

RPKIトラブルシューティングハンズオン - ROAキャッシュサーバ編 -

司会・フォロー 岡田雅之

講師・アシスタント 齋藤 脩愉, 石川 嶺, 山本 くるみ, 後藤 汰珠

自己紹介

- 氏名
 - 岡田 雅之
- 所属
 - 長崎県立大学



自己紹介

- 氏名
 - 齋藤 脩愉
- 所属
 - 長崎県立大学



自己紹介

- 氏名
 - 石川 嶺
- 所属
 - 長崎県立大学情報セキュリティ学科
 - 学部3年
- 出身
 - 兵庫県神戸市



自己紹介

- 氏名
 - 山本 くるみ
- 所属
 - 長崎県立大学情報セキュリティ学科
 - 学部3年
- 出身
 - 福岡県北九州市



自己紹介

- 氏名
 - 後藤 汰珠
- 所属
 - 長崎県立大学
- 出身
 - 大分県大分市



ハンズオンの流れ

- RPKIの仕組みと用語説明
- ハンズオン環境の説明
- ROAキャッシュサーバの動作確認
 - 各種オプションの説明
- SLURM機能の利用方法
- 小休憩

ハンズオンの流れ

- トラブルシューティング
 - 監視基盤の導入について
 - VRPが正常に取得できていない場合の対処方法について

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を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。

RPKIの仕組みと用語説明

- ROA (Route Origin Authorization)
 - BGPで使われるAS番号とIPアドレスの組み合わせに対して、それが正しい組み合わせであることを示す電子署名が施されたデータ (<https://www.nic.ad.jp/ja/basics/terms/roa.html>)
- VRP (Validated ROA Payload)
 - ROAキャッシュサーバがROAを検証し、署名が正しいと判断したものをVRPとして出力する
 - VRPはRTRプロトコルを用いてルータに転送され、ROVに利用される

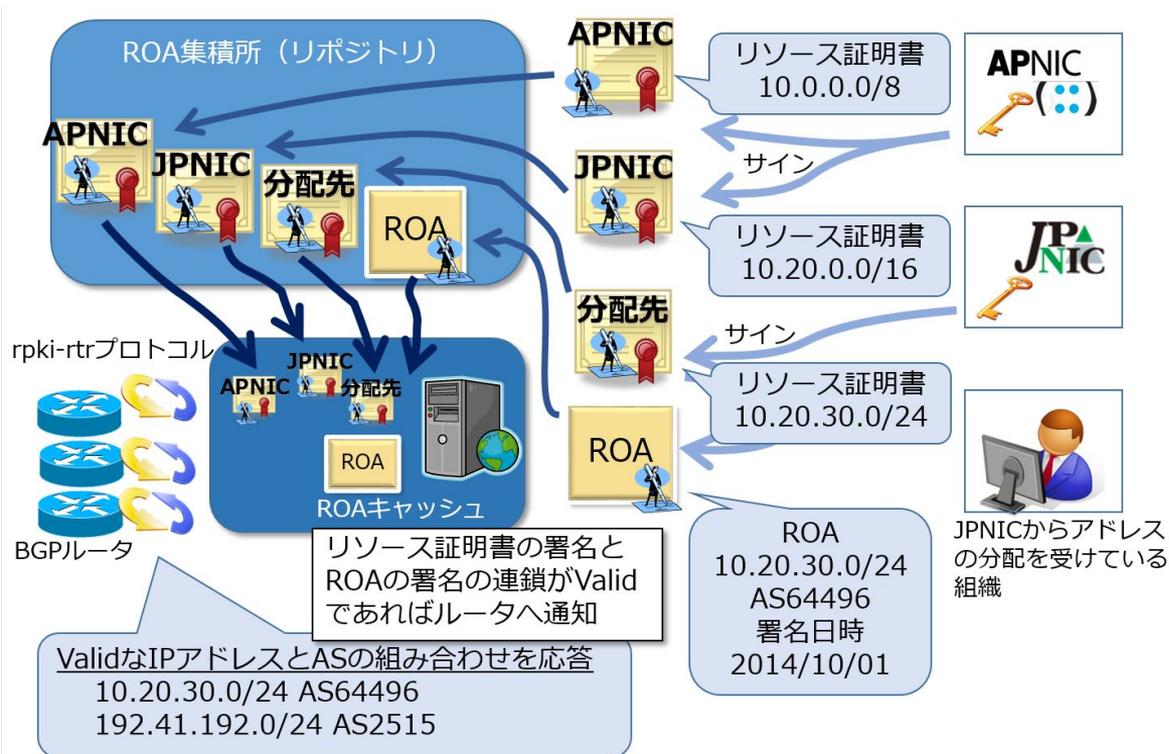
RPKIの仕組みと用語説明

- RP(Relying Party)
 - RPKIの分散データを収集・検証し、キャッシュを構築するソフトウェア (RFC8210)
 - ROAキャッシュサーバとほぼ同義で利用される
- ROV(Route origin validation) = オリジン検証
 - BGPルーターにおいて、経路情報中のIPアドレスとAS番号の組み合わせが正しいかどうかをVRPに基づいて検証出来るようにするための仕組み (<https://www.nic.ad.jp/ja/basics/terms/rov.html>)

RPKIの仕組みと用語説明

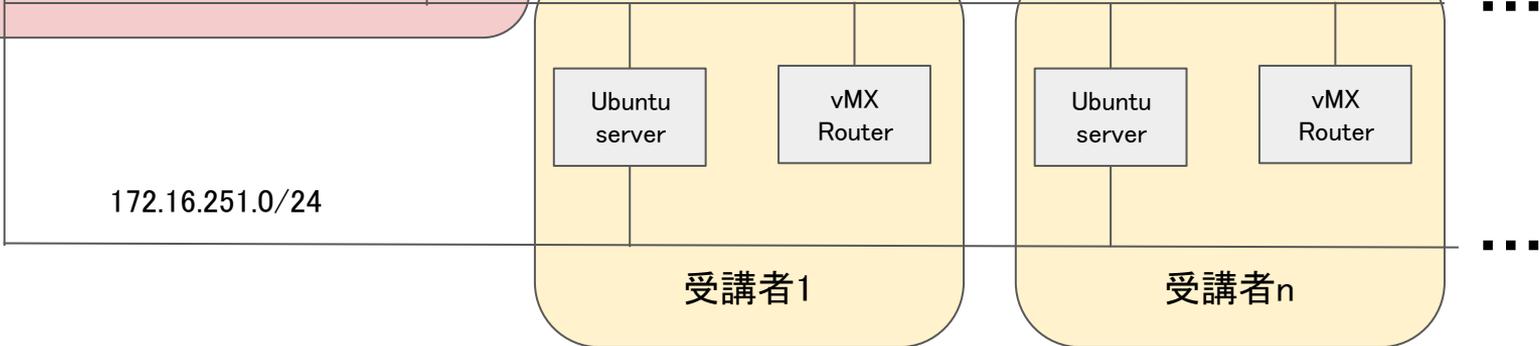
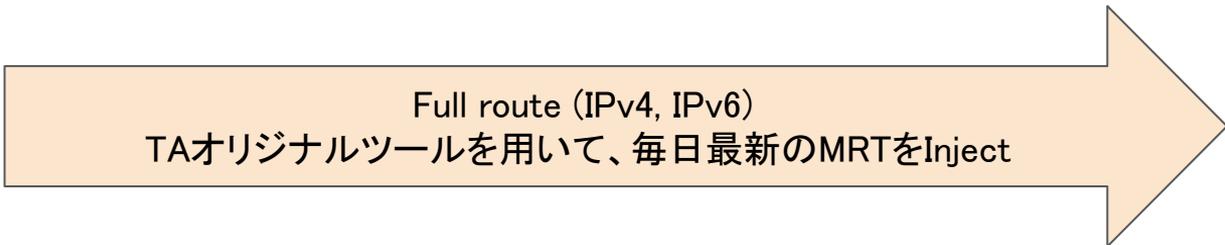
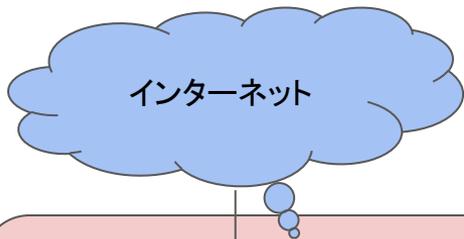
- リポジトリ
 - JPNICが発行するリソース証明書などの電子証明書やROAなどの電子署名が付与されたデータを公開するサーバ
(<https://www.nic.ad.jp/ja/rpki/rpki-terms-conditions.pdf>)
- 認証局
 - リソース証明書を発行する機関

RPKIの仕組みと用語説明



(<https://www.nic.ad.jp/ja/rpki/> , 2025/11/18)

