Internet Week 2021

インターネットルーティングの新常識
RPKIをはじめよう！

IIJ
Internet Initiative Japan

2021年11月22日

株式会社インターネットイニシアティブ
蓬田 裕一 (Yomogita Yuichi)
y-yomogita@iij.ad.jp

経歴

- 蓬田 裕一 (よもぎた ゆういち)
- 2008-2014 IIJ
  - トランジットサービス運営
  - バックボーン運用
- 2015-2019 JPNAP
  - IXサービスの運営・設計・構築・運用
- 2019- IIJ
  - バックボーン企画・設計・構築・運用
  - Peeringコーディネータ

今日の発表への思い

- IIJで行ったRPKI対応の運用状況や考え方、培ったノウハウを共有します
- 普段お話しない内容なども含まれていますのでぜひご活用ください
  - ご不明な点があれば遠慮なくご連絡ください
  - SNS/Slack/メール、何でも結構です

- RPKIを積極導入するモチベーションはインターネット全体の安定稼働への協力と参加ASとしての責任です

バックボーン

- 日々の構築・運用業務
  - 機器オペレーションや障害対応
  - 運用フローの構築
- 次世代バックボーンネットワークの検討
  - 新機種選定、検証
  - インターネット基盤の企画・設計
  - 対外接続収容設計・構築

対外対応

- Peering戦略検討と交渉
  - 顧客、ニーズを意識した戦略を検討
- 社外交渉
  - Peering以外にも、海外回線調達、海外機器の保守事業者選定
- 社外コラボレーション
  - 他社とのサービス協業の検討

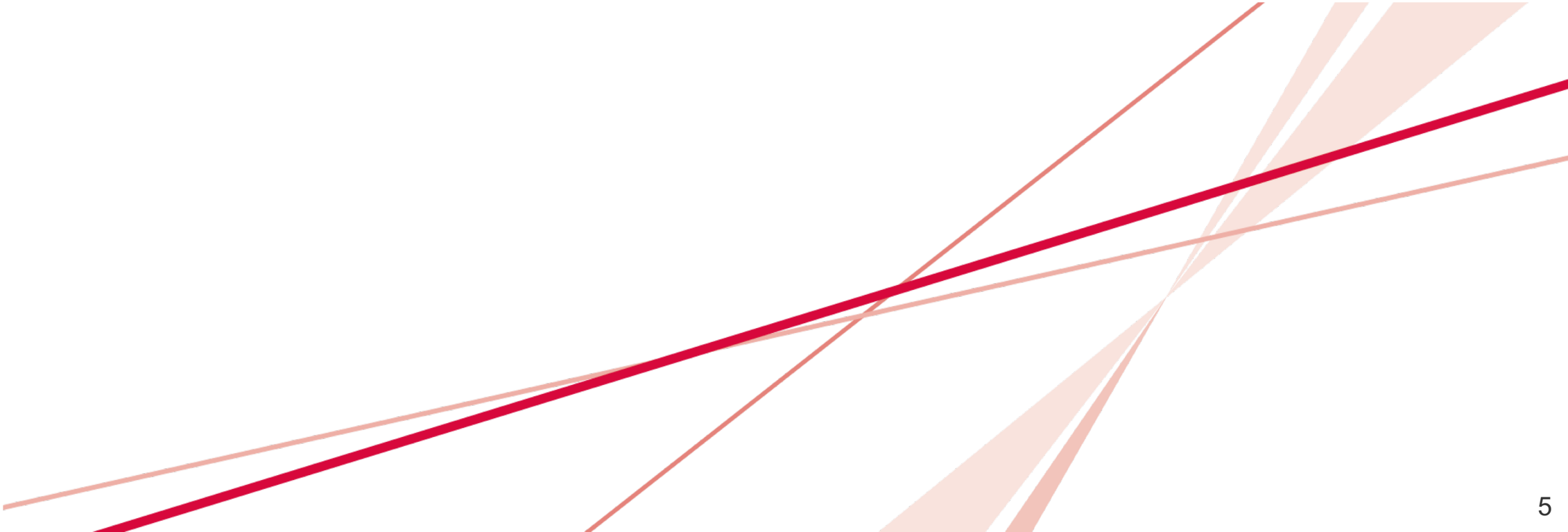
- **RPKIとはInternet Registryが番号資源の割り振り/割り当てを検証する仕組み**
- **これを自社で有効活用することによりBGPルーティングのセキュリティを高めることができる**

1. **自社保有アドレスのROA (Route Origin Authorization) を発行し、外部組織がRPKIによる検証を可能にする**
   - 自社保有アドレスの経路ハイジャックを抑止する

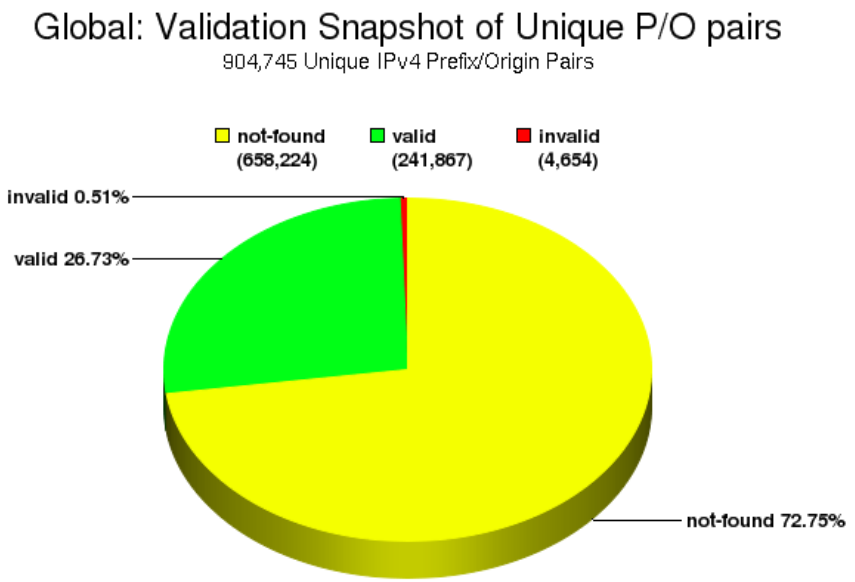
2. **他社から受信するBGP経路をRPKIで検証する**
   - ROV (Route Origin Validation)
   - ハイジャックされた経路に誘導しないことでセキュリティリスクや通信不具合から自社サービスやユーザを守る
   - (主にトランジット事業者)不正経路を伝搬させないことでインターネット全体の治安を守る

