.xml)
Nov 18 23:28:35 iw-r77 routinator[90964]: rsync://rpki.miralium.net/repo/: rsync: getaddrinfo: rpki.miralium.net 873: No address associated with hostname
Nov 18 23:28:35 iw-r77 routinator[90964]: rsync://rpki.miralium.net/repo/: rsync error: error in socket IO (code 10) at clientserver.c(139) [Receiver=3.2.7]
Nov 18 23:28:35 iw-r77 routinator[90964]: rsync://krill.accuristechnologies.ca/repo/Accuris-Technologies/0/8A15107195E63966ABA1997AD31382979C75F736.cer: no valid manifes
t rsync://rpki.miralium.net/repo/Miralium-Research-RPKI-CA-A1/0/8A15107195E63966ABA1997AD31382979C75F736.mft found.
Nov 18 23:28:35 iw-r77 routinator[90964]: rsync://rpki.komorebi.network/repo/komorebi/1/7C9EE20E59DEF5F8B5FE68585EDE906D17D3040C.cer: no valid manifest rsync://krill.uta
.ng/repo/pongery/3/7C9EE20E59DEF5F8B5FE68585EDE906D17D3040C.mft found.
Nov 18 23:28:36 iw-r77 routinator[90964]: rsync://rpki.xa.wiki/repo/AXNETS-CA/1/3073BED8ADF785ABB31D49FB9E84FDB524550720.mft: certificate has expired.
Nov 18 23:28:36 iw-r77 routinator[90964]: rsync://rsync.paas.rpki.ripe.net/repository/3253d973-d5bf-4541-bcc1-276543a25c7d/0/3073BED8ADF785ABB31D49FB9E84FDB524550720.cer
: no valid manifest rsync://rpki.xa.wiki/repo/AXNETS-CA/1/3073BED8ADF785ABB31D49FB9E84FDB524550720.mft found.
Nov 18 23:28:56 iw-r77 routinator[90964]: RRDP https://krill.stonham.uk/rrdp/notification.xml: HTTP status server error (522 <unknown status code>) for url (https://kril
l.stonham.uk/rrdp/notification.xml)
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://rpki.cnnic.cn/rpki/A9162E3D0000/1029/A_XQOpMuSOf5T61oEuoovsj1LQI.mft: certificate has expired.
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://rpki.cnnic.cn/rpki/A9162E3D0000/A_XQOpMuSOf5T61oEuoovsj1LQI.cer: no valid manifest rsync://rpki.cnnic.cn/rpki/A9162E3D0
000/1029/A_XQOpMuSOf5T61oEuoovsj1LQI.mft found.
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://rpki-repository.haruue.net/repo/YC3254-RPKI/2/3F0AC25D352C83DA8307594B98ED061BE8489682.mft: certificate has expired.
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://cloudie-repo.rpki.app/repo/CLOUDIE-RPKI/0/3F0AC25D352C83DA8307594B98ED061BE8489682.cer: no valid manifest rsync://rpki-
repository.haruue.net/repo/YC3254-RPKI/2/3F0AC25D352C83DA8307594B98ED061BE8489682.mft found.
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://rpki.xa.wiki/repo/AXNETS-CA/0/1717ACD294A2C988167DF42BBB5A5E20A0C2831B.mft: certificate has expired.
Nov 18 23:28:56 iw-r77 routinator[90964]: rsync://rsync.paas.rpki.ripe.net/repository/89270f6c-a3fe-4299-b079-309ed97f3824/0/1717ACD294A2C988167DF42BBB5A5E20A0C2831B.cer
: no valid manifest rsync://rpki.xa.wiki/repo/AXNETS-CA/0/1717ACD294A2C988167DF42BBB5A5E20A0C2831B.mft found.
Nov 18 23:29:06 iw-r77 routinator[90964]: rsync://krill.stonham.info/repo/: rsync error: timeout waiting for daemon connection (code 35) at socket.c(278) [Receiver=3.2.7
]
Nov 18 23:29:06 iw-r77 routinator[90964]: rsync://cloudie-repo.rpki.app/repo/CLOUDIE-RPKI/0/635C29FF238CC286AC1625A68EFCC04E2E460171.cer: no valid manifest rsync://krill
.stonham.info/repo/Stonham/1/635C29FF238CC286AC1625A68EFCC04E2E460171.mft found.
```

Routinatorの導入

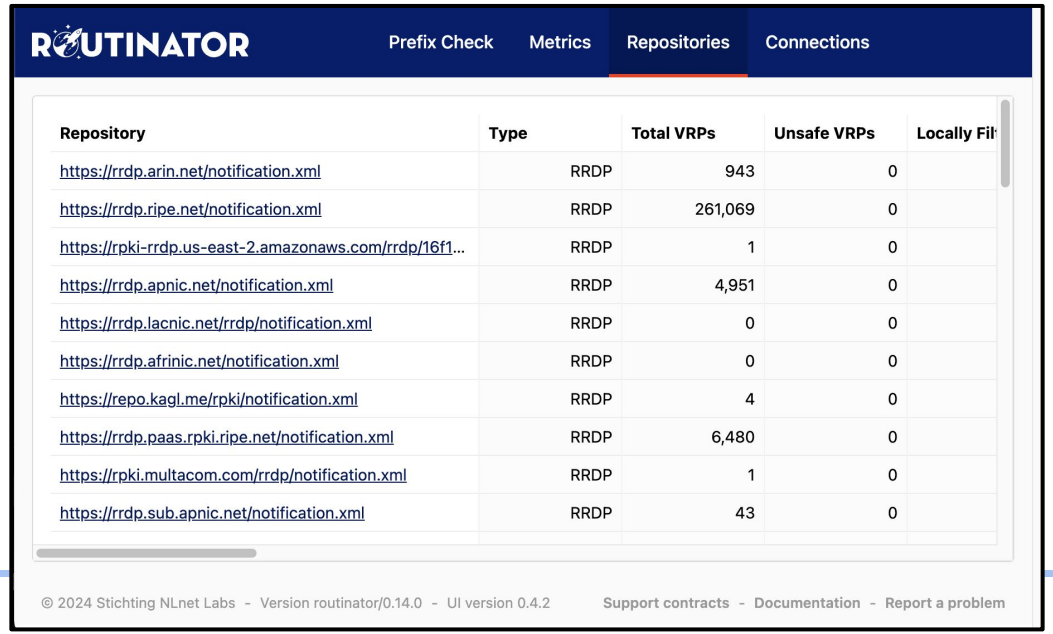
- 収穫したROAは/var/lib/routinator/rpki-cacheに格納されている

```
tree /var/lib/routinator/rpki-cache | more
```



Routinatorの導入

- 今回はSSHのみを許可しているため、確認できないがWebでRoutinatorの状態を確認することもできる



Routinatorの導入

- 国内ISPにとっては重要であろう、JPNICのリポジトリについての収穫状況を確認する