本日の環境



本日の環境



インターネット

AS65100~AS
65103

踏み台

GoBGP
01~04

172.16.250.0/24

運営

172.16.251.0/24

Ubuntu
server

vMX
Router

受講者1

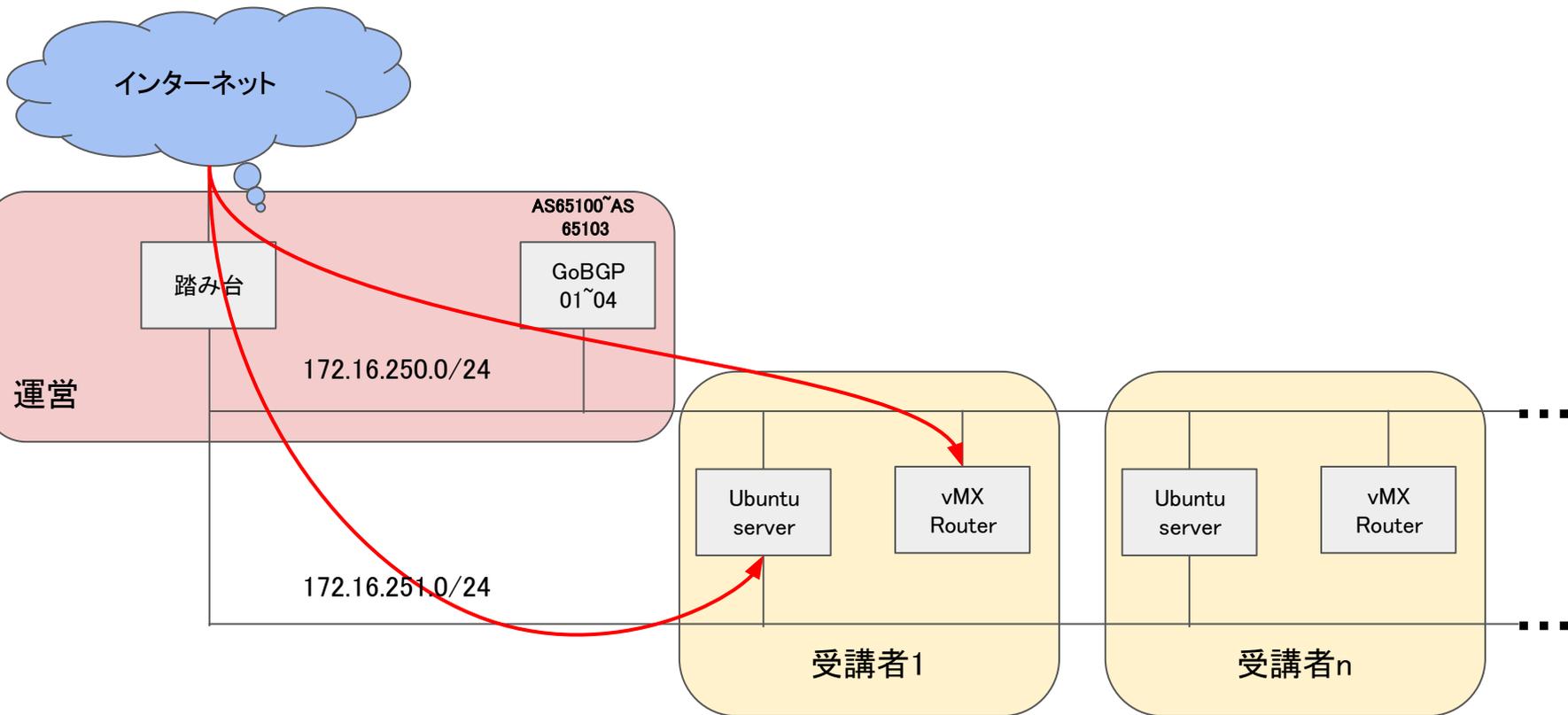
lanセグメント

- Internetとの疎通を持つ
- vMXがGoBGPからfullrouteを取得
- トラブルシューティングでの操作対象

peerセグメント

- Internetとの疎通を持たない
- 踏み台経由でのログイン用
- 基本的に操作を行わない

ハンズオン環境へのログイン



ハンズオン環境へのログイン

- 参加者全員分のログイン情報を以下に用意済み
 - [https://docs.google.com/spreadsheets/d/1Boc4z2yRjD00o0HDyLodEmBvfI3GIq0SvpF3M2_jxrE/edit?usp=sharing]

```
(local PC) ssh [username]@iw25.orumoru.com -p 10022  
Password: [fjn_fhc4QKU1mwz_trw]
```

ハンズオン環境へのログイン

- 踏み台にログインが成功すると以下のような、アクセス先のVMを選択するメニューが表示される

```
Welcome to InternetWeek2025 ROA cache server trouble shooting handson's ssh gateway machine!
```

```
-----MACHINE LIST-----
```

```
Username: user40
```

NAME	IP
server	172.16.250.118
router	172.16.251.120
exit	Do exit

```
Machine name?:
```

ハンズオン環境へのログイン

- 踏み台経由でログインしたい機器名を入力することでログインが可能

```
Welcome to InternetWeek2025 ROA cache server trouble shooting handson's ssh gateway machine!
```

```
-----MACHINE LIST-----
```

```
Username: user40
```

NAME	IP
server	172.16.250.118
router	172.16.251.120
exit	Do exit

```
Machine name?: server
```

```
Machine Password is iw25-rpki
```

```
Connecting to server (172.16.251.118)...
```

```
user@iw25-user28-server:~$
```

ハンズオン環境へのログイン

- このとき、アクセス先VMでPW入力が求められた際は、赤文字の部分を参照して入力する

```
Welcome to InternetWeek2025 ROA cache server trouble shooting handson's ssh gateway machine!
```

```
-----MACHINE LIST-----
```

```
Username: user40
```

NAME	IP
server	172.16.250.118
router	172.16.251.120
exit	Do exit

```
Machine name?: server
```

```
Machine Password is iw25-rpki
```

```
Connecting to server (172.16.251.118)...
```

```
user@iw25-user28-server:~$
```

ハンズオン環境へのログイン

- 戻るときは” exit” を利用(以下は、アクセス先VMでexitを発行している様子)

```
user@iw25-user28-server:~$ exit
logout
Shared connection to 172.16.251.118 closed.
    server 172.16.250.118
    router 172.16.251.120
    exit   Do exit
Machine name?:
```

**みなさまの接続を確認していますので、
少しお待ち下さい**

ROAキャッシュサーバソフトウェアについて

- **rpki-client**

- RRDPもしくはrsyncプロトコルを用いてROA取得し、VRPとして出力を行うソフトウェア
- ルータとのRTRセッション機能は提供されない
- 監視用metric fileの出力が可能
- ROAの収穫を行う度にコマンドを実行する必要がある

- **stayrtr**

- ルータとのRTRセッション機能のみを有したソフトウェア
 - ローカルのVRP(json形式)を参照する

ROAキャッシュサーバソフトウェアについて

- routinator
 - リポジトリからのROA取得機能とRTRセッション機能の両方を持ち合わせたソフトウェア
 - httpで監視用エンドポイントの提供がある
 - serviceとして動作するため、一度起動するだけで定期的にROAの取得が可能

O2 ROAキャッシュサーバハンズオン～RPKI/ROVの普及を目指して～

日時	2023年11月15日 15:00～18:00
会場	オンライン



[プログラム紹介ページ](#)

タイトル	講演者	配布資料 (PDF)
ROAキャッシュサーバハンズオン (敬称略)	岡田 雅之(長崎県立大学)	1.26MB

このページを評価してください

<https://www.nic.ad.jp/ja/materials/iw/2024/proceedings/h3/>

H3 ROAキャッシュサーバハンズオン ～ROVをはじめよう～

日時	2024年11月21日 14:00～17:00
会場	オンライン



[プログラム紹介ページ](#)

タイトル	講演者	配布資料 (PDF)
ROAキャッシュサーバハンズオン ～RPKI/ROVの普及を目指して～	井上 七星(長崎県立大学) 岡田 雅之(長崎県立大学) 後藤 汰珠(長崎県立大学) 齋藤 脩愉(長崎県立大学)	2.21MB

(敬称略)

このページを評価してください

Routinator

ROAキャッシュサーバの動作確認

- 今回は事前にrpki-client & stayrtrをインストール済み
 - インストール方法は過去資料を参照

```
Machine name?: server
Machine Password is iw25-rpki
Connecting to server (172.16.251.118)..

user@iw25-user28-server:~$ which rpki-client
/usr/local/sbin/rpki-client
user@iw25-user28-server:~$ which stayrtr
/usr/local/bin/stayrtr
```

ROAキャッシュサーバの動作確認

- rpki-clientはcronを用いて1時間毎に実行するよう設定済み
 - 普段の運用ではcron等で定期実行する想定
 - 本ハンズオンでは手動実行も併用する

```
user@iw25-user28-server:~$ sudo su
[sudo] password for user:iw25-rpki
root@iw25-user-base:/home/user# crontab -e
```

```
~~~~~
# m h dom mon dow    command
5 0 * * * update
0 * * * * /usr/local/sbin/rpki-client > /var/log/rpki-client.log 2>&1
```

rpki-clientの
標準出力と標準エラー出力を保存

ROAキャッシュサーバの動作確認

- VRPの状態変化を防ぐため、cron設定を無効化
 - 監視基盤閲覧用データの収集のため、cron設定を事前に投入済み

```
user@iw25-user28-server:~$ sudo su
root@iw25-user-base:/home/user# crontab -e
```

```
~~~~~
# m h dom mon dow    command
5 0 * * * update
#0 * * * * /usr/local/sbin/rpki-client > /var/log/rpki-client.log 2>&1
```

ROAキャッシュサーバの動作確認

- rpki-clientが出力するVRPの保存先を確認する

```
root@iw25-user-base:/home/user# exit
exit
user@iw25-user28-server:~$ ls -l /usr/local/var/db/rpki-client/
total 201672
-rw-r--r-- 1 _rpki-client _rpki-client 31063730 Nov 18 17:02 bird
-rw-r--r-- 1 _rpki-client _rpki-client 33670651 Nov 18 17:02 csv
-rw-r--r-- 1 _rpki-client _rpki-client 80020003 Nov 18 17:02 json
-rw-r--r-- 1 _rpki-client _rpki-client 822739 Nov 18 17:02 metrics
-rw-r--r-- 1 _rpki-client _rpki-client 41421285 Nov 18 17:02 openbgpd
-rw-r--r-- 1 _rpki-client _rpki-client 19504965 Nov 18 17:02 rpki.ccr
```

最終更新日時を確認し、
cronが適切に動作していたか確認

ROAキャッシュサーバの動作確認

- rpki-clientを手動実行する
 - 5～10分程度待機。
 - 実行時logを確認したい方は” /var/log/rpki-client_test.log” を参照

```
user@iw25-user28-server:~$ sudo rpki-client -mj 2>&1 | sudo tee  
/var/log/rpki-client_test.log
```

改行ではなく、半角スペース！

ROAキャッシュサーバの動作確認

- rpki-clientが出力するVRPの保存先を再度確認する

```
user@iw25-user28-server:~$ ls -l /usr/local/var/db/rpki-client/  
total 201672  
-rw-r--r-- 1 _rpki-client _rpki-client 31063730 Nov 18 17:02 bird  
-rw-r--r-- 1 _rpki-client _rpki-client 33670651 Nov 18 17:02 csv  
-rw-r--r-- 1 _rpki-client _rpki-client 80022441 Nov 18 17:37 json  
-rw-r--r-- 1 _rpki-client _rpki-client 822752 Nov 18 17:37 metrics  
-rw-r--r-- 1 _rpki-client _rpki-client 41421285 Nov 18 17:02 openbgpd  
-rw-r--r-- 1 _rpki-client _rpki-client 19505148 Nov 18 17:37 rpki.ccr
```