2016	社内でRPKI導入の気運が高まる (散発的に検証は進めるも停滞)
2019末	導入の気運が再燃。本気出し始める
2020/3	ROV: 機能検証開始@Lab
2020/x	ROV: 機能検証開始@実網
2020/7	手応えを掴み実導入に向けてプロジェクト化
2020/10	ROA: 試験登録、検証
	ROV: 本番機でROAキャッシュとのRTR接続開始
2020/11	ROA: (現時点で可能な自社保有IP) 本番登録開始
	ROV: (Peer,Upstream) invalid経路のreject開始
2020/12	ROV: (Peer,Upstream) invalid経路のreject完了
	ROA: (一部を除く) 本番登録完了
2021/x	ROA: 随時ROA登録中
2021/11	ROV: ROAキャッシュのアップグレード
2022(予定)	ROV: (Transit Customer) invalid経路のreject 予定
	ROA: (自社保有IP全て) 登録完了
	(可能な限りの持ち込みアドレスの) 登録完了

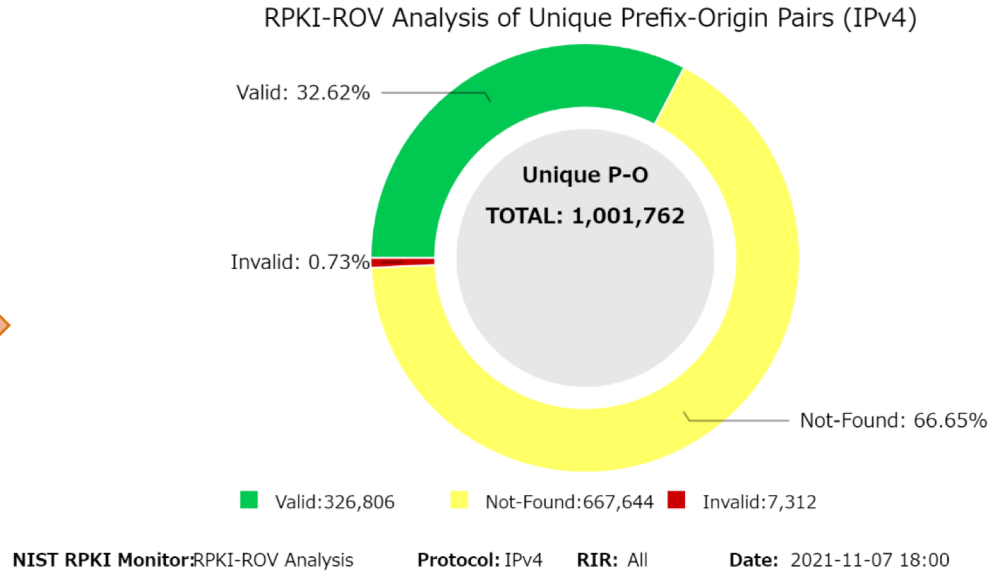
ROA (Route Origin Authorization)



2021-01-14



2021-11-08



出典: https://rpki-monitor.antd.nist.gov/

- **JPNIC保有アドレスのROAカバー率の状況**
  - IPv4: 44.4%(アドレスの個数)
  - IPv6: 57.2% (/48単位での個数)

IPv4: 45.5%
IPv6: 57.1%

JPNIC調べ

- **ROA作成ポリシーの策定**
  - ROA作成手法の検討
    - RIR/NIRのRPKIシステムを利用 (RPKI as a service)
    - 自社でCAを運用する (BPKI: Business PKI)
  - ROA発行すべきIPアドレスの調査・把握・選択
    - 自社保有アドレスの利用状況とポリシーの再確認
    - 外部への広告状況(Origin ASやprefix長)の確認
    - 自社AS利用 or 他社AS利用？
  - ROA発行パラメータの決定
    - Prefix/Subnet
    - Maximum-Prefix
    - Origin AS
- **ROA発行・運用体制の確保**
  - ROA発行者の決定
  - ROA発行の作業体制
  - ROA発行状況確認と監視

- **手段は2つ**
  - NIR/RIRのRPKIシステムを使う (RPKI as a service)
  - 自社でCA(Certificate Authority)を運用する (BPKI: Business PKI)
- **RPKI as a service**
  - ROA発行/削除について既にRIR/NIRでシステムを持っており、コスト負担無く開始できる
  - 発行後のROAの管理もRIR/NIRで対応
  - 懸念事項: RIR/NIRのシステムと運用に依存することになる
- **BPKI**
  - 自社でCAを独自運用
    - NIR/RIRからPublication Pointを向けてNIR/RIR配下のCAとなる
    - オープンなソフトウェアも存在
      - Krill (https://www.nlnetlabs.nl/projects/rpki/krill/), rpkid (https://github.com/dragonresearch/rpki.net)
  - RIR/NIRに依存しない運用 (RIR/NIR RPKIシステム障害時のROAとrepositoryの維持)
  - 自社アドレス管理システムとの統合、自動反映などもできる
  - 懸念事項: 運用コスト (実際、運用が適当なCAも散見される)