```
curl http://localhost:80/metrics | grep nic.ad.jp
```

```
[root@localhost /]# curl http://localhost:80/metrics | grep nic.ad.jp
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0routinator_repository_publication_points_total{uri="https://
routinator_repository_publication_points_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", state="rejected"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="manifest", state="valid"} 530
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="manifest", state="invalid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="manifest", state="premature"} 
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="manifest", state="stale"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="manifest", state="missing"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="crl", state="valid"} 530
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="crl", state="invalid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="crl", state="stale"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="crl", state="stray"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="ca_cert", state="valid"} 526
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="router_cert", state="valid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="cert", state="invalid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="roa", state="valid"} 4759
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="roa", state="invalid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="gbr", state="valid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="gbr", state="invalid"} 0
routinator_repository_objects_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml", type="other", state="invalid"} 0
routinator_repository_valid_vrps_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml"} 4759
routinator_repository_locally_filtered_vrps_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml"} 0
routinator_repository_duplicate_vrps_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml"} 0
routinator_repository_contributed_vrps_total{uri="https://rpki-repository.nic.ad.jp/rrdp/ap/notification.xml"} 4759
```

取得したROAの総数が確認可能

RTRプロトコルとは

- キャッシュが収穫したRPKIデータをIPアドレスとAS番号の正しい組み合わせのリスト(VRP)としてルータに引き渡すためのプロトコル
- RFC8210で定義
  - https://tex2e.github.io/rfc-translater/html/rfc8210.html

Junos基本操作

- ここから先はJunosの操作をしていく
  - operation mode
    - 黒色背景で示します(プロンプトが>になります)
  - configuration mode
    - 黄色背景で示します(プロンプトが#になります)
    - operation modeからはconfigureコマンドで移行
    - 抜け出す時はexitコマンド
    - runを先頭に付けると一時的にoperation modeで実行

Junos基本操作

- configure modeで、
  - 変更はcommitしない限り適用されません
  - 変更を適用する前に、次のオススメコマンドを
    - commit check
      - 変更が論理的に正しいかチェック
    - show
      - 現在の設定をすべて表示
    - show | compare
      - 変更した部分のみを表示
    - commit and-quit
      - 変更を適用して、operation modeへ遷移

接続準備

- 一旦Ubuntuから退室します(exit)
- 操作したいマシン名(vmx)を入力してEnterします
  - **serverのIPを覚えて** おきましょう(入力してEnter)

```
------------------------------MACHINE LIST-------------------------------
Username: user28
    NAME     IP
    server   172.16.250.64
    vmx      172.16.250.65
    exit     Do exit
Machine name?: vmx
Machine Password is iw24-rpki
user@172.16.250.64's password:
```

Junos-Routinatorの接続

```
configure
```

```
set routing-options validation group RPKI session {serverのIPアドレス} port
3323
commit check
commit and-quit
```

Junos-Routinatorの接続

- Routinatorとの接続状況を確認する

```
user@iw24-user28-vmx> show validation session detail
Session 172.16.250.64, State: up, Session index: 3
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 0
  Serial (Incremental Update): 0
    Session flaps: 0
    Session uptime: 00:01:13
    Last PDU received: 00:01:07
    IPv4 prefix count: 227377
    IPv6 prefix count: 48814
```

Junos-Routinatorの接続

- Routinatorとの接続状況を確認する

```
user@iw24-user28-vmx> show validation group
master
  Group: RPKI, Maximum sessions: 2
    Session 172.16.250.64, State: Up, Preference: 100
```

Junos-Routinatorの接続

- VRPのデータベースを確認してみる

```
user@iw24-user28-vmx> show validation database
RV database: default
Prefix                  Origin-AS Session                                 State   Mismatch
1.178.112.0/20-24           12975 172.16.250.64                           valid   *
1.178.112.0/20-24           19905 172.16.250.64                           valid   *
1.178.128.0/20-20           12975 172.16.250.64                           valid
1.178.128.0/20-24           19905 172.16.250.64                           valid
1.178.208.0/20-24           12975 172.16.250.64                           valid   *
1.178.208.0/20-24           19905 172.16.250.64                           valid   *
1.178.224.0/19-24           12479 172.16.250.64                           valid
1.178.224.0/21-21           12479 172.16.250.64                           valid
1.178.232.0/21-21           12479 172.16.250.64                           valid
1.178.240.0/21-21           12479 172.16.250.64                           valid
1.178.248.0/21-21           12479 172.16.250.64                           valid
1.179.40.0/21-21            12975 172.16.250.64                           valid
 …（以下、省略）
```

ハンズオンの流れ

1. 接続準備
2. Routinatorの構築・接続
3. **ROVと各種パラメータの挙動確認**
4. SLURMの設定・挙動確認
5. トラブルシューティング
6. Tips及びまとめ

事前確認

- フルルートが受信できているか、確認する

```
user@iw24-user28-vmx> show bgp summary
Threading mode: BGP I/O
Default eBGP mode: advertise - accept, receive - accept
Groups: 1 Peers: 2 Down peers: 0
Table            Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0
                    988923     988922          0          0          0          0
inet6.0
                    216725     216724          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn
State|#Active/Received/Accepted/Damped...
172.16.250.241        65101     989141        239       0       5     1:47:30 Establ
  inet.0: 988922/988923/988923/0
2400:5320:ff00:16fa::241       65101     216942        237       0       5     1:47:29
Establ
  inet6.0: 216724/216725/216725/0
```

JunosでROVを行う (1/2)

```
configure
```

```
set policy-options policy-statement ROV term valid from protocol bgp
set policy-options policy-statement ROV term valid from validation-database valid
set policy-options policy-statement ROV term valid then local-preference 200
set policy-options policy-statement ROV term valid then validation-state valid
set policy-options policy-statement ROV term valid then accept
set policy-options policy-statement ROV term invalid from protocol bgp
set policy-options policy-statement ROV term invalid from validation-database invalid
set policy-options policy-statement ROV term invalid then local-preference 50
set policy-options policy-statement ROV term invalid then validation-state invalid
set policy-options policy-statement ROV term invalid then accept
```

JunosでROVを行う (2/2)

```
set policy-options policy-statement ROV term unknown from protocol bgp
set policy-options policy-statement ROV term unknown from validation-database unknown
set policy-options policy-statement ROV term unknown then local-preference 100
set policy-options policy-statement ROV term unknown then validation-state unknown
set policy-options policy-statement ROV term unknown then accept
set protocols bgp group GoBGP import ROV
commit check
commit and-quit
```

JunosでROVを行う（ show | compare）

```
[edit]
user@iw24-user28-vmx# show | compare rollback 2
[edit policy-options]
+   policy-statement ROV {
+       term valid {
+           from {
+               protocol bgp;
+               validation-database valid;
+           }
+           then {
+               local-preference 200;
+               validation-state valid;
+               accept;
+           }
+       }
+       term invalid {
+           from {
+               protocol bgp;
+               validation-database invalid;
+           }
+           then {
+               local-preference 50;
+               validation-state invalid;
+               accept;
+           }
+       }
+       term unknown {
+           from {
+               protocol bgp;
+               validation-database unknown;
+           }
+           then {
+               local-preference 100;
+               validation-state unknown;
+               accept;
+           }
+       }
+   }
[edit protocols bgp group GoBGP]
+    import ROV;
```

JunosでROVを行う（確認）

- ROVの統計情報を確認する

```
user@iw24-user28-vmx> show validation statistics
Total RV records: 625023
Total Replication RV records: 625023
  Prefix entries: 561148
  Origin-AS entries: 625023
Memory utilization: 244453712 bytes
RV database: default
  RV records in Database: 625023
  Origin-AS entries in Database: 625023
  Database origin-validation re-evaluation statistics: 6944695
    Attempts resulting Valid: 1950411
    Attempts resulting Invalid: 11134
    Attempts resulting Unknown: 4983150
  BGP import policy reevaluation notifications: 84
    inet.0, 82
    inet6.0, 2
Policy origin-validation re-evaluation statistics: 6944695
  Attempts resulting Valid: 1950411
  Attempts resulting Invalid: 11134
  Attempts resulting Unknown: 4983150
BGP import policy reevaluation notifications: 84
```

JunosでROVを行う（確認）

- Validな経路を確認する