最終更新日時を確認し、
手動実行が適切に動作していたか確認

ROAキャッシュサーバの動作確認

- 検証済みの内容をのぞいてみよう

```
user@iw25-user27-server:~$ cat /usr/local/var/db/rpki-client/csv | more
ASN,IP Prefix,Max Length,Trust Anchor,Expires
AS13335,1.0.0.0/24,24,apnic,1764080227
AS18144,1.0.64.0/18,18,apnic,1764125416
AS13335,1.1.1.0/24,24,apnic,1764080227
AS4134,1.1.4.0/22,22,apnic,1764122659
AS4134,1.1.16.0/20,20,apnic,1764122659
AS4134,1.1.32.0/19,19,apnic,1764122659
AS4134,1.2.9.0/24,24,apnic,1764122659
AS4134,1.2.10.0/24,24,apnic,1764122659
AS4134,1.2.11.0/24,24,apnic,1764122659
```

ROAキャッシュサーバの動作確認

- 検証済みの内容をのぞいてみよう

```
user@iw25-user27-server:~$ cat /usr/local/var/db/rpki-client/csv | more
ASN,IP Prefix,Max Length,Trust Anchor,Expires
AS13335,1.0.0.0/24,24,apnic,1764080227
AS18144,1.0.64.0/18,18,apnic,1764125416
AS13335,1.1.1.0/24,24,apnic,1764080227
AS4134,1.1.4.0/22,22,apnic,1764122659
AS4134,1.1.16.0/20,20,apnic,1764122659
AS4134,1.1.32.0/19,19,apnic,1764122659
AS4134,1.2.9.0/24,24,apnic,1764122659
AS4134,1.2.10.0/24,24,apnic,1764122659
AS4134,1.2.11.0/24,24,apnic,1764122659
```

ROAキャッシュサーバの動作確認

- stayrtrのservice確認
 - stayrtrは事前にserviceとして動くよう設定変更&起動済み

```
user@iw25-user28-server:~$ sudo cat /etc/systemd/system/stayrtr.service
[Unit]
Description = Stay-rtr

[Service]
ExecStart = /usr/local/bin/stayrtr -bind :3323 -cache
/usr/local/var/db/rpki-client/json -slurm /usr/local/etc/slurm.json
ExecStop = /bin/kill -WINCH ${MAINPID}
Restart = always
Type = simple

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

- RTRはport :3323 でlisten
- VRPIは /usr/local/var/db/rpki-client/jsonを指定
- slurmは /usr/local/etc/slurm.jsonを指定

routerの動作確認

- routerへログインする

```
user@iw25-user28-server:~$ exit
logout
Connection to 172.16.251.82 closed.
    server 172.16.251.82
    router 172.16.250.84
    exit    Do exit

Machine name?: router
Machine Password is iw25-rpki
Connecting to router (172.16.250.84)...

Last login: Fri Nov 15 03:55:07 2024 from 172.27.1.227
--- JUNOS 23.1R1.8 Kernel 64-bit  JNPR-12.1-20230307.3e7c4b6_buil
user@iw25-user28-vmx>
```

routerの動作確認

- BGPのセッションを確認する

```
user@iw25-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	1043832	1043831	0	0	0	0	0
--------	---------	---------	---	---	---	---	---

inet6.0	236354	236353	0	0	0	0	0
---------	--------	--------	---	---	---	---	---

Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last	Up/Dwn
------	----	-------	--------	------	-------	------	--------

```
State|#Active/Received/Accepted/Damped...
```

172.16.250.250	65102	147181	36	0	0	15:56	Establ
----------------	-------	--------	----	---	---	-------	--------

```
inet.0: 1043831/1043832/1043832/0
```

2400:5320:ff00:16fa::250	65102	236388	36	0	0	15:46	
--------------------------	-------	--------	----	---	---	-------	--

```
Establ
```

```
inet6.0: 236353/236354/236354/0
```

routerの動作確認

- ROAキャッシュサーバとの接続設定を投入する

```
user@iw25-user28-vmx> configure
Entering configuration mode

[edit]
user@iw25-vMX-base#set routing-options validation group rpki-client session [ROA cacheIP]
port 3323
user@iw25-vMX-base# show | compare
[edit routing-options validation group rpki-client]
+   session [ROA cacheIP] {
+     port 3323;
+   }
user@iw25-vMX-base# commit
commit complete

[edit]
user@iw25-vMX-base# exit
```

172.16.250.0/24セグのIP

routerの動作確認

- ROAキャッシュサーバとのセッションを確認する

```
user@iw25-user28-vmx> show validation session
Session                               State   Flaps    Uptime #IPv4/IPv6 records
172.16.251.118                        Up      2        00:50:58 609951/161270
```

- ROAキャッシュサーバとの接続状況
- uptime
- v4, v6のVRP総数

が確認可能

routerの動作確認

- ROVのポリシー設定を行う①

```
user@iw25-user28-vmx> configure
Entering configuration mode

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
```

routerの動作確認

- ROVのポリシー設定を行う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
```

- 受け取った経路がValidである場合
 - 経路をacceptし、LP200を設定
- 受け取った経路がinvalidである場合
 - 経路をacceptし、LP50を設定
 - 通常の運用ではrejectを利用する
- 受け取った経路がunknownである場合
 - 経路をacceptし、LP100を設定

routerの動作確認

- ROVの結果を確認する

```
user@iw25-user28-vmx> show route validation-state valid detail
user@iw25-user28-vmx> show route validation-state invalid detail
user@iw25-user28-vmx> show route validation-state unknown detail
```

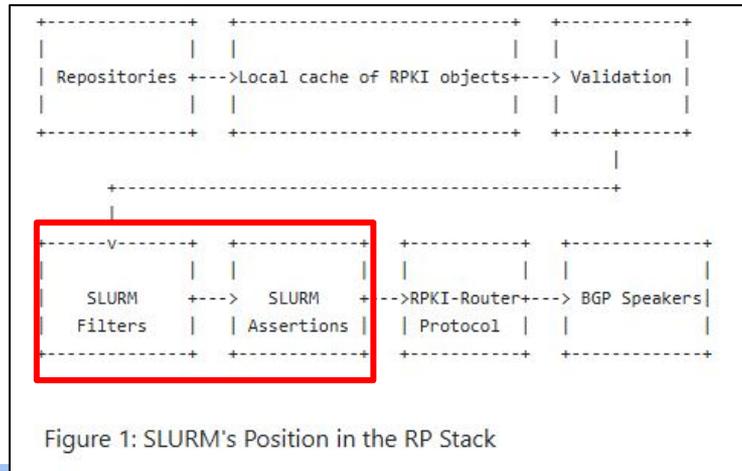
- valid
 - LP200, accept
 - unknown
 - LP100, accept
 - invalid
 - LP50, accept
- がそれぞれ設定されていることが確認できる

*slurm*編

進みに不安のある方は、挙手ください

SLURMとは

- ROA キャッシュサーバに指定された ROA が存在するかのよう
に VRP を生成する仕組みのこと
(【[RPKI ガイドライン案](#)】より)
- SLURMは生成済みのVRPに対してフィルタのような働きをして
結果を書き換えることができる



SLURMファイル記入例

- **validationOutputFilters**
 - 記載された内容と合致するVRPがルータに転送されなくなる
 - “prefix”
 - 該当ネットワークを非転送
 - 包括する場合も同様
 - “asn”
 - 該当ASNを非転送
 - “prefix” と “asn” を同時に使用する場合はどちらも該当するVRPがルータに転送されなくなる
 - 結果として記載対象はunknown判定となる

```
##slurmファイル 例
{
  "slurmVersion": 1,
  "validationOutputFilters": {
    "prefixFilters": [
      {
        "prefix": "198.51.100.0/24",
        "asn": 64497,
      }
    ],
    "bgpsecFilters": []
  },
  "locallyAddedAssertions": {
    "prefixAssertions": [],
    "bgpsecAssertions": []
  }
}
```

SLURMファイル記入例

- **locallyAddedAssertions**

- 記載された内容のVRPがVRPsに追加され、ルータに転送されるようになる
- “prefix” ”asn”
 - 必須パラメータ。追加するVRPのprefixとasnを指定
- “maxPrefixLength”
 - 追加するVRPの最大長を指定
 -
- 結果として記載対象はvalidとなる

```
##slurmファイル 例
```

```
{  
  "slurmVersion": 1,  
  "validationOutputFilters": {  
    "prefixFilters": [],  
    "bgpsecFilters": []  
  },  
  "locallyAddedAssertions": {  
    "prefixAssertions": [  
      {  
        "prefix": "198.51.100.0/24",  
        "asn": 64497,  
        "maxPrefixLength": 26,  
      }  
    ],  
    "bgpsecAssertions": []  
  }  
}
```

slurmを適用し、ROVの結果を書き換える

- 直接ファイルを編集する方法もあるが、視覚的にわかりやすいオリジナルソフトウェアである“[control-slurm.py](#)”を利用する

現在のslurmの中身を確認する

- ROAキャッシュサーバからslurmの中身を確認する

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py --view-plane  
Path: /usr/local/etc/slurm.json
```

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

slurmの実体は”usr/local/etc/slurm.json”にあること、slurmの中身が空なことが確認できる

slurm追加前のROV結果を確認する_valid

```
user@iw25-user28-vmx> show route validation-state valid
```

```
inet.0: 1042771 destinations, 1042771 routes (1042770 active, 0 holddown, 1 hidden)  
+ = Active Route, - = Last Active, * = Both
```

```
1.0.0.0/24
```

```
*[BGP/170] 00:38:22, localpref 200, from 172.16.250.249  
AS path: 65103 2497 13335 I, validation-state: valid  
> to 172.16.250.254 via fxp0.0
```

```
1.0.64.0/18
```

```
*[BGP/170] 00:38:22, localpref 200, from 172.16.250.249  
AS path: 65103 2497 2519 7610 I, validation-state: valid  
> to 172.16.250.254 via fxp0.0
```

```
1.1.1.0/24
```

```
*[BGP/170] 00:38:22, localpref 200, from 172.16.250.249  
AS path: 65103 2497 13335 I, validation-state: valid  
> to 172.16.250.254 via fxp0.0
```