- **IIJはJPNIC等のRIR/NIRが用意するRPKIシステムを利用**
  - 現状では運用コストに見合ったメリットが見出せない

CAを保有している事業者 (2021-11-08時点)

https://andromeda.heficed.net/
https://ca.rg.net/
https://cb.rg.net/
https://cc.rg.net/
https://chloe.sobornost.net/
https://d23f0z9k6235a5.cloudfront.net/
https://krill-eval-ctec.charter.com/
https://magellan.ipxo.com/
https://nostromo.heficed.net/
https://repo-rpki.idnic.net/
https://repo1.rpki.qs.nu/
https://rpki-ca.idnic.net/
https://rpki-repo.registro.br/
https://rpki-rrdp.mnihyc.com/
https://rpki.admin.freerangecloud.com/
https://rpki.akrn.net/
https://rpki.as207960.net/
https://rpki.blade.sh/
https://rpki.caramelfox.net/
https://rpki.cnnic.cn/
https://rpki.console.luys.cloud/
https://rpki.dataplane.org/
https://rpki.e15f.net/
https://rpki.lir.services/
https://rpki.luys.cloud/
https://rpki.meerval.net/
https://rpki.multacom.com/
https://rpki.rand.apnic.net/
https://rpki.roa.net/
https://rpki.rpkitest.ml/
https://rpki.sailx.co/
https://rpki.tools.westconnect.ca/
https://rpki.xindi.eu/
https://rpki1.rpki-test.sit.fraunhofer.de/
https://rpki1.terratransit.de/
https://rpkica.mckay.com/
https://rrdp.afrinic.net/
https://rrdp.apnic.net/
https://rrdp.arin.net/
https://rrdp.lacnic.net/
https://rrdp.ripe.net/
https://rrdp.rpki.nlnetlabs.nl/
https://rrdp.sub.apnic.net/
https://rrdp.taaa.eu/
https://rrdp.twnic.tw/
https://rrpd.rpki.a2b-internet.com/
rsync://nostromo.heficed.net/
rsync://repository.lacnic.net/
rsync://rpki-repo.registro.br/
rsync://rpki-repository.nic.ad.jp/
rsync://rpki.afrinic.net/
rsync://rpki.apnic.net/
rsync://rpki.arin.net/
rsync://rpki.ripe.net/

- **自身で発行できる範囲**
  - 自身が保有するIPアドレスのみ
  - 現時点ではIRRような代行登録はできない

- **現状のアドレス利用、広告状況の把握**
  - 利用形態に応じたROA発行が必要
    - 広告元のOrigin AS / Prefixと広告(予定)サイズ
  - 経路広告ポリシーの再確認
    - 経路の広告サイズや分割ポリシーの再確認
      - 外部への広告状況(Origin ASやprefix長)をきちんと把握する
    - Origin ASをきちんと把握する
    - パンチングホールや他のASからの経路広告はあるか？
      - 自社AS利用 or 他社AS利用
      - 自社のIPアドレスを他社へ持ち込んでいる可能性
      - クラウド型DDoS対策サービスの契約等はないか
      - IPアドレスの貸出をしていないか

- **ROA発行に必要なパラメータは3つ**
  - Prefix / Origin AS / Maximum-length
    - Maximum-length: ROAで許容されるPrefix長のサイズ

- **IIJのROA発行ポリシー (参考)**
  - IIJへ割振済みで AS2497 Origin で経路未広告
    - Prefix: 割振アドレス / Origin AS: 0 / Max-Length: 割振サイズ
  - IIJへ割当済みで AS2497 Origin で経路広告
    - Prefix: 割振アドレス / Origin AS: 2497 / Max-Length: 広告サイズ
  - IIJへ割当済みで一部を顧客Originで経路広告
    - パンチングホールの状態
    - Prefix: 割振アドレス / Origin AS: 2497 / Max-Length: 広告サイズ
    - Prefix: 割当アドレス / Origin AS: 顧客 / Max-Length: 顧客と相談
  - Max-length
    - 原則広告経路と一致
    - Max-Lengthを不必要に大きくするべきではないという考え
    - Origin AS詐称による経路ハイジャックのリスクを低減 (ref. RFC7115)

ROAと広告経路の状態一覧

BGP Prefix	Origin AS	ROA Prefix	ROA Origin AS	ROA Maximum length	状況説明	ROV状況
172.122.0.0/15	2497	なし	なし	なし	ROAなし	unknown
172.122.0.0/22	2497	172.122.0.0/22	2497	22	完全一致ROAあり	valid
172.122.16.0/22	2497	172.122.16.0/22	2497	24	これを包含するROAあり	valid
172.122.32.0/22	2497	172.122.32.0/20	2497	20	SuperNetのROAしかない	invalid length
172.122.48.0/22	2497	172.122.48.0/20	2497	24	これを包含するROAあり	valid
172.122.64.0/22	2497	172.122.64.0/22	61215	22	Origin違いで完全一致ROAあり	invalid ASN
172.122.80.0/22	2497	172.122.80.0/22	61215	24	Origin違いでこれを包含するROAあり	invalid ASN
172.122.96.0/22	2497	172.122.96.0/20	61215	20	Origin違いでSuperNetのROAしかない	invalid ASN
172.122.112.0/22	2497	172.122.112.0/20	61215	24	Origin違いでこれを包含するROAあり	invalid ASN
172.122.128.0/22	2497	172.122.128.0/22	0	22	Origin0で完全一致ROAあり	invalid ASN
172.122.144.0/22	2497	172.122.144.0/22	0	24	Origin0でこれを包含するROAあり	invalid ASN
172.122.160.0/22	2497	172.122.160.0/20	0	20	Origin0でSuperNetのROAしかない	invalid ASN
172.122.176.0/22	2497	172.122.176.0/20	0	24	Origin0でこれを包含するROAあり	invalid ASN
202.16.104.0/21	2497	202.16.104.0/21	2497	21	Exact Match	valid
202.16.104.0/24	61215	202.16.104.0/24	61215	24	パンチングホール	valid
202.48.108.0/23	2497	202.48.108.0/23	2497	23	Exact Match	valid
202.48.108.0/24	61215	202.48.108.0/23	2497	23	経路ハイジャック	invalid ASN

- **自社のIPアドレス管理状況の把握**
  - RIR/NIRのRPKIシステムを使うためには準備が必要
    - JPNICの場合は個人に紐づく資源申請者証明書が必要
    - RIRによってはアカウント作成や公開鍵の登録が必要

- **IIJのROA作成・削除・変更の役割**
  - IIJの場合はアドレス管理およびRIR/NIRの窓口業務を行う組織が存在
    - LIR (Local Internet Registry)の機能を保有
    - RPKIを始めるまではネットワーク運用部隊(NOC)にROA発行権限はなかった
  - AS運用ポリシーに従ってROAを作成したいので、NOCメンバへROA発行権限を付与
    - LIRは非運用チームのため、密に連携を取ることは困難であった