```
user@iw24-user28-vmx> show route validation-state valid

inet.0: 988926 destinations, 988926 routes (988925 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.0.0/24          *[BGP/170] 00:17:37, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 13335 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.0.64.0/18         *[BGP/170] 00:17:48, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 2519 7670 18144 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.1.1.0/24          *[BGP/170] 00:17:37, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 13335 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.6.0.0/22          *[BGP/170] 00:17:42, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 6453 9583 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.6.1.0/24          *[BGP/170] 00:17:37, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 6453 9583 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

JunosでROVを行う（確認）

- Invalidな経路を確認する

```
user@iw24-user28-vmx> show route validation-state invalid

inet.0: 988926 destinations, 988926 routes (988925 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

1.6.168.0/24       *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 ?, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
1.6.169.0/24       *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 I, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
1.6.183.0/24       *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 I, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
1.6.219.0/24       *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 137130 I, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
1.6.247.0/24       *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 I, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

JunosでROVを行う（確認）

- Unknownな経路を確認する

```
user@iw24-user28-vmx> show route validation-state unknown

inet.0: 988926 destinations, 988926 routes (988925 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.4.0/22          *[BGP/170] 00:19:52, localpref 100, from 172.16.250.241
                       AS path: 65101 2497 6453 7545 2764 38803 38803 38803 I, validation-state: unknown
                     >  to 172.16.250.254 via fxp0.0
1.0.4.0/24          *[BGP/170] 00:19:47, localpref 100, from 172.16.250.241
                       AS path: 65101 2497 6453 7545 38803 I, validation-state: unknown
                     >  to 172.16.250.254 via fxp0.0
1.0.5.0/24          *[BGP/170] 00:19:47, localpref 100, from 172.16.250.241
                       AS path: 65101 2497 6453 7545 38803 I, validation-state: unknown
                     >  to 172.16.250.254 via fxp0.0
1.0.6.0/24          *[BGP/170] 00:19:47, localpref 100, from 172.16.250.241
                       AS path: 65101 2497 6453 7545 38803 I, validation-state: unknown
                     >  to 172.16.250.254 via fxp0.0
1.0.7.0/24          *[BGP/170] 00:19:47, localpref 100, from 172.16.250.241
                       AS path: 65101 2497 6453 7545 38803 I, validation-state: unknown
                     >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

validation-state

- validation-stateの意味は、次の通り
  - valid
    - ROAレコードとprefix及びAS番号が完全一致している
  - invalid
    - ROAレコードとprefix及びAS番号が完全一致していない
  - unknown
    - 対応するROAレコードが存在しない
  - unverified
    - 未検証のレコード

local-prefrence

- local-prefrenceとは、経路選択時の優先度を示すものであり、数値が高いほど優先度は高く扱われる
- 今回は、ROV結果に応じて次の値をセットしている
  - valid
    - 200
  - invalid
    - 50　※不正経路のため実環境ではrejectすることが望ましい
  - unknown
    - 100

Invalid経路をreject

- Invalidな経路をrejectしましょう

```
configure
```

```
set policy-options policy-statement ROV term invalid then reject
commit check
commit and-quit
```

Invalid経路をrejectする（確認）

- Invalidな経路を確認する
  - 先ほどと違い、該当する経路が無いことがわかる
  - hiddenをつければreject経路を確認できる

```
user@iw24-user28-vmx> show route validation-state invalid

inet.0: 988926 destinations, 988926 routes (987384 active, 0 holddown, 1542 hidden)

inet6.0: 216730 destinations, 216730 routes (216405 active, 0 holddown, 325 hidden)

user@iw24-user28-vmx> show route validation-state invalid hidden

inet.0: 988926 destinations, 988926 routes (987384 active, 0 holddown, 1542 hidden)
+ = Active Route, - = Last Active, * = Both

1.6.168.0/24        [BGP ] 00:00:55, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 ?, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

Preference値について

- 接続の優先順位を決めるための値
- 値の大きいものが優先され、デフォルトは100
  - 最も優先度の高いRPKIキャッシュサーバから順に接続
- 次ページより、動作確認のための手順を踏んでいく
  - Stay-RTRとの接続

Preference値の挙動確認（事前準備）

```
configure
```

```
set routing-options validation group RPKI session 172.16.250.245 port 3323
commit check
commit and-quit
```

Preference値の挙動確認（事前準備）

- Stay-RTRとの接続状況を確認する

```
user@iw24-user28-vmx> show validation session
Session                                 State    Flaps     Uptime #IPv4/IPv6 records
172.16.250.64                           Up           0   02:51:41 502348/122688
172.16.250.245                          Up           0   00:00:14 502345/122682

user@iw24-user28-vmx> show validation group
master
  Group: RPKI, Maximum sessions: 2
    Session 172.16.250.64, State: Up, Preference: 100
    Session 172.16.250.245, State: Up, Preference: 100
```

Preference値の挙動確認

- group RPKI内で利用するROAキャッシュサーバを1台に限定
  - max-sessions 1
- Preference値
  - Routinatorを200, Stay-RTRを100にセット

Preference値の挙動確認

```
configure
```

```
set routing-options validation group RPKI max-sessions 1
set routing-options validation group RPKI session {serverのIPアドレス} preference 200
set routing-options validation group RPKI session 172.16.250.245 preference 100
commit check
commit and-quit
```

Preference値の挙動確認

- 接続状況を確認する

```
user@iw24-user28-vmx> show validation session
Session                                 State    Flaps     Uptime #IPv4/IPv6 records
172.16.250.64                           Up           0   02:58:48 502343/122693
172.16.250.245                          Connect      1            502345/122682

user@iw24-user28-vmx> show validation group
master
  Group: RPKI, Maximum sessions: 1
    Session 172.16.250.64, State: Up, Preference: 200
    Session 172.16.250.245, State: Connect, Preference: 100
```

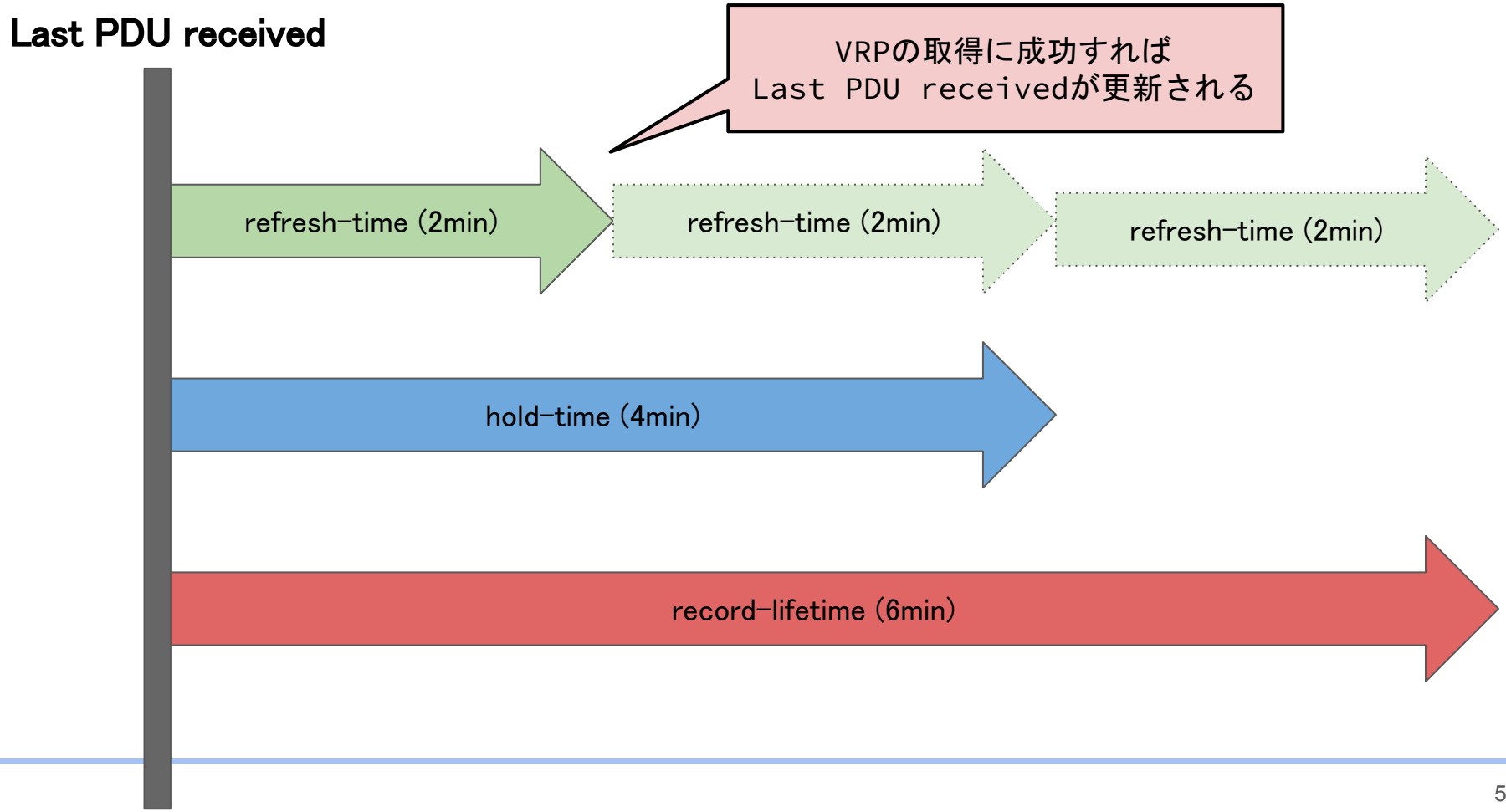
Preference値の挙動確認

- もし、両方ともUpならclear validation session