後々” validationOutputFilters” に追加予定のためprefixを控えておく

slurm追加前のROV結果を確認する_Invalid

```
user@iw25-user28-vmx> show route validation-state invalid
```

```
inet.0: 1042771 destinations, 1042771 routes (1042770 active, 0 holddown, 1 hidden)  
+ = Active Route, - = Last Active, * = Both
```

```
1.39.81.0/24
```

```
*[BGP/170] 00:42:14, localpref 50, from 172.16.250.249  
AS path: 65103 4777 2516 6762 9498 55410 38266 ?, validation-state:
```

```
invalid
```

```
> to 172.16.250.254 via fxp0.0
```

```
1.44.160.0/23
```

```
*[BGP/170] 00:42:14, localpref 50, from 172.16.250.249  
AS path: 65103 4777 2516 7410 38266 ?, validation-state:  
> to 172.16.250.254 via fxp0.0
```

後々”locallyAddedAssertions”に追加予定のためprefixとorigin-asを控えておく

validationOutputFiltersの追加(Valid経路の無効化)

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py -a
[sudo] password for user:
1: prefixFilters / 2: prefixAssertions?: 1
ASN?:
Prefix?: 1.0.0.0/24
comment?:

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

locallyAddedAssertionsの追加(Invalid経路のValid化)

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py -a
1: prefixFilters / 2: prefixAssertions?: 2
ASN?: 38266
Prefix?: 1.39.81.0/24
MaxPrefixLength /?: 26
comment?:
```

Are you sure you want to add the following information?

```
-----
Type: locallyAddedAssertions.prefixAssertions
ASN: 38266
Prefix: 1.39.81.0/24
MaxPrefixLength: /26
y/n: y
```

slurmが正常に追加されているか確認する

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py --view-plane
```

```
Path: /usr/local/etc/slurm.json
```

```
{
  "slurmVersion": 1,
  "validationOutputFilters": {
    "prefixFilters": [
      {
        "prefix": "1.0.0.0/24"
      }
    ],
    "bgpsecFilters": []
  },
  "locallyAddedAssertions": {
    "prefixAssertions": [
      {
        "asn": 38266,
        "prefix": "1.39.81.0/24",
        "maxPrefixLength": 26
      }
    ],
    "bgpsecAssertions": []
  }
}
```

stayrtrをrestartする

```
user@iw25-user28-server:~$ sudo systemctl restart stayrtr
```

```
user@iw25-user28-server:~$ systemctl status stayrtr
```

```
● stayrtr.service - Stay-rtr
```

```
Loaded: loaded (/etc/systemd/system/stayrtr.service; enabled; preset: enab>
```

```
Active: active (running) since Wed 2025-11-19 11:20:33 JST; 7s ago
```

```
Main PID: 2807 (stayrtr)
```

```
Tasks: 8 (limit: 9432)
```

```
Memory: 540.0M (peak: 540.7M)
```

```
CPU: 4.868s
```

```
CGroup: /system.slice/stayrtr.service
```

```
└─2807 /usr/local/bin/stayrtr
```

slurmの変更を確実に適用するためにrestartを行う
stayrtrは定期的にVRPとslurmを読み込むため、
restartしなくても15分ほどで適用される

slurm追加後のROV結果を確認する_Valid

```
user@iw25-user28-vmx> show route 1.0.0.0/24 detail
```

```
inet.0: 1042771 destinations, 1042771 routes (1042770 active, 0 holddown, 1 hidden)
```

```
1.0.0.0/24 (1 entry, 1 announced)
```

```
*BGP Preference: 170/-101
Next hop type: Indirect, Next hop index: 0
Address: 0x38fd818e52dc
Next-hop reference count: 2023998
Kernel Table Id: 0
Source: 172.16.250.249
Next hop type: Router, Next hop index: 337
Next hop: 172.16.250.254 via fxp0.0, selected
Session Id: 0
Protocol next hop: 202.249.2.169
Indirect next hop: 0x38fd819b5408 363 INH Session ID: 0
Indirect next hop: INH non-key opaque: 0x0 INH key opaque: 0x0
State: <Active Ext>
Local AS: 64551 Peer AS: 65103
Age: 1:07:08 Metric2: 0
Validation State: unknown
Task: BGP_65103.172.16.250.249
Announcement bits (2): 0-KRT 3-Resolve tree 1
AS path: 65103 2497 13335 I
Aggregator: 13335 10.34.7.24
Accepted
Localpref: 100
Router ID: 172.16.250.249
Thread: junos-main
```

先ほどslurmに記載したrecordが適用され、Valid経路だったものがunknown経路に変化したことが確認できる

slurm追加後のROV結果を確認する_Invalid

```
user@iw25-user28-vmx> show route 1.39.81.0/24 detail
```

```
inet.0: 1042771 destinations, 1042771 routes (1042770 active, 0 holddown, 1 hidden)
```

```
1.39.81.0/24 (1 entry, 1 announced)
```

```
*BGP Preference: 170/-201
```

```
Next hop type: Indirect, Next hop index: 0
```

```
Address: 0x38fd951252fc
```

```
Next-hop reference count: 2374
```

```
Kernel Table Id: 0
```

```
Source: 172.16.250.249
```

```
Next hop type: Router, Next hop index: 337
```

```
Next hop: 172.16.250.254 via fxp0.0, selected
```

```
Session Id: 0
```

```
Protocol next hop: 202.249.2.110
```

```
Indirect next hop: 0x38fd819faa08 370 INH Session ID: 0
```

```
Indirect next hop: INH non-key opaque: 0x0 INH key opaque: 0x0
```

```
State: <Active Ext>
```

```
Local AS: 64551 Peer AS: 65103
```

```
Age: 4:20 Metric2: 0
```

```
Validation State: valid
```

```
Task: BGP_65103.172.16.250.249
```

```
Announcement bits (2): 0-KRT 3-Resolve tree 1
```

```
AS path: 65103 4777 2516 6762 9498 55410 38266 ?
```

```
Aggregator: 65010 10.191.81.98
```

```
Accepted
```

```
Localpref: 200
```

```
Router ID: 172.16.250.249
```

```
Thread: junos-main
```

先ほどslurmに記載したrecordが適用され、Invalid経路だったものがvalid経路に変化したことが確認できる

validationOutputFiltersの削除(slurmの無効化)

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py -d
1: prefixFilters / 2: prefixAssertions?: 1
ASN?:
Prefix?: 1.0.0.0/24

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

y/n: y
```

locallyAddedAssertionsの削除(slurmの無効化)

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py -d
1: prefixFilters / 2: prefixAssertions?: 2
ASN?: 38266
Prefix?: 1.39.81.0/24
MaxPrefixLength /?: 26

Are you sure you want to delete the following information?
-----
asn:38266, prefix:1.39.81.0/24, maxPrefixLength:26,

y/n: y
```

slurmが削除されているか確認する

```
user@iw25-user28-server:~$ sudo python3 control-slurm.py --view-plane  
Path: /usr/local/etc/slurm.json
```

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

stayrtrをrestartする

```
user@iw25-user28-server:~$ sudo systemctl restart stayrtr
```

router側のsessionをclearする

```
user@iw25-user28-vmx> clear validation session  
Cleared 1 sessions
```

```
user@iw25-user28-vmx> clear validation database  
Clearing database
```

slurm削除後のROV結果を確認する_Valid

```
user@iw25-user28-vmx> show route 1.0.0.0/24 detail
```

```
inet.0: 1042773 destinations, 1042774 routes (1042772 active, 0 holddown, 1 hidden)
```

```
1.0.0.0/24 (1 entry, 1 announced)
```

```
*BGP Preference: 170/-201
Next hop type: Indirect, Next hop index: 0
Address: 0x38fd818e52dc
Next-hop reference count: 2023999
Kernel Table Id: 0
Source: 172.16.250.249
Next hop type: Router, Next hop index: 337
Next hop: 172.16.250.254 via fxp0.0, selected
Session Id: 0
Protocol next hop: 202.249.2.169
Indirect next hop: 0x38fd819b5408 363 INH Session ID: 0
Indirect next hop: INH non-key opaque: 0x0 INH key opaque: 0x0
State: <Active Ext>
Local AS: 64551 Peer AS: 65103
Age: 19:46 Metric2: 0
Validation State: valid
Task: BGP_65103.172.16.250.249
Announcement bits (2): 0-KRT 3-Resolve tree 1
AS path: 65103 2497 13335 I
Aggregator: 13335 10.34.7.24
Accepted
Localpref: 200
Router ID: 172.16.250.249
Thread: junos-main
```

先ほどslurmに記載したrecordが無効化され
Valid経路に戻ったことが確認できる

slurm削除後のROV結果を確認する_Invalid

```
user@iw25-user28-vmx> show route 1.39.81.0/24 detail
```

```
inet.0: 1042773 destinations, 1042774 routes (1042772 active, 0 holddown, 1 hidden)
```

```
1.39.81.0/24 (1 entry, 1 announced)
```

```
*BGP Preference: 170/-51
```

```
Next hop type: Indirect, Next hop index: 0
```

```
Address: 0x38fd951252fc
```

```
Next-hop reference count: 2375
```

```
Kernel Table Id: 0
```

```
Source: 172.16.250.249
```

```
Next hop type: Router, Next hop index: 337
```

```
Next hop: 172.16.250.254 via fxp0.0, selected
```

```
Session Id: 0
```

```
Protocol next hop: 202.249.2.110
```

```
Indirect next hop: 0x38fd819faa08 370 INH Session ID: 0
```

```
Indirect next hop: INH non-key opaque: 0x0 INH key opaque: 0x0
```

```
State: <Active Ext>
```

```
Local AS: 64551 Peer AS: 65103
```

```
Age: 11:32 Metric2: 0
```

```
Validation State: invalid Reason: Max-Length mismatch
```

```
Task: BGP_65103.172.16.250.249
```

```
Announcement bits (2): 0-KRT 3-Resolve tree 1
```

```
AS path: 65103 4777 2516 6762 9498 55410 38266 ?
```

```
Aggregator: 65010 10.191.81.98
```

```
Accepted
```

```
Localpref: 50
```

```
Router ID: 172.16.250.249
```

```
Thread: junos-main
```