- **LIR/NOCの作業分担 (参考)**
  - LIR: 新規にアドレスが割り振られたらOrigin AS0でROA作成 (未広告アドレスのフリーライド防止)
  - NOC: 実利用開始時(= AS2497広告時)にOrigin AS2497へ変更
  - NOC: 実利用終了時にOrigin AS0へ変更しLIRに返上
  - LIR: Origin AS0なアドレスを返却 (ミス防止)

- **JPNIC ROA Web**
  - https://www.nic.ad.jp/ja/rpki/
- **オペレーション**
  - **危険/リスクが伴う変更作業。一瞬で自社への到達性を失わせることも可能性**
  - JPNIC ROA WebはROA登録直前に想定されるROV状態を表示してくれるため安心
  - 変更はなく、削除 + 再作成
  - 現時点ではWeb UIゆえ自動化は困難
    - IIJの場合: 作業は必ず2人体制。操作履歴はTeamsを利用し画面録画
- **作業前後のROA発行状況の確認**
  - APNIC: https://netox.apnic.net/
  - JPNIC: http://roa2.nic.ad.jp:8080/roas
  - RIPE: https://rpki-validator.ripe.net/ui/
  - Cloudflare: https://rpki.cloudflare.com/?view=validator
  - IIJ: 自社ネットワーク内のROAキャッシュサーバ

- **リソース証明書**
  - 存在すればROAが発行できる

リソース証明書の一覧

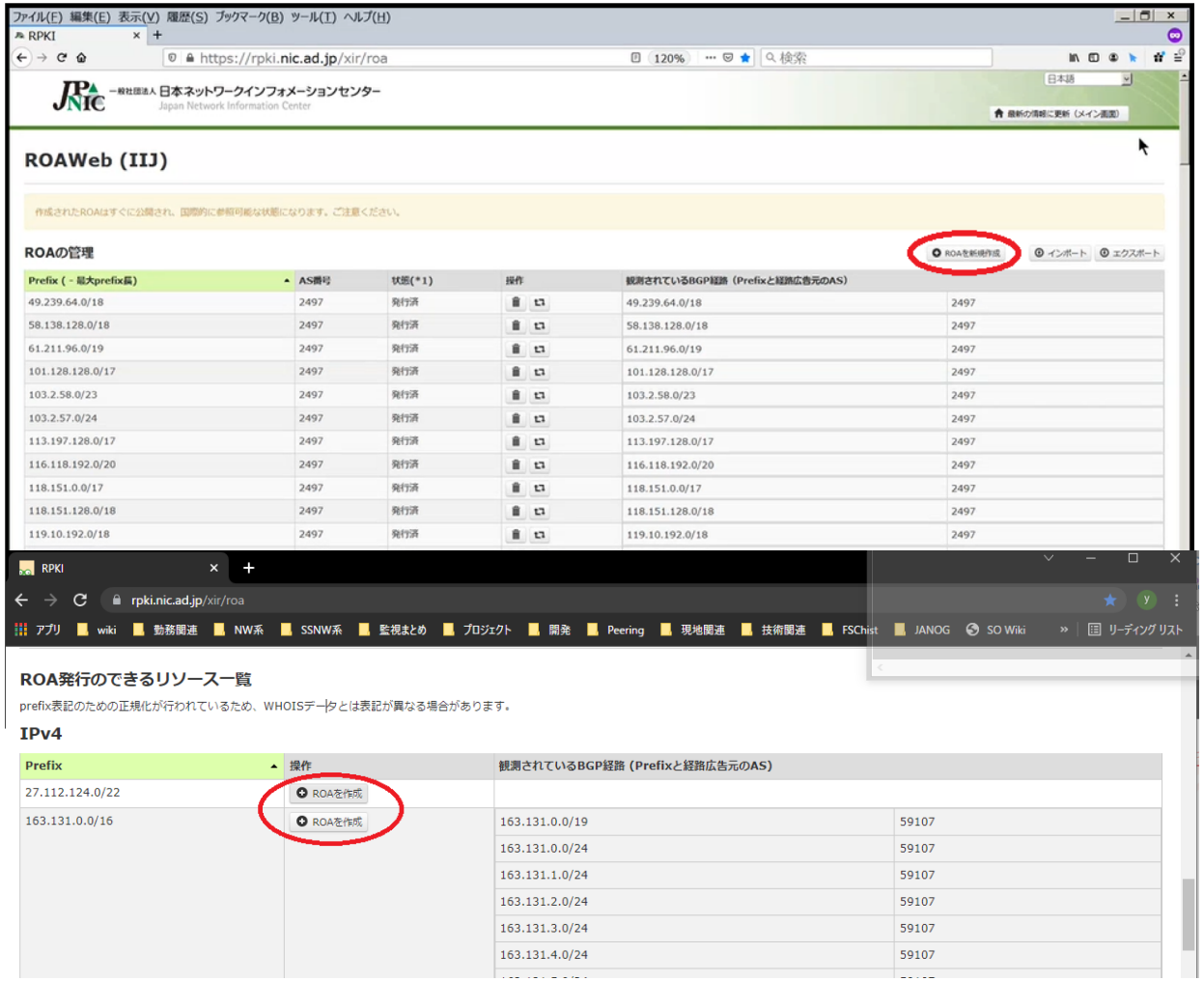
リソース証明書が「発行済」になるとROAを作成できます。リソース証明書が「発行済」になるまでに2分程度かかることがあります。

ファイル名	KWlm3XIMMFdfYtj9QUlFipCqyqQ.cer
状態	処理中
有効期限 - 自動更新	2022年10月15日10:30:02 (日本時間/UTC+9)
IPv4	157.65.8.0/21 157.65.176.0/21 157.65.192.0/21 157.65.216.0/21 160.13.0.0/16 172.122.0.0/15 192.244.32.0/19
ファイル名	**KYPP1fjK-gBo0uuiXtD3ASRLEvk.cer**
状態	処理中
有効期限 - 自動更新	2022年10月15日10:30:02 (日本時間/UTC+9)
IPv4	27.112.124.0/22