```
user@iw24-user28-vmx> clear validation session
```

各種タイマ値について

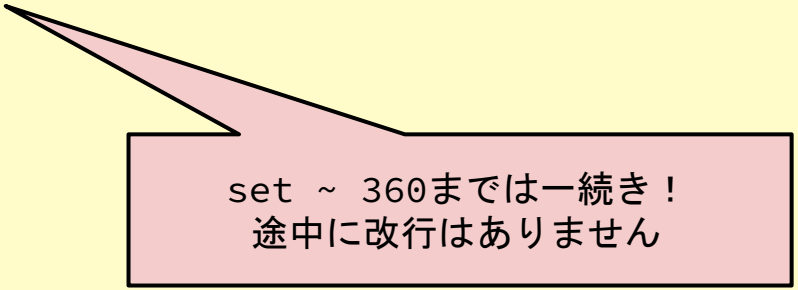
- refresh-time
  - ルータがキャッシュに対してデータ更新を要求する間隔
  - 今回は120秒 (=2分)にセット
- hold-time
  - キャッシュとの接続が切れた際に接続を有効とみなす時間
  - 今回は240秒 (=4分)にセット
- record-lifetime
  - 同一のデータを保持する最大時間
  - 今回は360秒 (=6分)にセット


Last PDU received
VRPの取得に成功すれば
Last PDU receivedが更新される
refresh-time (2min)
refresh-time (2min)
refresh-time (2min)
hold-time (4min)
record-lifetime (6min)

各種タイマ値の挙動確認

```
configure
```

```
delete routing-options validation group RPKI max-sessions
set routing-options validation group RPKI session 172.16.250.245 refresh-time
120 hold-time 240 record-lifetime 360
commit check
commit and-quit
```



各種タイマ値の挙動確認

- 接続状況を確認する

```
user@iw24-user28-vmx> show validation session detail
Session 172.16.250.64, State: up, Session index: 3
  Group: RPKI, Preference: 200
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 20
  Serial (Incremental Update): 21
    Session flaps: 1
    Session uptime: 00:15:46
    Last PDU received: 00:04:12
    IPv4 prefix count: 502349
    IPv6 prefix count: 122696
Session 172.16.250.245, State: up, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 1
    Session uptime: 00:00:46
    Last PDU received: 00:00:35
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- これからSAがStay-RTRの接続を切ります
- show validation session 172.16.250.245 detailを定期的に叩きながら状況を監視してみる
  - Last PDU receivedを特に見る

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: up, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 1
    Session uptime: 00:03:14
    Last PDU received: 00:01:03
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- Last PDU receivedが2分未満
  - 変化なし

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: up, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 1
    Session uptime: 00:23:52
    Last PDU received: 00:00:14
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- Last PDU receivedが2分を超え、4分未満
  - Refresh-timeの周期で更新しようとしたが、疎通性なし
  - StateがUpからexchange-incrementalへ遷移

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: exchange-incremental, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 1
    Session uptime: 00:27:54
    Last PDU received: 00:02:05
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- Last PDU receivedが4分を超え、6分未満
  - Hold-time経過でも疎通性復帰しないので、切断と判断
  - StateがExchange-incrementalからconnectへ遷移

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: connect, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 2
    Last PDU received: 00:05:31
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- Last PDU receivedが6分を超える
  - record-lifetime経過でも疎通性復帰しないので、VRPがExpire(期限切れ)する

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: connect, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 2
    Last PDU received: 00:06:02
    IPv4 prefix count: 0
    IPv6 prefix count: 0
```

VRPがExpireしたら。。。

- 1つのROAキャッシュサーバの接続のみであれば、ROVが不可能になるのでValidation-stateはunknownに遷移する
- 今回は、Routinatorがまだ接続されているので継続してROVできている

VRPがExpireしたら。。。

- 接続状況とROVの状況を確認

```
user@iw24-user28-vmx> show validation session
Session                                 State   Flaps     Uptime #IPv4/IPv6 records
172.16.250.64                           Up          1   00:53:27 502307/122699
172.16.250.245                          Connect     2            0/0

user@iw24-user28-vmx> show route validation-state valid

inet.0: 988926 destinations, 988926 routes (987380 active, 0 holddown, 1546 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.0.0/24          *[BGP/170] 01:52:20, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 13335 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.0.64.0/18         *[BGP/170] 01:52:31, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 2519 7670 18144 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

各種タイマ値の挙動確認

- これからSAがStay-RTRの接続を戻します
- show validation session 172.16.250.245 detailを定期的に叩きながら状況を監視してみる
  - Last PDU receivedを特に見る

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: up, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 8
  Serial (Incremental Update): 8
    Session flaps: 1
    Session uptime: 00:03:14
    Last PDU received: 00:01:03
    IPv4 prefix count: 502345
    IPv6 prefix count: 122682
```

各種タイマ値の挙動確認

- refresh-timeを待たずして、セッションが戻る