先ほどslurmに記載したrecordが無効化され
Invalid経路に戻ったことが確認できる

Break time

進みに不安のある方は、挙手ください

トラブルシューティング

- トラブルシューティング用監視基盤の確認
- VRPが正常に取得できない場合のトラブルシューティング方法について
- トラブルシューティング実践

監視基盤について

- トラブルシューティングを行う上で視覚的にROAキャッシュサーバの状態を確認できるように監視基盤を導入済み
 - grafana
 - 取得したデータのグラフ化
 - prometheus
 - 取得したデータのデータベースへの保存
 - node-exporter
 - prometheusがmetrics fileを参照するためのサービス公開

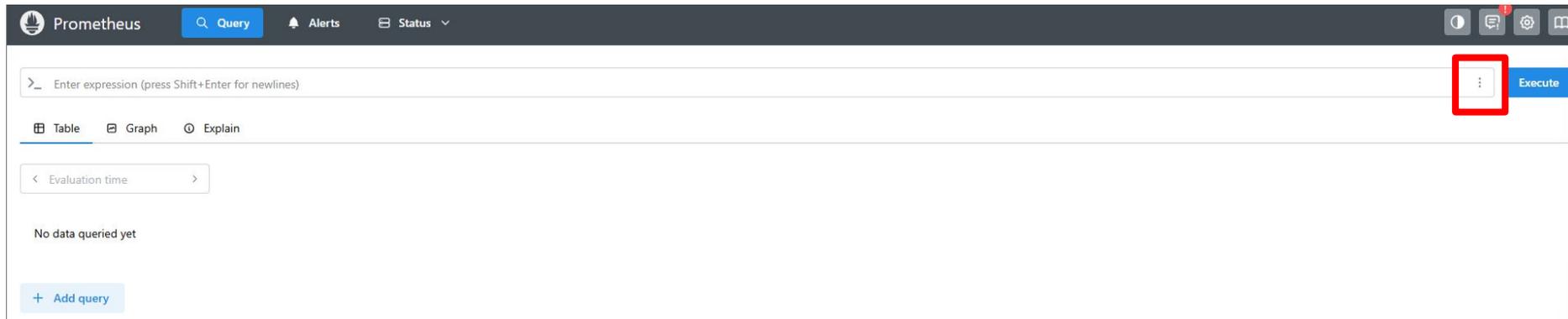
監視基盤にログインを行う

- ローカルのブラウザから以下prometheusのURLにアクセスする
 - <https://iw25.orumoru.com/userXX/prometheus>
 - アドレスリストに記載されたuserと踏み台passwordを入力



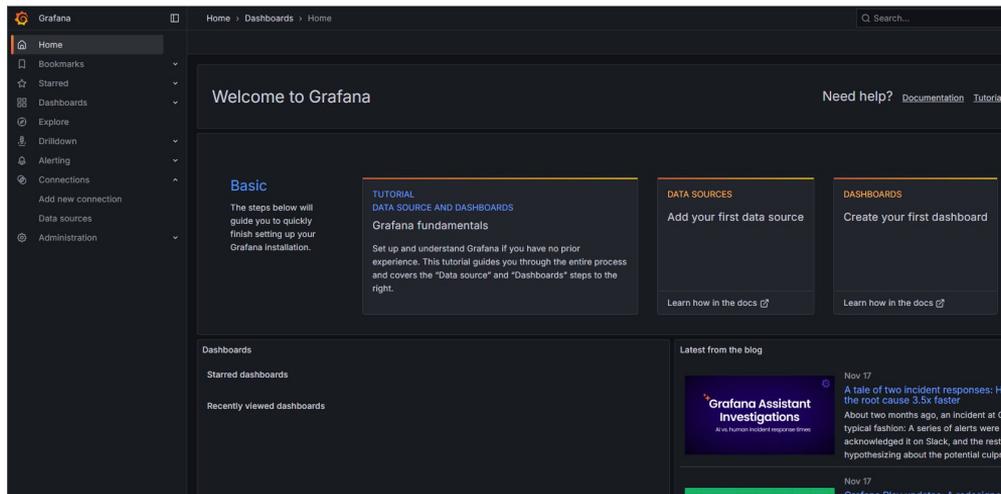
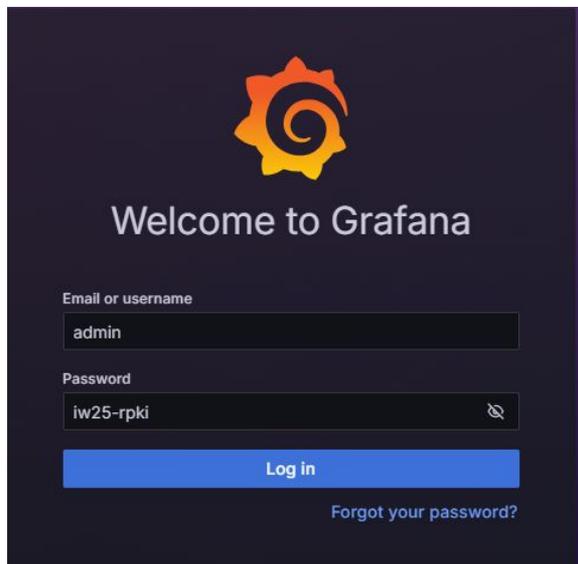
監視基盤にログインを行う

- ローカルのブラウザからprometheusにログインする
 - 図のアイコンから「Explore metrics」を選択することで監視データのmetricsを確認可能



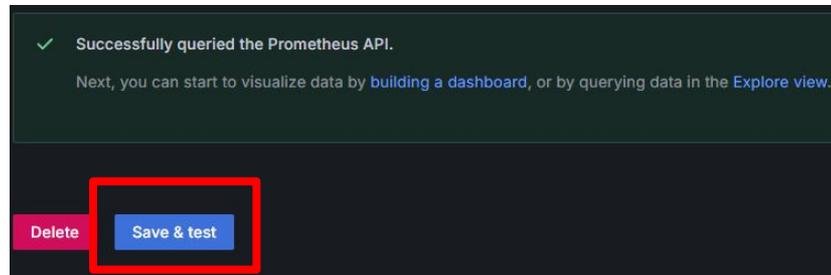
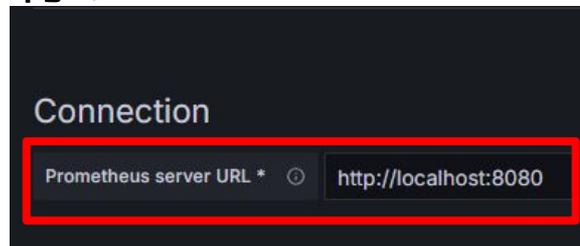
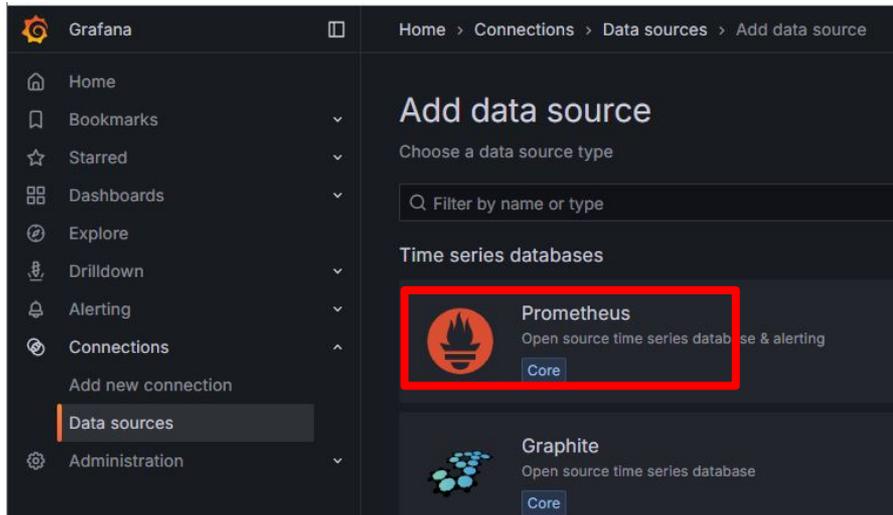
監視基盤にログインを行う

- ローカルのブラウザから以下grafanaのURLにアクセスする
 - <https://iw25.orumoru.com/userXX/grafana>
 - admin:iw25-rpkiでgrafanaにログイン可能



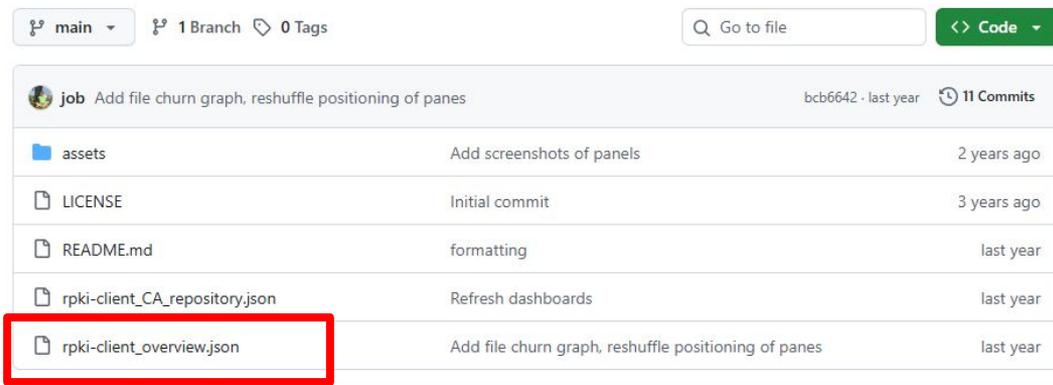
監視基盤の構築を行う

- grafanaに設定を追加し、prometheusからデータを取得し、グラフの作成を行う
- grafana上でdata sourceの指定を行う



監視基盤の構築を行う

- 以下URLからrpki-client公式が公開しているgrafanaのダッシュボードをコピーする
 - <https://github.com/rpki-client/rpki-client-grafana>