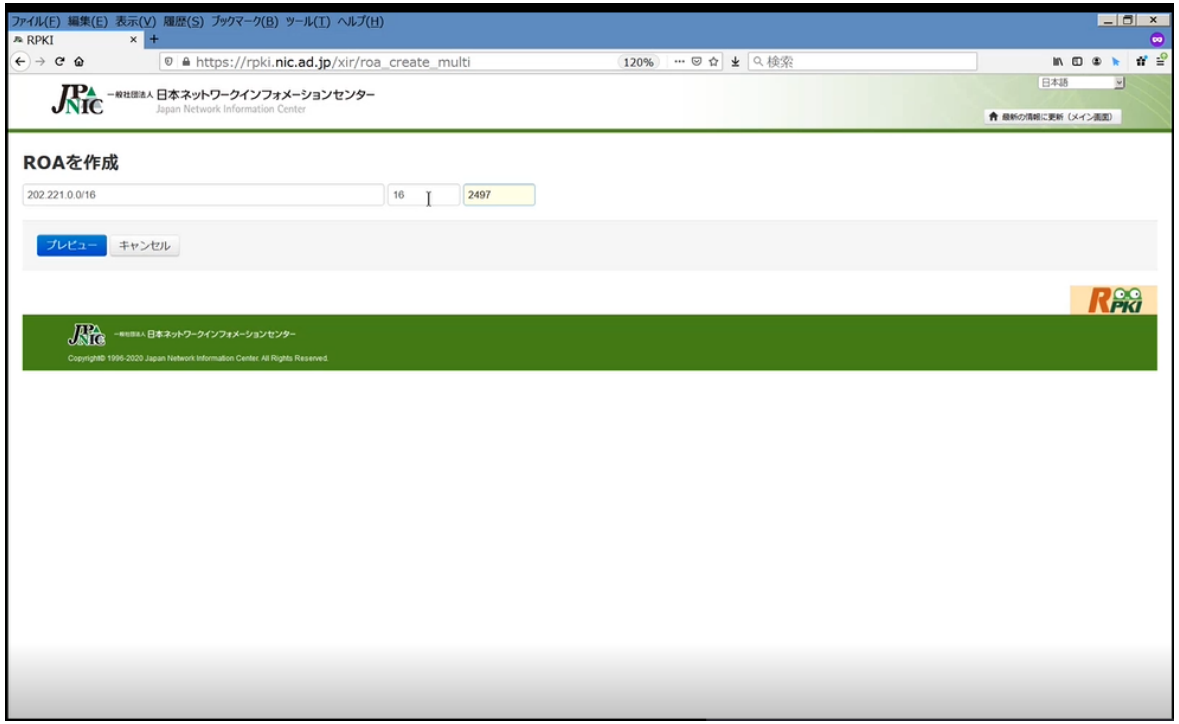
- **リソース証明書が存在しない場合はJPNICへご相談**
  - JPNICで手動で登録可能な場合も

	割り振られているが証明書に存在しない
IPv4	133.110.0.0/16 133.137.0.0/16 133.142.0.0/16 133.159.0.0/16 133.218.0.0/16 133.236.0.0/16 133.238.8.0-133.238.255.255