```
user@iw24-user28-vmx> show validation session 172.16.250.245 detail
Session 172.16.250.245, State: up, Session index: 5
  Group: RPKI, Preference: 100
  Port: 3323
  Refresh time: 120s
  Hold time: 240s
  Record Life time: 360s
  Serial (Full Update): 9
  Serial (Incremental Update): 9
    Session flaps: 2
    Session uptime: 00:00:39
    Last PDU received: 00:00:28
    IPv4 prefix count: 502305
    IPv6 prefix count: 122700
```

ハンズオンの流れ

1. 接続準備
2. Routinatorの構築・接続
3. ROVと各種パラメータの挙動確認
4. **SLURMの設定・挙動確認**
5. トラブルシューティング
6. Tips及びまとめ

SLURMとは

- RFC8416で定義されている
- ROAを無視してVRPを上書きすることが可能
- validationOutputFiltersとは
  - 対象prefixをキャッシュの出力から削除(無効化)する項目
  - 記載されたprefixは「unknown」として扱われる
- locallyAddedAssertionsとは
  - 対象prefixをvalidとしてキャッシュから出力させる項目
  - 記載されたprefixは「valid」として扱われる

SLURMとは

- 中身はJSONファイル
  - 手書きやjqコマンドの使用によって編集が可能
- もう少し人間に優しくするために、今回のハンズオン用にツールを自作
  - 皆さんのホームディレクトリに**control-slurm.py** を置いている

SLURMの設定と挙動確認（確認）

- Invalidな経路を確認する
  - どれか一つ選んで、そのPrefixを覚えておく(例: 1.6.168.0/24)

```
user@iw24-user28-vmx> show route validation-state invalid hidden

inet.0: 988926 destinations, 988926 routes (988925 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

1.6.168.0/24        *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                       AS path: 65101 2500 17676 6453 4755 9583 ?, validation-state: invalid
                     >  to 172.16.250.254 via fxp0.0
1.6.169.0/24        *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                       AS path: 65101 2500 17676 6453 4755 9583 I, validation-state: invalid
                     >  to 172.16.250.254 via fxp0.0
1.6.183.0/24        *[BGP/170] 00:18:55, localpref 50, from 172.16.250.241
                       AS path: 65101 2500 17676 6453 4755 9583 I, validation-state: invalid
                     >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

SLURMの設定と挙動確認（確認）

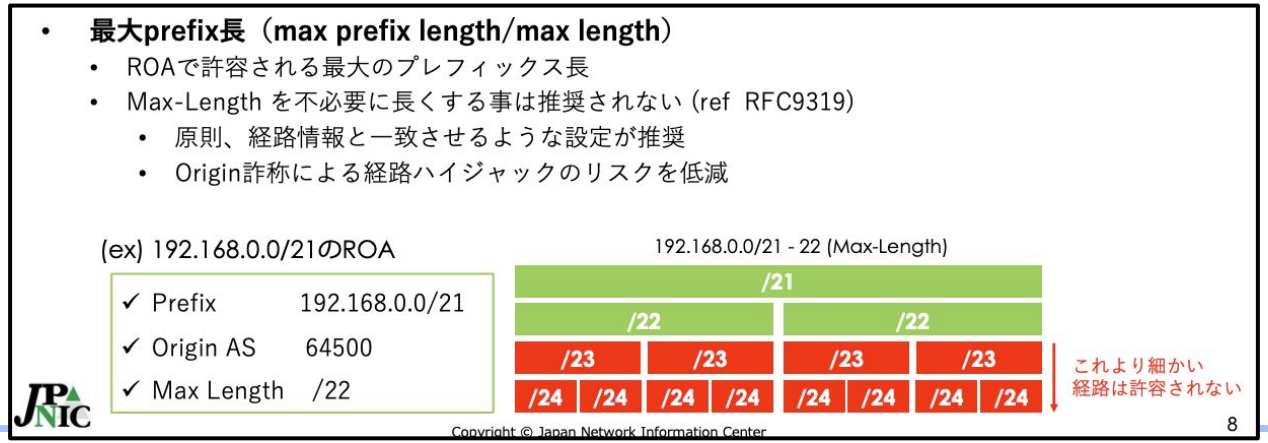
- このPrefixに関連するVRPは存在するか確認する
  - Prefix（Maskなし）で検索
  - IPアドレスの部分は適宜、自分の使用するPrefixに置き換える！

```
user@iw24-user28-vmx> show validation database record 1.6.168.0
RV database: default
Prefix                 Origin-AS Session                                 State   Mismatch
1.6.168.0/22-22             9583 172.16.250.64                           valid
1.6.168.0/22-22             9583 172.16.250.245                          valid

  IPv4 records: 2
  IPv6 records: 0
```

SLURMの設定と挙動確認（確認）

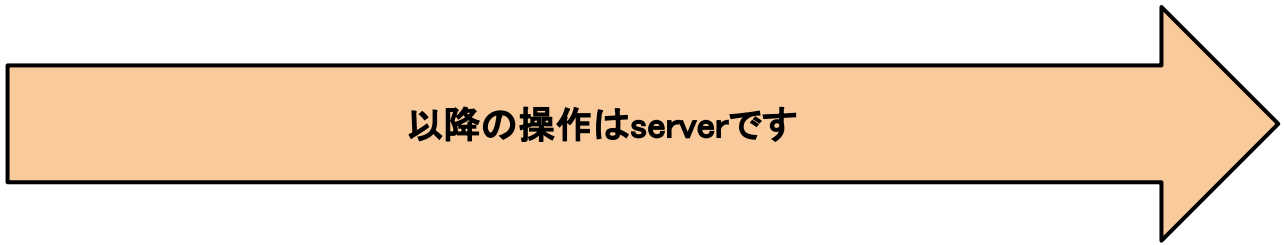
- 先ほどの確認より、
  - 1.6.168.0/22は最大マスク長/22までのROAが存在する
  - /24で広報されている経路はカバーされていないのでInvalid
- Max Prefixと経路のカバーとは



https://www.janog.gr.jp/meeting/janog51/wp-content/uploads/2023/01/janog51-roa-shiozawa.pdf

SLURMの設定と挙動確認（確認）

- 1.6.168.0/24をカバーするSLURMを作成する
  - origin AS9583



SLURMの設定

- 空のSLURMを確認する

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py --view-plane
Path: /opt/routinator/slurm.json

{
    "slurmVersion": 1,
    "validationOutputFilters": {
        "prefixFilters": [],
        "bgpsecFilters": []
    },
    "locallyAddedAssertions": {
        "prefixAssertions": [],
        "bgpsecAssertions": []
    }
}
```

SLURMの設定

- 空のSLURMを確認する（優しい出力もできる）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -v
validationOutputFilters
     prefixFilters
          No items.
locallyAddedAssertions
     prefixAssertions
          No items.
```

SLURMの設定

- ちなみに、ツールヘルプ（ヘルプ出力）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -h
help
        description: Display help
        option: -h, --help
init
        description: Initialize SLURM
        option: -i, --init
view
        description: Check the current SLURM file
        option: -v, --view
view-plane
        description: Display SLURM file as is
        option: --view-plane
add
        description: Add the SLURM information
        option: -a, --add
delete
        description: Delete the SLURM information
        option: -d, --del
```

SLURMの設定

- SLURMを追加する(1.6.168.0/24, AS9583)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -a
1: prefixFilters / 2: prefixAssertions?: 2
ASN?: 9583
Prefix?: 1.6.168.0/22
MaxPrefixLength /?: 24
comment?:

Are you sure you want to add the following information?
-----------------------------------------------------
Type: locallyAddedAssertions.prefixAssertions
ASN: 9583
Prefix: 1.6.168.0/22
MaxPrefixLength: /24
y/n: y
```