File Name	Description	Last Commit
assets	Add screenshots of panels	2 years ago
LICENSE	Initial commit	3 years ago
README.md	formatting	last year
rpki-client_CA_repository.json	Refresh dashboards	last year
rpki-client_overview.json	Add file churn graph, reshuffle positioning of panes	last year

rpki-client-grafana / rpki-client_overview.json



rpki-client-grafana / rpki-client_overview.json

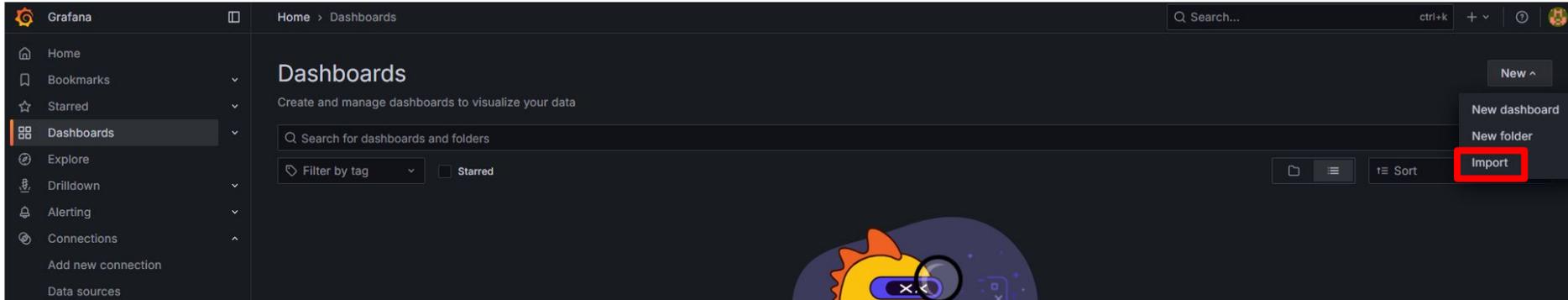
job Add file churn graph, reshuffle positioning of panes bcb6642 · last year History

Code Blame 1726 lines (1726 loc) · 41.3 KB

Raw

監視基盤の構築を行う

- grafanaのダッシュボードページから新規ダッシュボードのインポートを行う



監視基盤の構築を行う

- grafanaのダッシュボードページから新規ダッシュボードのインポートを行う

Import dashboard

Import dashboard from file or Grafana.com

Upload dashboard JSON file

Drag and drop here or click to browse

Accepted file types: .json, .txt

Find and import dashboards for common applications at grafana.com/dashboards

Grafana.com dashboard URL or ID

Import via dashboard JSON model

```
{
  "title": "Example - Repeating Dictionary variables",
  "uid": "_0HnEoN4z",
  "panels": [...],
  ...
}
```

Import dashboard

Import dashboard from file or Grafana.com

Options

Name

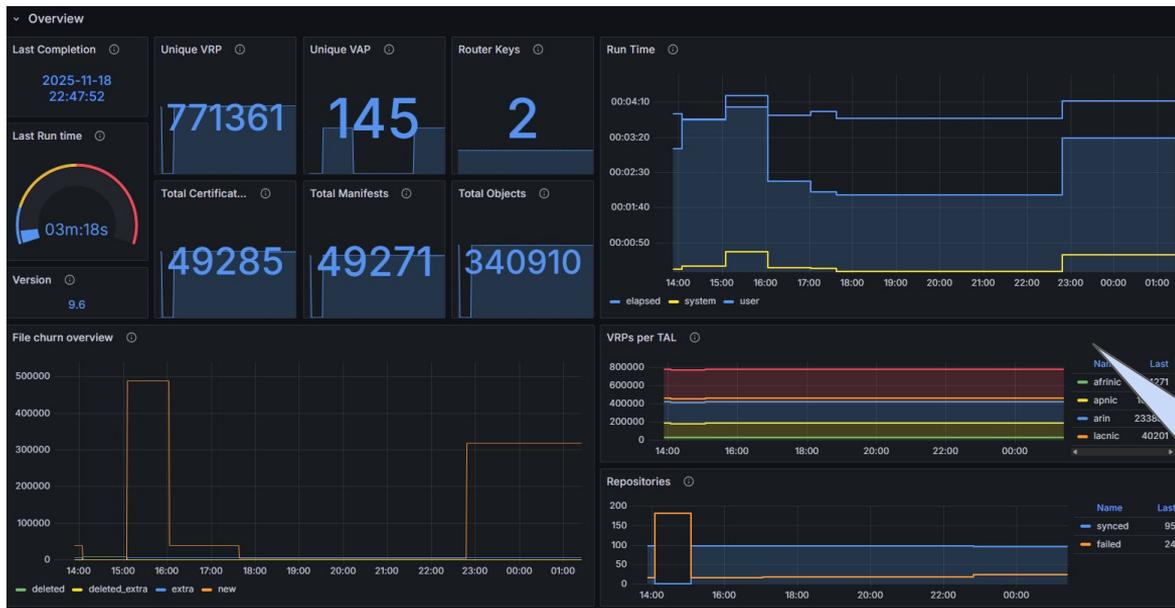
Folder

Unique identifier (UID)

The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

監視基盤の構築を行う

- インポートしたダッシュボードからrpki-clientの実行時様々な監視情報を確認可能



余裕のある方はもう一つのダッシュボードもインポートしてみてください

ダッシュボードについて

- rpki-client実行時の基本情報
 - rpki-clientの最終実行時間
 - rpki-client実行時のrun-time
 - rpki-clientのversion
 - etc
- rpki-client実行時のVRP情報
 - TAL毎のVRP総数
 - VRP数の変動量
 - リポジトリとの接続数
 - etc

監視基盤の導入について

- 監視用の元データには手を加えておらず、rpki-clientから出力される内容(metrics)をそのまま利用している
- prometheus, node-exporterの設定に関しても特別なことはしておらず、通常の監視基盤と同様の手順で構築可能
- ROAキャッシュサーバだけ動かしてRPKI関連の情報収集を行う用途としても活用可能

トラブルシューティング

進みに不安のある方は、挙手ください

トラブルシューティングについて

- 「VRPが正常に取得できない状態」の原因は多岐にわたる

単純な設定ミス

設定に問題はないが自身のネットワークに
問題があるパターン

設定も自身のネットワークにも
問題がないパターン

その他のパターン

トラブルシューティングについて

- その中でもパターン青とパターン黄色について触れる

設定に問題はないが自身のネットワークに
問題があるパターン

設定も自身のネットワークにも
問題がないパターン

設定に問題はないが自身のネットワークに問題があるパターン

- ルータとROAキャッシュサーバ間の通信に問題はないか
 - 間に別の機器が挟まっている場合、ルーティングに問題がある可能性有
 - nat, napt, FWによるフィルタ
- ROAキャッシュサーバのUPLINKに問題はないか
 - 間の通信経路に問題がなくてもROAキャッシュサーバのUPLINKに問題がある場合、VRPの更新が行われない

ルータとROAキャッシュサーバ間の通信に問題がないか

- ping
 - routerとROAキャッシュサーバに対して疎通性の確認を行う
 - 環境によってROAキャッシュサーバから戻りの経路がないような場合もある
- traceroute
 - RPまでの経路が正常か確認を行う

ルータとROAキャッシュサーバ間の通信に問題がないか

- state
 - RPとのセッション状態を確認
- session uptime
 - セッションが初めて上がってから現在まで経過した時間
- Last PDU received
 - RPから最後にデータを送られた時間が記載

```
user@iw25-user28-vmx> show validation
session detail
Session 172.16.251.118, State: up, Session
index: 3
  Group: rpki-client, Preference: 100
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 4
  Serial (Incremental Update): 6
  Session flaps: 18
  Session uptime: 00:49:40
  Last PDU received: 00:02:38
  IPv4 prefix count: 604955
  IPv6 prefix count: 160835
```

ルータとROAキャッシュサーバ間の通信に問題がないか

- ssコマンドを用いてstayrtrがlistenしているか確認を行う

```
user@iw25-user28-server:~$ ss -antu | grep 3323
tcp    LISTEN 0      4096          *:3323          *:*
```

tcp ESTAB 0 0 [::ffff:172.16.251.118]:3323 [::ffff:172.16.251.120]:62148

ルータとROAキャッシュサーバ間の通信に問題がないか

- ROAキャッシュサーバとの接続に問題がない場合stateは「up」「Ex-full」となる

```
user@iw25-user28-vmx> show validation session detail
Session 172.16.251.118, State: up, Session index: 3
  Group: rpki-client, Preference: 100
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 4
  Serial (Incremental Update): 6
    Session flaps: 18
    Session uptime: 00:49:40
  Last PDU received: 00:02:38
  IPv4 prefix count: 604955
  IPv6 prefix count: 160835
```

ROAキャッシュサーバのUPLINKに問題はないか

- ROAキャッシュサーバのUPLINKに問題がある場合
router, ROAキャッシュサーバ上でVRPの更新が行われずVRP総数が不一致になることや、0になってしまう場合がある

ROAキャッシュサーバのUPLINKに問題はないか

- ROAキャッシュサーバ上で一時的にUPLINKを削除する

```
user@iw25-user28-server:~$ sudo apt install net-tools
user@iw25-user28-server:~$ sudo ip route del default via 172.16.250.254 dev lan
user@iw25-user28-server:~$ sudo ip -6 route del default via 2400:5320:ff00:16fa::ffff dev
lan
user@iw25-user28-server:~$ route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
172.16.250.0     0.0.0.0         255.255.255.0   U        0      0      0 lan
172.16.251.0     0.0.0.0         255.255.255.0   U        0      0      0 peer
172.16.251.0     172.16.251.254 255.255.255.0   UG       0      0      0 peer
user@iw25-user28-server:~$ ping 1.1.1.1
ping: connect: Network is unreachable
```