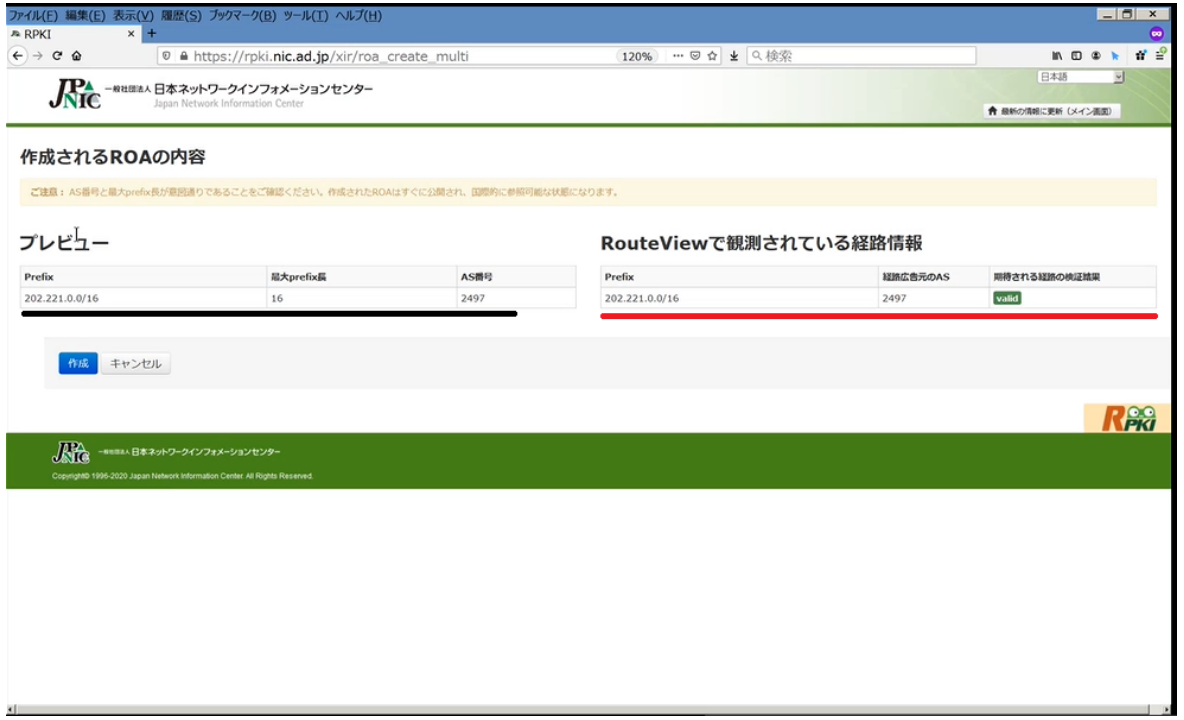
- **発行は “ROAを新規作成”をクリック**



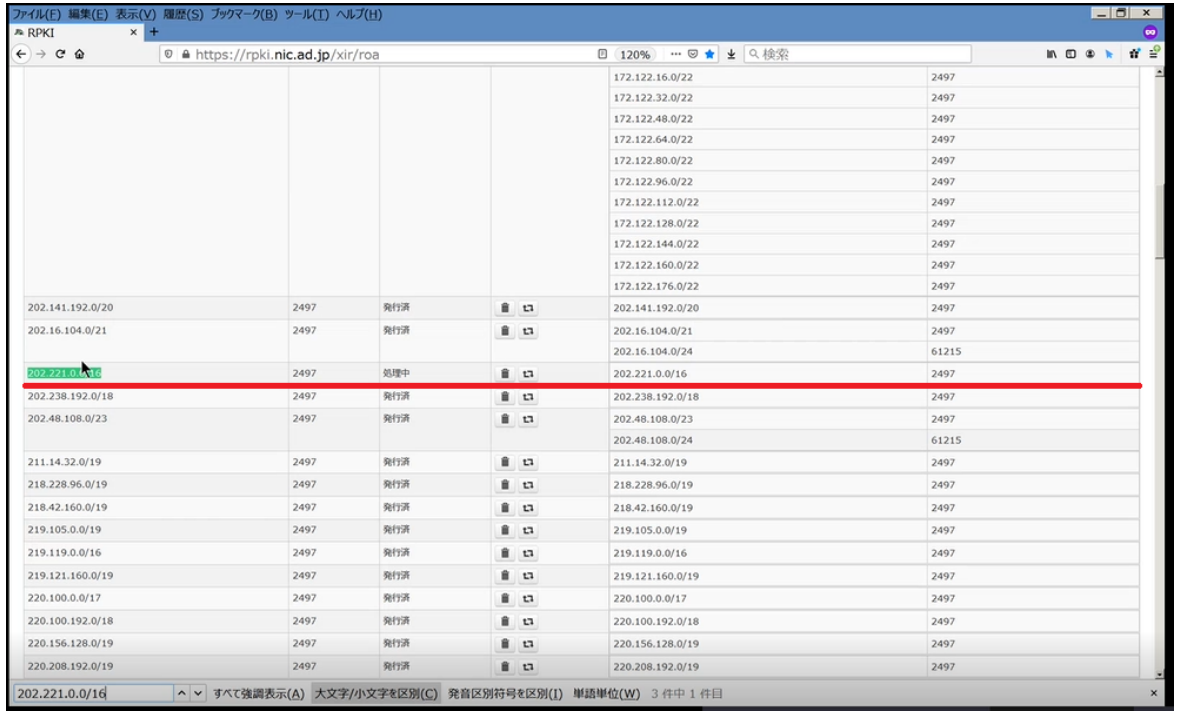
- **作成時に入力する情報**
  - Prefix / Max-length / Origin AS
  - しっかり確認！



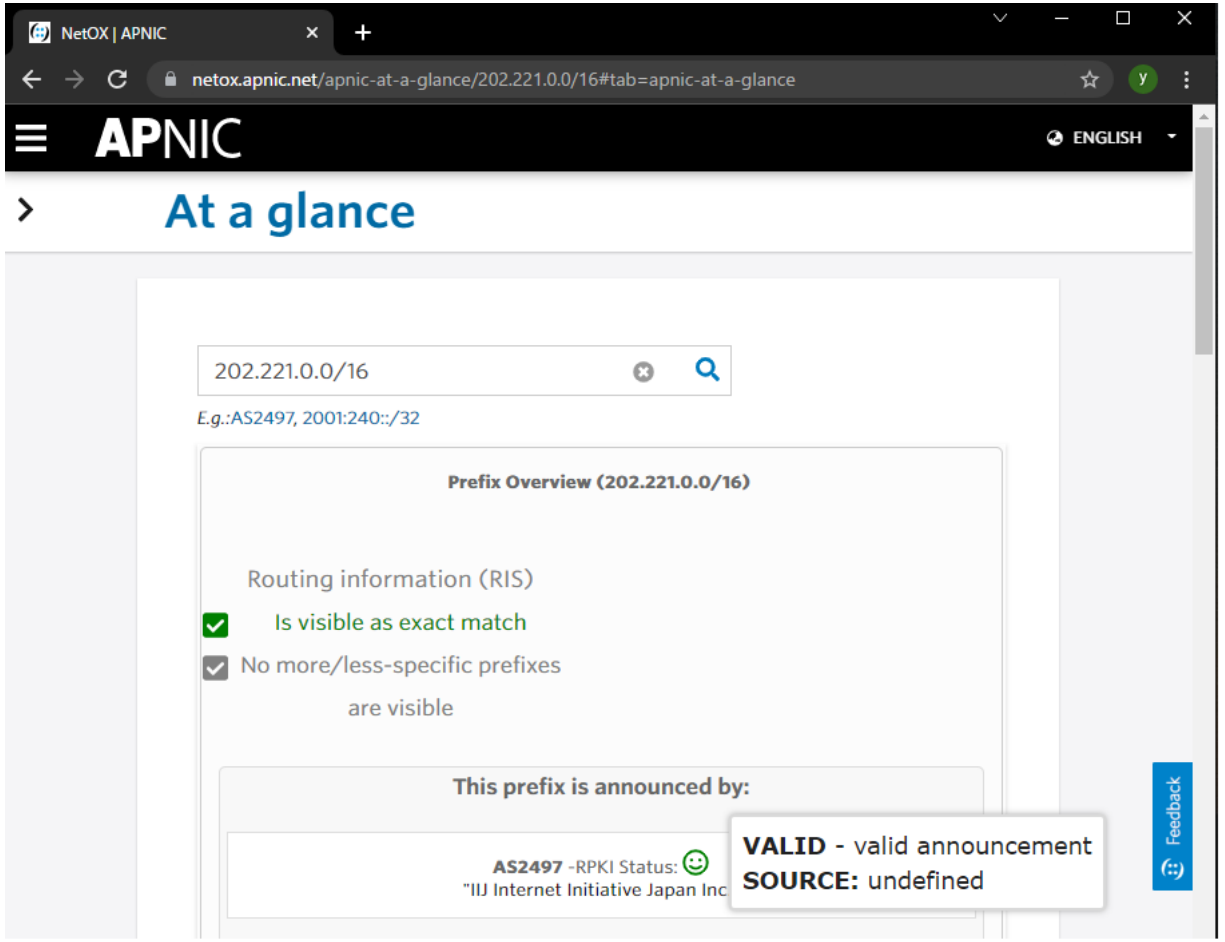
- **プレビュー画面**
  - ROA作成後のROV状況がわかる
  - invalidは要確認！



- **作業後はトップページで発行済**



- **APNICなどの確認ツール確認**
  - 外部参照可能になっているはず



- **ROA参照可能後の想定される動き**
  1. NIR/RIRのRPKIシステムで登録
  2. NIR/RIRがROA作成
  3. (場合によっては?) RIR CAからNIR CAにPublication Pointが向く?