SLURMの設定

- SLURMを確認する（優しい出力も）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -v
validationOutputFilters
     prefixFilters
          No items.
locallyAddedAssertions
     prefixAssertions
          asn:9583, prefix:1.6.168.0/22, maxPrefixLength:24,
```

SLURMの設定

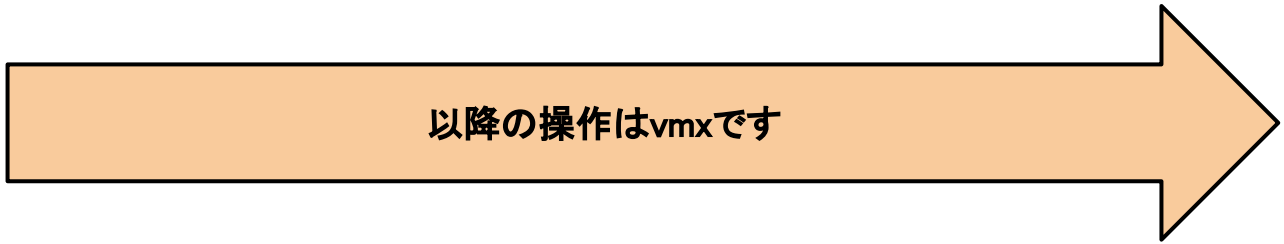
- SLURMを確認する(そのまま出力)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py --view-plane
Path: /opt/routinator/slurm.json

{
    "slurmVersion": 1,
    "validationOutputFilters": {
        "prefixFilters": [],
        "bgpsecFilters": []
    },
    "locallyAddedAssertions": {
        "prefixAssertions": [
            {
                "asn": 9583,
                "prefix": "1.6.168.0/22",
                "maxPrefixLength": 24
            }
        ],
        "bgpsecAssertions": []
    }
}
```

SLURMの挙動確認

- SLURMが届いているか確認する



SLURMの挙動確認

- SLURMが届いているか確認する

```
user@iw24-user28-vmx> show validation database record 1.6.168.0
RV database: default
Prefix                 Origin-AS Session                                 State   Mismatch
1.6.168.0/22-22             9583 172.16.250.64                           valid
1.6.168.0/22-22             9583 172.16.250.245                          valid
1.6.168.0/22-24             9583 172.16.250.64                           valid

  IPv4 records: 3
  IPv6 records: 0
```

SLURMの挙動確認

- 対象のPrefixがValidに変化したか確認する

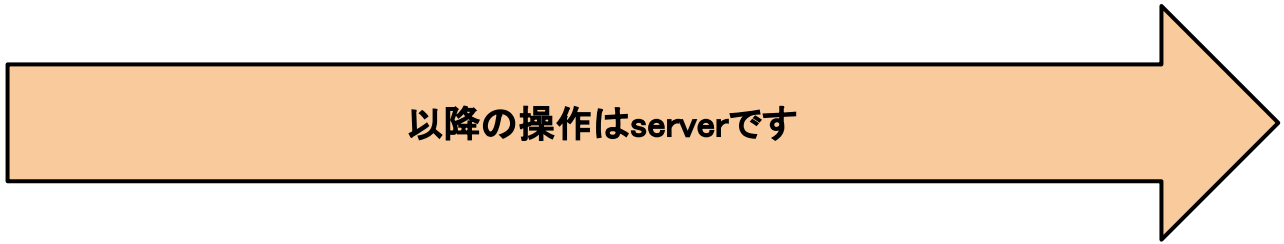
```
user@iw24-user28-vmx> show route 1.6.168.0/24

inet.0: 988926 destinations, 988926 routes (987382 active, 0 holddown, 1544 hidden)
+ = Active Route, - = Last Active, * = Both

1.6.168.0/24       *[BGP/170] 00:05:09, localpref 200, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 ?, validation-state: valid
                    >  to 172.16.250.254 via fxp0.0
```

SLURMの削除

- SLURMを削除する



SLURMの削除

- SLURMを削除する (1.6.168.0/24, AS9583)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -d
1: prefixFilters / 2: prefixAssertions?: 2
ASN?: 9583
Prefix?: 1.6.168.0/22
MaxPrefixLength /?: 24

Are you sure you want to delete the following information?
-----------------------------------------------------
asn:9583, prefix:1.6.168.0/22, maxPrefixLength:24,

y/n: y
```

SLURMの削除

- SLURMを確認する（優しい出力も）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -v
validationOutputFilters
     prefixFilters
          No items.
locallyAddedAssertions
     prefixAssertions
          No items.
```

SLURMの削除

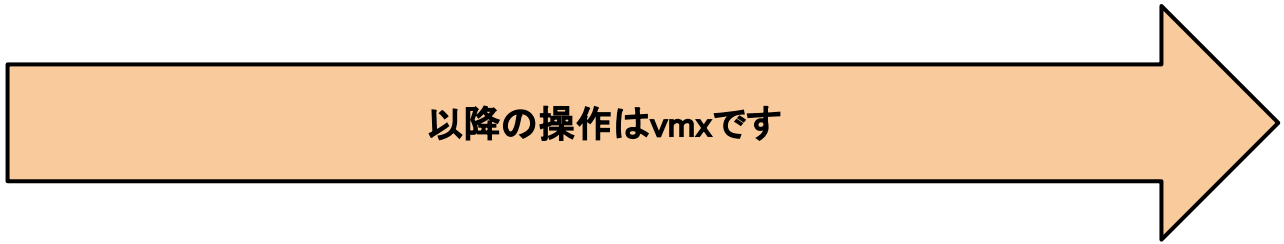
- SLURMを確認する(そのまま出力)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py --view-plane
Path: /opt/routinator/slurm.json

{
    "slurmVersion": 1,
    "validationOutputFilters": {
        "prefixFilters": [],
        "bgpsecFilters": []
    },
    "locallyAddedAssertions": {
        "prefixAssertions": [],
        "bgpsecAssertions": []
    }
}
```

SLURMの挙動確認

- SLURMが削除されたか確認する



SLURMの挙動確認

- SLURMが削除されたか確認する

```
user@iw24-user28-vmx> show validation database record 1.6.168.0
RV database: default
Prefix                 Origin-AS Session                                 State   Mismatch
1.6.168.0/22-22             9583 172.16.250.64                           valid
1.6.168.0/22-22             9583 172.16.250.245                          valid

  IPv4 records: 2
  IPv6 records: 0
```

SLURMの挙動確認

- 対象のPrefixがInvalidに変化したか確認する

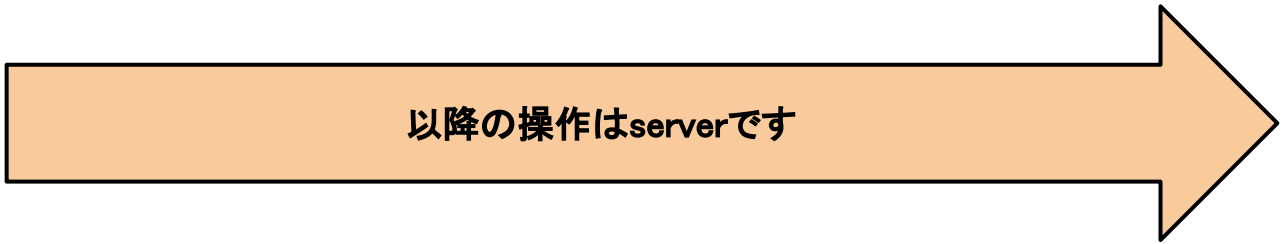
```
user@iw24-user28-vmx> show route 1.6.168.0/24 hidden

inet.0: 988926 destinations, 988926 routes (987382 active, 0 holddown, 1544 hidden)
+ = Active Route, - = Last Active, * = Both

1.6.168.0/24        [BGP ] 00:04:41, localpref 50, from 172.16.250.241
                      AS path: 65101 2500 17676 6453 4755 9583 ?, validation-state: invalid
                    >  to 172.16.250.254 via fxp0.0
```

SLURMによるROAの無効化

- 先程は、Invalidな経路をValid判定に変化させました
  - 意図しないInvalid経路への対応策を確認した
- ここから、Validな経路をUnknown判定に変化させます



SLURMによるROAの無効化（事前準備）

- Stay-RTRとの接続を無効化します