ROAキャッシュサーバのUPLINKに問題はないか

- ROAキャッシュサーバ上でROA cacheを削除する

```
user@iw25-user28-server:~$sudo rm -r /usr/local/var/cache/rpki-client/*
```

cacheが残っていると正しく「UPLINKがない場合」の再現ができないため

ROAキャッシュサーバのUPLINKに問題はないか

- UPLINKを削除した状態でrpk-clientを実行する
 - -s 300(秒)で実行終了までの時間を指定

```
user@iw25-user28-server:~$ sudo rpk-client -mj -s 300
```

```
Processing time 100 seconds (0 seconds user, 0 seconds system)
Skiplist entries: 0
CCR manifest state hash: 5PYNCqbX8902pk1LHIYbmfZJxvnsUauvIBsg8pcyfJU=
CCR ROA payloads hash: 5PYNCqbX8902pk1LHIYbmfZJxvnsUauvIBsg8pcyfJU=
CCR ASPA payloads hash: 5PYNCqbX8902pk1LHIYbmfZJxvnsUauvIBsg8pcyfJU=
Route Origin Authorizations: 0 (0 failed parse, 0 invalid)
AS Provider Attestations: 0 (0 failed parse, 0 invalid)
BGPsec Router Certificates: 0
Certificates: 5 (5 invalid, 0 non-functional)
Trust Anchor Locators: 5 (0 invalid)
Manifests: 0 (0 failed parse, 0 seqnum gaps)
Certificate revocation lists: 0
Ghostbuster records: 0
Trust Anchor Keys: 0
Repositories: 5
New files moved into validated cache: 0
Cleanup: removed 1 files, 12 directories
Repository cleanup: kept 0 and removed 0 superfluous files
VRP Entries: 0 (0 unique)
VAP Entries: 0 (0 unique)
```

timeoutを指定しない場合、
デフォルトのtimeoutである1時間待機する

ROAキャッシュサーバのUPLINKに問題はないか

- ROAキャッシュサーバ上でVRPが収穫できているか確認し、stayrtrをrestartする

```
user@iw25-user28-server:~$ cat /usr/local/var/db/rpki-client/json | jq '.roas | length'  
0  
user@iw25-user28-server:~$ sudo systemctl restart stayrtr
```

ROAキャッシュサーバのUPLINKに問題はないか

- ルータ上で過去のVRP情報を削除する

```
user@iw25-user28-vmx> clear validation database  
Clearing database
```

```
user@iw25-user28-vmx> clear validation session  
Cleared 1 sessions
```

ROAキャッシュサーバのUPLINKに問題はないか

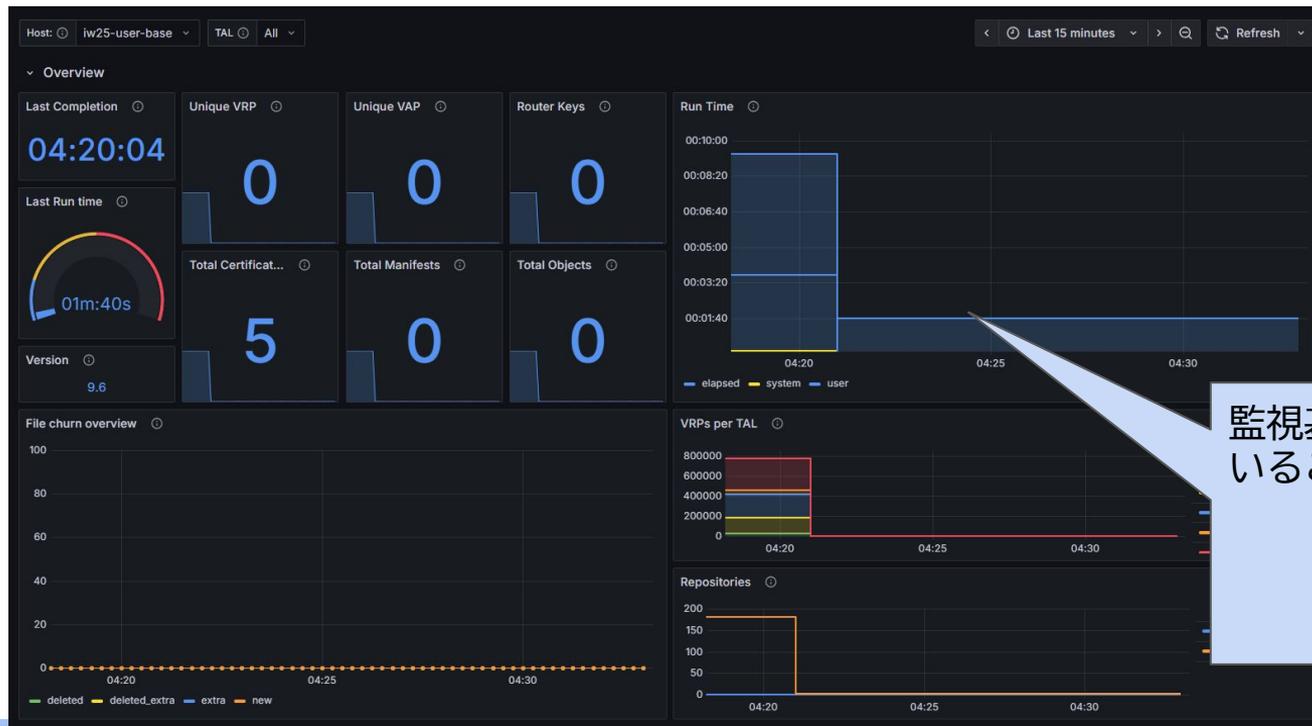
- ルータ上でROAキャッシュサーバとの接続を確認する

```
user@iw25-user28-vmx> show validation session detail
Session 172.16.250.82, State: up, Session index: 3
  Group: rpki-client, Preference: 100
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 0
  Serial (Incremental Update): 0
  Session flaps: 4
  Session uptime: 00:00:08
  Last PDU received: 00:00:05
  IPv4 prefix count: 0
  IPv6 prefix count: 0
```

stateは「up」であるがROAキャッシュサーバがROAを取得できないという状態を再現できた。
接続がうまくいっており、設定にも問題ないがVRPsは0になる状態を確認

ROAキャッシュサーバのUPLINKに問題はないか

- 監視基盤でROAの取得状況を確認する



ROAキャッシュサーバのUPLINKに問題はないか

- 元に戻す際はUPLINKを再度追加する

```
user@iw25-user28-server:~$ sudo ip route add default via 172.16.250.254 dev lan
user@iw25-user28-server:~$ sudo ip -6 route add default via 2400:5320:ff00:16fa::ffff dev
lan metric 1024
user@iw25-user28-server:~$ ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=53 time=20.6 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=53 time=20.6 ms
```

ROAキャッシュサーバのUPLINKに問題はないか

- UPLINKを戻した後rpki-clientを実行

```
user@iw25-user28-server:~$ sudo rpki-client -mj
Processing time 158 seconds (240 seconds user, 35 seconds system)
Skiplist entries: 0
CCR manifest state hash: NrRDLnO3UM+j+iumpOuM7Xy2qRf6G73Ij0xBHlt3r/Y=
CCR ROA payloads hash: lk8266pZgj8m4vM3zFmzkoEq6TD80pWrD5Q30jBojY=
CCR ASPA payloads hash: 5PFA2N6g3lZF1NZdPz01smDQ6DmBnJpoq3cCi77fzfg=
Route Origin Authorizations: 340815 (15 failed parse, 0 invalid)
AS Provider Attestations: 145 (0 failed parse, 0 invalid)
BGPsec Router Certificates: 2
Certificates: 49295 (14 invalid, 130 non-functional)
Trust Anchor Locators: 5 (0 invalid)
Manifests: 49281 (130 failed parse, 0 seqnum gaps)
Certificate revocation lists: 49151
Ghostbuster records: 1
Trust Anchor Keys: 0
Repositories: 106
New files moved into validated cache: 488555
Cleanup: removed 0 files, 87195 directories
Repository cleanup: kept 6157 and removed 3 superfluous files
VRP Entries: 776663 (771469 unique)
VAP Entries: 145 (145 unique, 0 overflowed)
VSP Entries: 0 (0 unique)
```

ROAキャッシュサーバのUPLINKに問題はないか

- ROAキャッシュサーバ上でVRPが収穫できているか確認し、stayrtrをrestartする

```
user@iw25-user28-server:~$ cat /usr/local/var/db/rpki-client/json | jq '.roas | length'  
771469  
user@iw25-user28-server:~$ sudo systemctl restart stayrtr
```

ROAキャッシュサーバのUPLINKに問題はないか

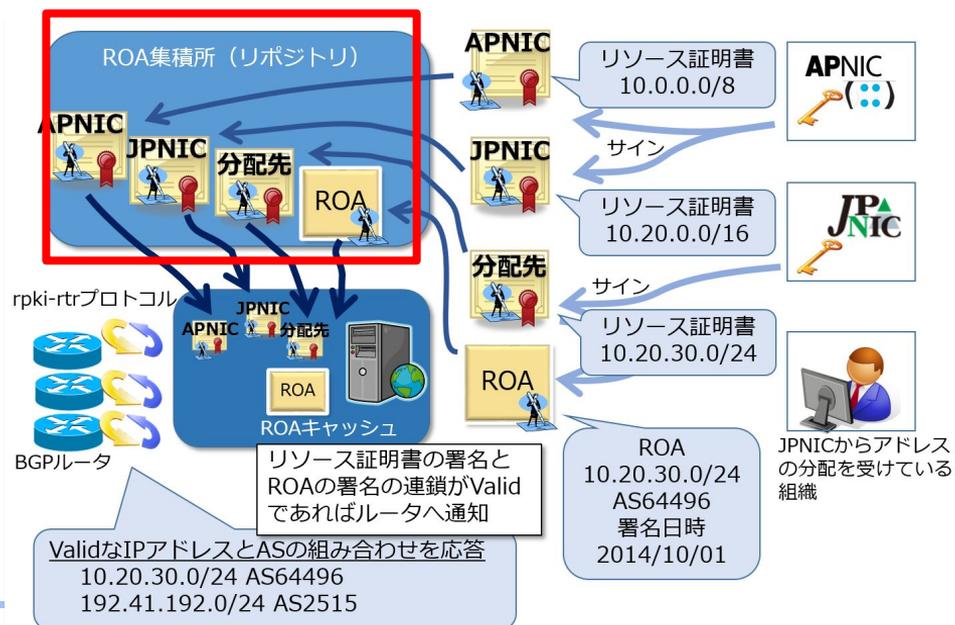
- ルータ上でROAキャッシュサーバとの接続を確認する

```
user@iw25-user28-vmx> show validation session detail
Session 172.16.251.118, State: up, Session index: 3
  Group: rpki-client, Preference: 100
  Port: 3323
  Refresh time: 300s
  Hold time: 600s
  Record Life time: 3600s
  Serial (Full Update): 0
  Serial (Incremental Update): 0
  Session flaps: 21
  Session uptime: 00:08:06
  Last PDU received: 00:00:05
  IPv4 prefix count: 610282
  IPv6 prefix count: 161187
```