  (以下、世界の各ASで)
  4. TALを辿り各CAからROA取得・検証、VRP作成
  5. キャッシュがVRPロード
  6. キャッシュがルータにSerial Notify送信 or ルータがキャッシュにSerial Query送信
  7. ルータがキャッシュからVRP取得
  8. ルータがAdj-RIBs-Inを再評価、RIB/FIB更新
  9. AS内にベストパスが伝搬

- **各ASへの反映には時間が掛かる、かもしれない**
  - 各ASのROA/VRP取得タイミングはまちまち。反映時間は読めない
  - 確認方法
    - RIRでの存在確認/ 自社内のROAキャッシュで存在確認 (ROA発行の作業体制のページ参照)
    - RIRのwhoisツールでの確認 / ROVを実施している外部ASのlooking glassでvalid確認

- **何を監視するべき?**
  - 自社で作成したROAの存在確認
  - ROAの設定漏れや作成ミス
    - ROAが無いまま広告していないか?
    - ROAに反する経路を漏らしていないか?
    - パンチングホールは大丈夫か?
  - invalid経路
    - 他ASで正しくvalid扱いされているか?
    - 経路ハイジャックされていないか?

- **どうやって? 外形監視は難しいという悩み**
  - ROAキャッシュのレコード
  - BGPalerter
  - RIPE RIS
  - 経路奉行
  - その他外部サイトでの情報取得
    - https://ihr.iijlab.net/ihr/api/hegemony/prefixes/?rpki_status=Invalid&asn=2497

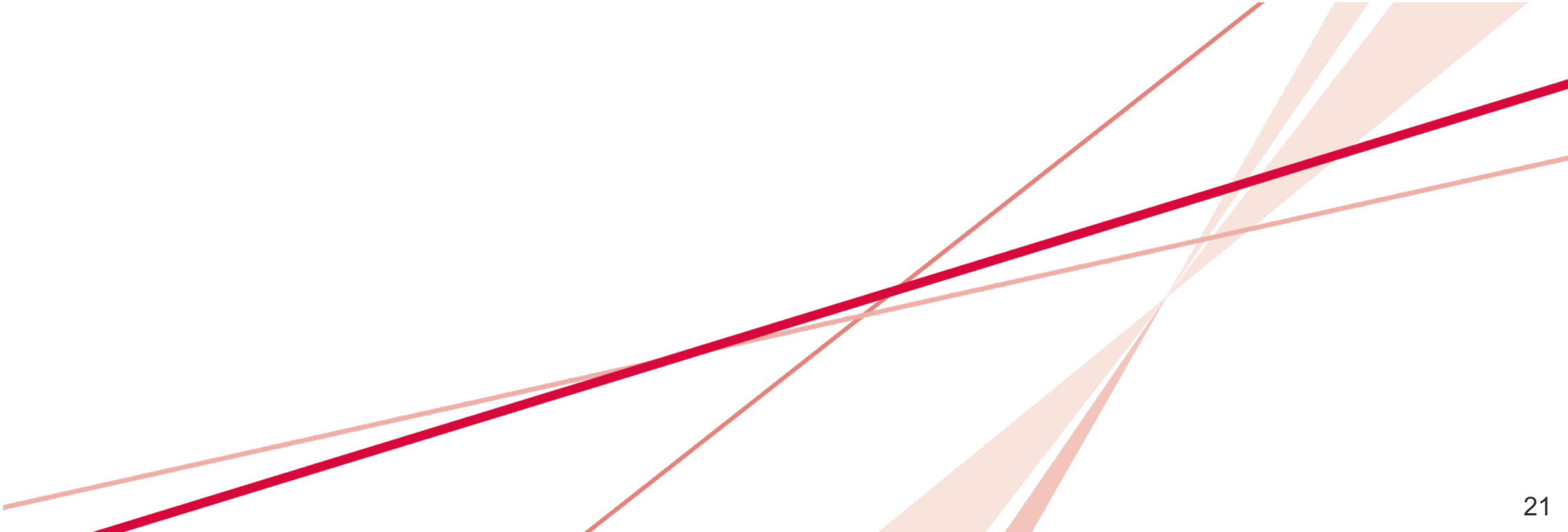
- **JPNIC ROA Webにリソース証明書がなくてROAが発行できない**
  - JPNICへ相談し、JPNICで手動対応が可能
  - 歴史的PIアドレスなどが該当する場合あり

- **Origin ASが異なる経路状況が発生しうる**
  - Origin ASごとにROA作成が必要
  - Origin AS作成ごとに作成が漏れると作成されていない側はinvalid

- **過去にROAをテストで作成して放置しているかも**
  - 現状のROA登録状況のご確認を
  - 外部に公開される情報を使えば登録状況の確認は簡単
  - 想定外の状況になっている可能性もあるためお早めに状況把握を。

- **今後IPアドレス持ち込みやPeeringの条件が”ROA発行済”となる可能性あり**
  - ROA発行は登録内容や運用方法を間違えなければデメリットはない

ROV (Route Origin Validation)



- **ROVポリシー策定**
  - そもそも実施する？
  - 基本ポリシーの策定
- **ROVパラメータの設計**
  - ルータでどう導入するか
  - ROA cache/RTRサーバはどうするか
- **導入前検証**
  - ルータ
  - Relying Party Software (ROA cache/RTRサーバ)
- **社内や顧客への導入調整と通知**
  - 自社内への導入前調整
  - 顧客/社外への導入前調整
- **ネットワークへデプロイ**

- **そもそもROV実施する？**
  - https://stats.labs.apnic.net/rpki/JP
    - APNICが調査しているROV状況(invalid経路の伝搬具合の調査)