```
configure
```

```
deactivate routing-options validation group RPKI session 172.16.250.245
commit check
commit and-quit
```

SLURMによるROAの無効化（確認）

- Validな経路を確認する
  - どれか一つ選んで、そのPrefixを覚えておく(例: 1.0.0.0/24)

```
user@iw24-user28-vmx> show route validation-state valid

inet.0: 988926 destinations, 988926 routes (987382 active, 0 holddown, 1544 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.0.0/24          *[BGP/170] 03:18:44, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 13335 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.0.64.0/18         *[BGP/170] 03:18:55, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 2519 7670 18144 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
1.1.1.0/24          *[BGP/170] 03:18:44, localpref 200, from 172.16.250.241
                       AS path: 65101 2497 13335 I, validation-state: valid
                     >  to 172.16.250.254 via fxp0.0
 …（以下、省略）
```

SLURMによるROAの無効化（確認）

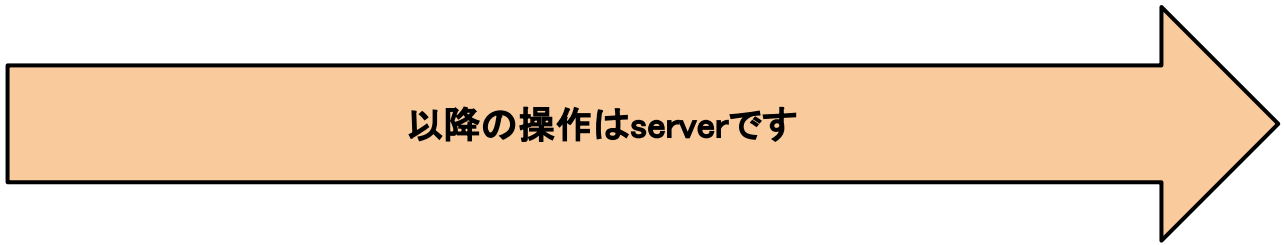
- このPrefixに関連するVRPは存在するか確認する
  - Prefix（Maskなし）で検索
  - IPアドレスの部分は適宜、自分の使用するPrefixに置き換える！

```
user@iw24-user28-vmx> show validation database record 1.0.0.0
RV database: default
Prefix                 Origin-AS Session                                 State   Mismatch
1.0.0.0/24-24              13335 172.16.250.64                           valid

  IPv4 records: 1
  IPv6 records: 0
```

SLURMによるROAの無効化

- 1.0.0.0/24を無効化するSLURMを作成する
  - origin AS13335



SLURMによるROAの無効化

- SLURMを追加する(1.0.0.0/24, AS13335)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -a
1: prefixFilters / 2: prefixAssertions?: 1
ASN?: 13335
Prefix?: 1.0.0.0/24
comment?:

Are you sure you want to add the following information?
-------------------------------------------------------
Type: validationOutputFilters.prefixFilters
ASN: 13335
Prefix: 1.0.0.0/24
y/n: y
```

SLURMによるROAの無効化

- SLURMを確認する（優しい出力も）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -v
validationOutputFilters
     prefixFilters
          asn:13335, prefix:1.0.0.0/24,

locallyAddedAssertions
     prefixAssertions
          No items.
```

SLURMによるROAの無効化

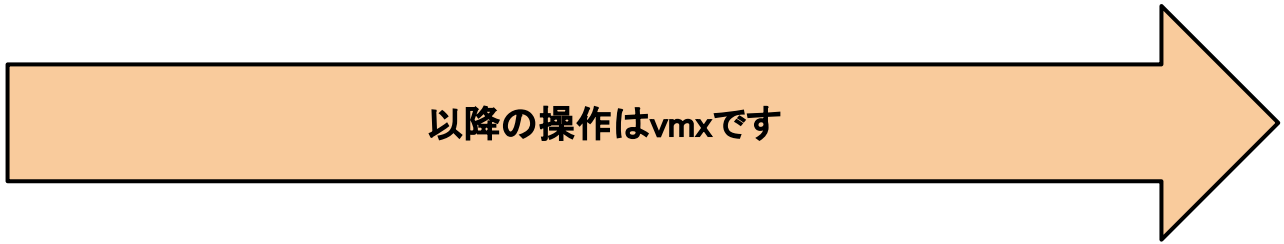
- SLURMを確認する（そのまま出力）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py --view-plane
Path: /opt/routinator/slurm.json

{
    "slurmVersion": 1,
    "validationOutputFilters": {
        "prefixFilters": [
            {
                "asn": 13335,
                "prefix": "1.0.0.0/24"
            }
        ],
        "bgpsecFilters": []
    },
    "locallyAddedAssertions": {
        "prefixAssertions": [],
        "bgpsecAssertions": []
    }
}
```

SLURMによるROAの無効化

- SLURMが無効化されているか確認する



SLURMによるROAの無効化

- ROAが無効化されているか確認する

```
user@iw24-user28-vmx> show validation database record 1.0.0.0
RV database: default

  IPv4 records: 0
  IPv6 records: 0
```

SLURMによるROAの無効化

- 対象のPrefixがUnknownに変化したか確認する

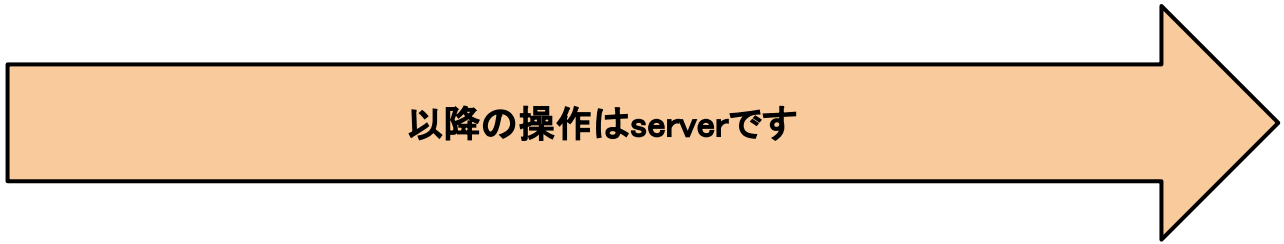
```
user@iw24-user28-vmx> show route 1.0.0.0/24

inet.0: 988926 destinations, 988926 routes (987376 active, 0 holddown, 1550 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.0.0/24          *[BGP/170] 00:03:40, localpref 100, from 172.16.250.241
                      AS path: 65101 2497 13335 I, validation-state: unknown
                    >  to 172.16.250.254 via fxp0.0
```

SLURMによるROAの無効化（削除）

- SLURMを削除する



SLURMによるROAの無効化（削除）

- SLURMを削除する (1.0.0.0/24, AS13335)

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -d
1: prefixFilters / 2: prefixAssertions?: 1
ASN?: 13335
Prefix?: 1.0.0.0/24

Are you sure you want to delete the following information?
------------------------------------------------------
asn:13335, prefix:1.0.0.0/24,

y/n: y
```

SLURMによるROAの無効化（削除）

- SLURMを確認する（優しい出力も）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py -v
validationOutputFilters
     prefixFilters
          No items.
locallyAddedAssertions
     prefixAssertions
          No items.
```