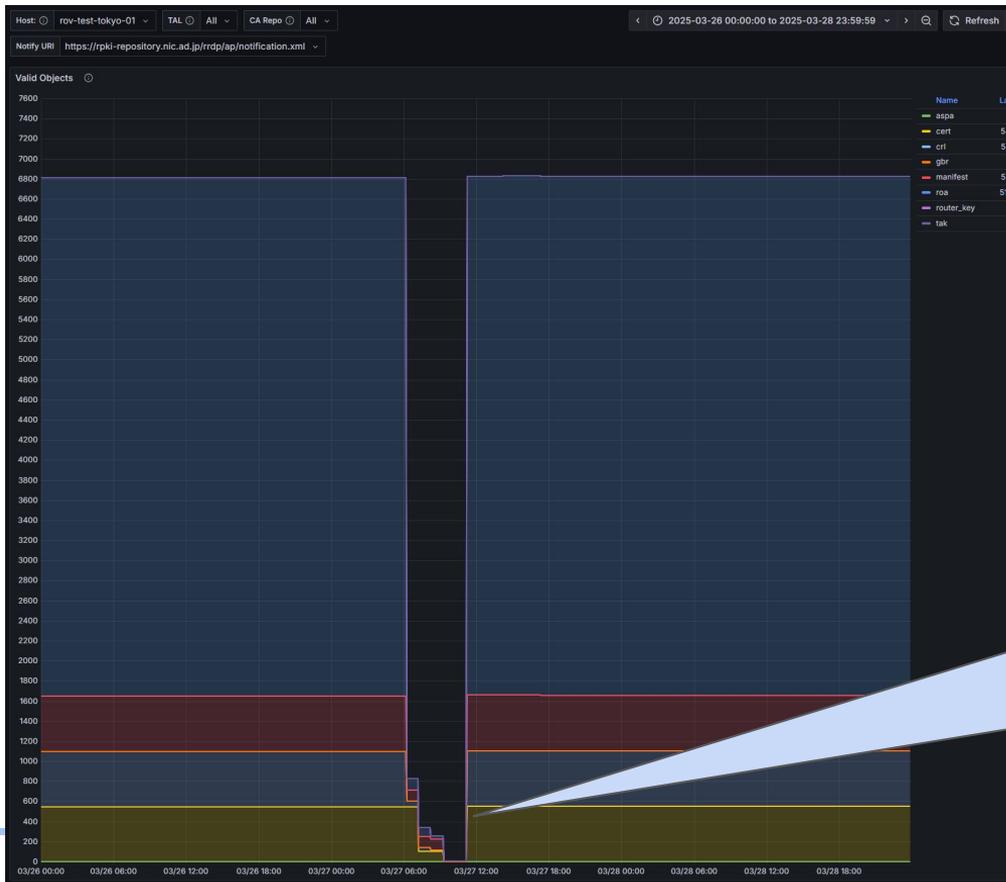
stateが「up」かつVRPを取得していることが確認できる

設定も自身のネットワークにも問題がないパターン

- 自身の環境外、リポジトリ側の問題について触れる



リポジトリ障害の実例①



障害報告：RPKIリポジトリの障害について

JPNIC RPKIシステムのリポジトリが下記の時間帯にアクセスできない状態になっております。現在は復旧しております。

利用者の皆様にご迷惑をおかけいたしましたことを、ここに改めて申し上げます。本件についてのご質問等は、下記の窓口までお問い合わせください。

記

障害によりアクセスできなかったサービスおよび日時

- RPKIシステム リポジトリ (rpk-repository.nic.ad.jp)
2024年4月15日(火) 01:15 ~ 4月16日(水) 08:40

考えられる影響

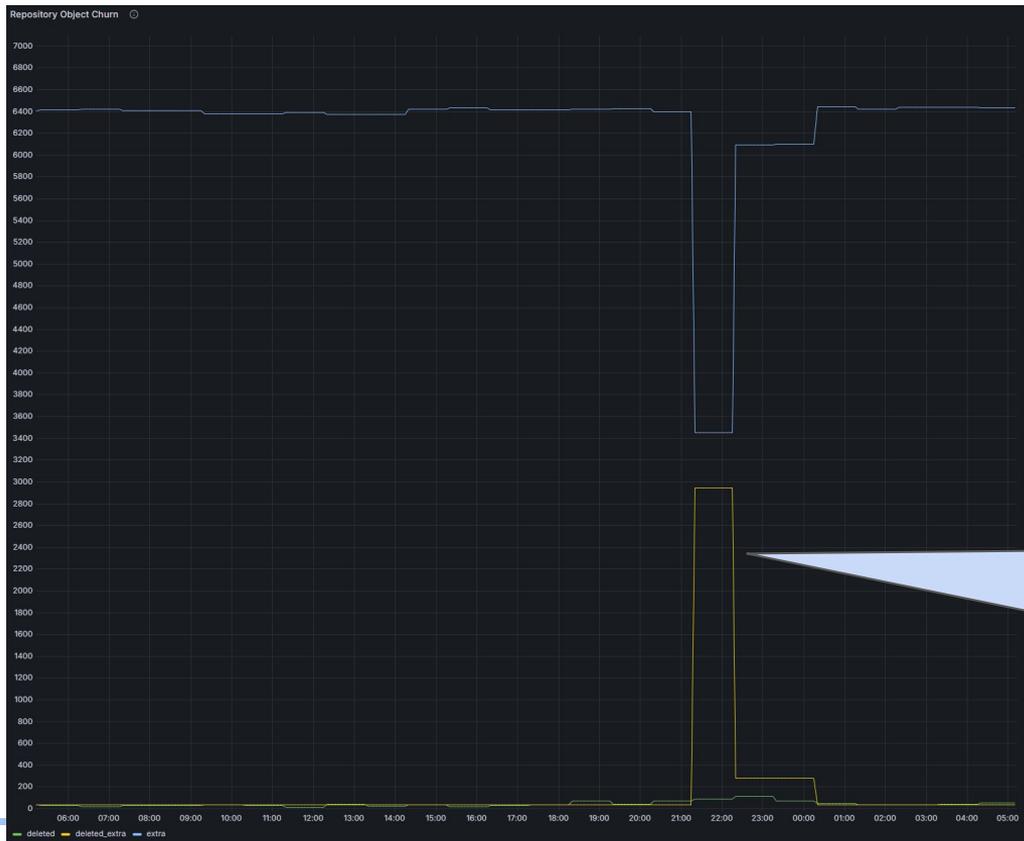
上記の期間中、ROAキャッシュサーバにおいてROAをダウンロードすることができませんでした。ROAキャッシュサーバにおけるROAキャッシュが一定期間¹を過ぎていた場合、JPNIC管理下のIPアドレスに関して最新のROAを利用したROV²を行うことができなかったと考えられます。

¹ RPKIのManifestにおけるnextUpdate(次回の更新日時)を意味しており、JPNICにおいては10日間に設定されています。

² ROAキャッシュサーバにおいて処理されたROAを使って、BGP経路の検証を行う仕組みのこと。

2025/3/27の6:00~12:00にかけて
JPNICの管理していた
5000近くあったすべてのROAが取得不可
であった

リポジトリ障害の実例②



クラウドサービスの「Cloudflare」で障害

11/18(火) 21:17 配信 44 44 44 44 44 44

ケータイ Watch



写真: ケータイ Watch

Cloudflareは18日夜、同社の設備に障害が発生していることを明らかにした。

同社サービスの状況を示すサポートページによると、20時50分ごろの情報として、グローバルネットワークに問題が発生していると案内。

さらに21時すぎに更新された情報では、**広範囲**に“500エラー”（サーバーの異常を示すコード）が発生し、ダッシュボードやAPIが機能しなくなったとし、**調査中**と伝えている。

2025/11/18の21:00～00:00にかけて
全リポジトリのオブジェクトファイルの内
3000ほどが削除(取得不可)となった。
同時期にcloudflareの障害があったことか
ら本インシデントが原因である可能性

リポジトリ側に問題が発生したパターン

- リポジトリで問題が起きるとどういった支障をきたすか

該当リポジトリの管理するオブジェクトの一部もしくは全てが取得できなくなる



本来ValidもしくはInvalidの経路がunknownとして扱われるようになる

リポジトリ側に問題が発生したパターン

- `accept[valid, unknown]`としている場合は大きな障害にはならない
 - 重みづけを変えている場合は経路が変わる可能性がある
- `accept[valid]`のみとしている場合は本来正当であるはずの経路すべてが`unknown`となる関係上、**すべての到達性を失うことになる**
- 上記の理由から基本的には`accept[valid, unknown]`とすべき

お疲れ様でした！

- 長時間にわたるハンズオン、お疲れ様でした！
- ご質問・お問い合わせは、Zoomまたは
 - okadams@sun.ac.jp
- までお願いします
- `control-slurm.py`は、必要に応じて各VMから抜き出してください
 - こちらはmc125005@sun.ac.jpまで