SLURMによるROAの無効化（削除）

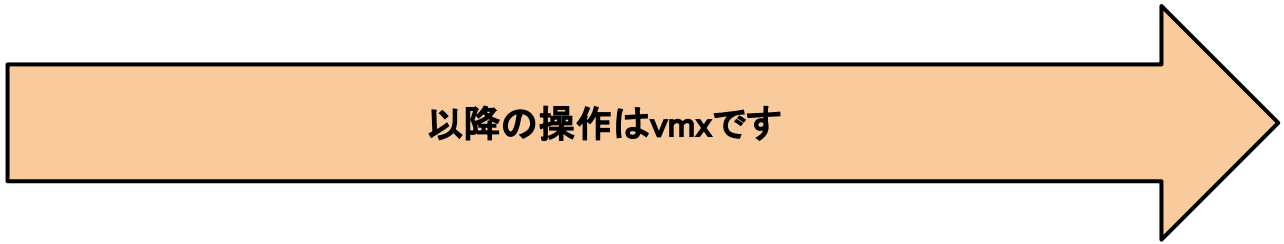
- SLURMを確認する（そのまま出力）

```
user@iw24-user28-server:~$ sudo python3 control-slurm.py --view-plane
Path: /opt/routinator/slurm.json

{
    "slurmVersion": 1,
    "validationOutputFilters": {
        "prefixFilters": [],
        "bgpsecFilters": []
    },
    "locallyAddedAssertions": {
        "prefixAssertions": [],
        "bgpsecAssertions": []
    }
}
```

SLURMによるROAの無効化（削除の確認）

- SLURMが削除されたか確認する



SLURMによるROAの無効化（削除の確認）

- SLURMが削除されたか確認する

```
user@iw24-user28-vmx> show validation database record 1.0.0.0
RV database: default
Prefix                 Origin-AS Session                                 State   Mismatch
1.0.0.0/24-24              13335 172.16.250.64                           valid

  IPv4 records: 1
  IPv6 records: 0
```

SLURMによるROAの無効化（削除の確認）

- 対象のPrefixがValidに変化したか確認する

```
user@iw24-user28-vmx> show route 1.0.0.0/24

inet.0: 988926 destinations, 988926 routes (987376 active, 0 holddown, 1550 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.0.0/24          *[BGP/170] 00:10:46, localpref 100, from 172.16.250.241
                      AS path: 65101 2497 13335 I, validation-state: valid
                    >  to 172.16.250.254 via fxp0.0
```

ハンズオンの流れ

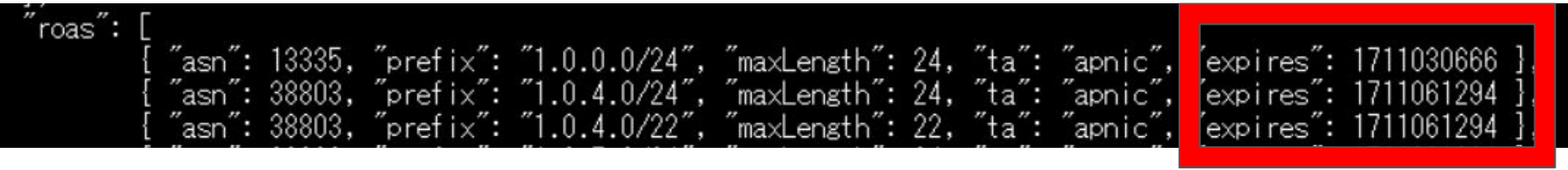
1. 接続準備
2. Routinatorの構築・接続
3. ROVと各種パラメータの挙動確認
4. SLURMの設定・挙動確認
5. **トラブルシューティング**
6. Tips及びまとめ

セッション再接続後に、Upにならない

- ROAキャッシュサーバとの接続が切れ、Expire後にセッションが復活したとき、全てのVRPを取得しきらずにexchange-fullやexchange-incrementalのまま変わらないことがある
  - clear validation session {serverのIPアドレス}で刺激を与える

ROAキャッシュサーバが最新のROAを取得できない時

- VRP上に記載されたrecordの有効期限が更新されなくなり、徐々に有効期限が切れrouter上で無効なrecordとして扱われるようになる



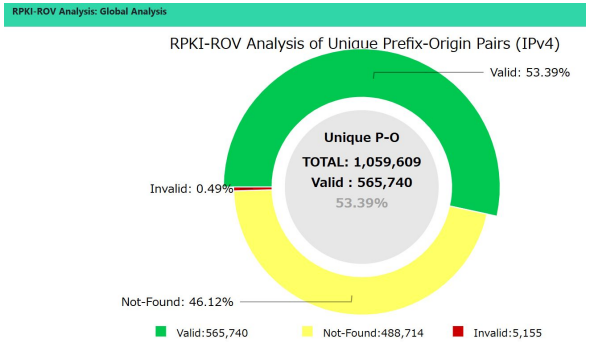
ハンズオンの流れ

1. 接続準備
2. Routinatorの構築・接続
3. ROVと各種パラメータの挙動確認
4. SLURMの設定・挙動確認
5. トラブルシューティング
6. **Tips及びまとめ**

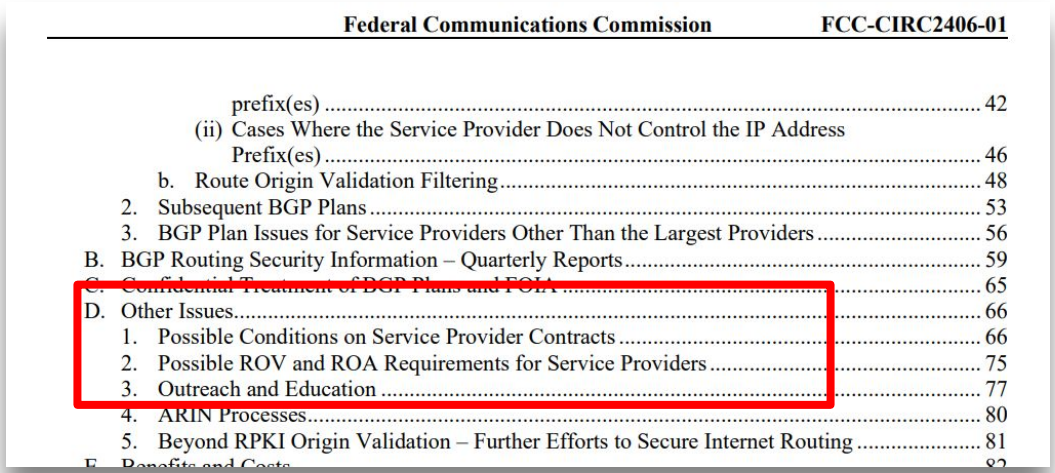
パブリックROAキャッシュサーバ

- JPNIC
  - https://www.nic.ad.jp/ja/rpki/howto-usepubcache.html
- インターネットマルチフィード
  - https://www.mfeed.ad.jp/rpki/tech.html
- BBIX
  - https://www.bbix.net/rpki/tech/

最新のトピックス等



- ROV導入ASの増加
- なぜ？US　FCCのNotice of Proposed Rulemaking（NPRM）にもROVが



Federal Communications Commission **FCC-CIRC2406-01**

参　考

Ubuntu 22.04におけるRoutinatorの構築

Routinatorの導入 (Ubuntu 22.04)

```
sudo apt install -y ca-certificates curl gnupg lsb-release
curl -fsSL https://packages.nlnetlabs.nl/aptkey.asc | sudo gpg --dearmor -o
/usr/share/keyrings/nlnetlabs-archive-keyring.gpg

echo "deb [arch=$(dpkg --print-architecture)
signed-by=/usr/share/keyrings/nlnetlabs-archive-keyring.gpg]
https://packages.nlnetlabs.nl/linux/ubuntu $(lsb_release -cs) main" | sudo tee
/etc/apt/sources.list.d/nlnetlabs.list > /dev/null

sudo apt update && sudo apt install -y routinator
{Routinator confを書く, /etc/routinator/routinator.conf}
```