ASN	AS Name	RPKI Validates	Samples
AS58650	CATVWINK Warabi Cable Vision Co., Ltd.	100.00%	62
AS53813	ZSCALER-INC	100.00%	139
AS18126	CTCX Chubu Telecommunications Company, Inc.	98.52%	2,156
AS10013	FBDC FreeBit Co.,Ltd.	98.17%	383
AS131916	BAYNET Tokyo Bay Network Co.,Ltd.	98.04%	51
AS4713	OCN NTT Communications Corporation	97.84%	21,846
AS131918	SCN-NET SHONAN CABLE NETWORK	96.77%	62
AS4686	BEKKOAME BEKKOAME INTERNET INC.	96.55%	58
AS2497	IIJ Internet Initiative Japan Inc.	96.55%	1,680
AS20473	AS-CHOOPA	96.17%	444
AS2519	VECTANT ARTERIA Networks Corporation	95.97%	2,429
AS17506	UCOM ARTERIA Networks Corporation	95.75%	2,049
AS45102	ALIBABA-CN-NET Alibaba US Technology Co., Ltd.	95.62%	137
AS131925	CYBERHOME-2 FAMILY NET JAPAN INCORPORATED	95.56%	90
AS2510	INFOWEB FUJITSU LIMITED	95.50%	556
AS2514	INFOSPHERE NTT PC Communications, Inc.	95.47%	1,301
AS36236	NETACTUATE	95.45%	88
AS63949	LINODE-AP Linode, LLC	95.45%	29,118
AS55391	MF-NATIVE6-E INTERNET MULTIFEED CO.	95.26%	738

    - 自身でROV実施していればもちろんOK
    - 自身のTransit Providerが実施済みであればinvalid経路は伝搬しない(擬似的にROV)

- **基本ポリシーの策定**
  - どこでROVを行う必要があるか
    - transit : Internetからのfull routeを不正経路の流入を防ぐ
    - peer: Peerからの不正経路の流入を防ぐ
    - customer: 顧客からの不正経路の流入を防ぐ。Internet上へ不正経路を伝搬させない
    - AS内部: Private AS等からの不正経路の流入を防ぐ
  - Validation State(valid/invalid/not-found)をどうroutingに反映させるか
    - Reject
    - BGP attribute
    - 経路の優先度へ反映

- **IIJの場合 (参考)**
  - どこで? → AS境界でROVする
    - 対Peer、対Upstream、 対Transit Customerが対象 (内部のeBGPは対象外)
  - Routingポリシー
    - invalid経路を一律破棄
    - valid, not found, unverifiedは等価に扱う

- **ルータ**
  - キャッシュとの接続(RTR)パラメータ (RFC8210)
    - Refresh Interval: ルータがROA Cacheに定期問い合わせする間隔
    - Retry Interval: ルータがROA Cacheへの問い合わせに失敗したときの再送間隔
    - Expire Interval: ROA Cacheとのやり取りが途絶えてから、ルータが保持しているデータを消すまでの時間
  - routingへの反映ポリシー
    - ROV結果を使ったPolicy-mapの作成と反映方法の検討
  - ROV結果をcommunity(RFC8097)としてiBGPに流すか
  - 対象OS選定
    - 検証するOSを選定する
  - 監視
    - RTRとの接続
    - VRP数、Validation State (SNMP等による外部からの取得は未実装なOSが多い)
    - RIPE RIS監視経路等を使ったROV動作のサンプリング監視
    - BMP等によるReject経路の記録

- **IIJの場合 (ルータ)**
  - キャッシュとの接続(RTR)パラメータ (RFC8210)
    - config例を参照。JUNOSとIOS-XRでパラメータは差異あり
  - routingへの反映ポリシー
    - invalidはreject、validとnot found/unverifiedは等価に扱うpolicy mapをneighborごとに適用
  - ROV結果をcommunity(RFC8097)としてiBGPに流すか
    - 流さない
    - 一時期流していたが想定通りの動作にならなかったので廃止
  - 対象OS選定
    - (現時点では) IOS-XR/JUNOS
  - 監視
    - RTR session
    - ROV後の経路状態(invalid経路数と特定時間のinvalid情報)

- **Relying Party Software/ ROA Cache & RTR server**
  - アプリ選定
    - OSSを利用する or 自社で開発する
    - 外部のPublic ROA Cacheを利用するのも手
      - JPNICも用意: https://www.nic.ad.jp/ja/rpki/howto-usepubcache.html
  - ROA取得検証とRTRを統合するか、分離するか
    - OSSの実装にもより、両方できるOSSもある
    - 片方の機能しかないものもあり、その場合は組み合わせが必要
  - 冗長性
    - どこまで冗長をもたせるか？
    - Host/OSの種類/データセンタ等の場所 etc
  - 監視
    - 各OSSともmetricは充実、GrafanaやPrometheus等のアプリによる可視化は容易
    - 何を監視するか(異常時にアラームを上げるか)の選定・しきい値設定が難しい
      - rsync/rrdp失敗は日常的に発生

- **IIJの場合 (Relying Party Software/ ROA Cache & RTR server)**
  - アプリ選定
    - OSSを利用
      - ROAの情報を取得したい/ 障害やメンテナンス,OSアップグレードの融通を取りたい /冗長性を担保したい / SLRUMを利用したい 等の理由より
    - Routinator / RPKI-Validator→ Fort
      - Routinator:デプロイも日々の運用も容易。GUIあり
      - RPKI-Validator: 2021/7開発終了したので次に評判がよさそうな Fortへ移行検討中
  - ROA取得検証とRTRを統合するか、分離するか
    - 統合
  - 冗長性
    - OSは2種類を用意。serverは東阪のデータセンタへ設置
    - 国外への設置は無し
  - 監視
    - RTRセッション
    - host監視は他の運用ホストと同様に実施(死活監視/リソース監視/プロセス監視 etc)
    - Prometheusによるmetrics取得とGrafanaによるmetricsの可視化

- **JUNOS config 例**

```
# RTR server session
routing-options {
  validation {
    group RPKI {
      session xxx.xxx.xxx.xxx {
        refresh-time 600;
        hold-time 1200;
        port 8323;
        local-address xxx.xxx.xxx.xxx;
      }
      session xxx.xxx.xxx.xxx; {
        refresh-time 600;
        hold-time 1200;
        port 8323;
        local-address xxx.xxx.xxx.xxx;;
      }
    }
  }
}

# BGP neighborに反映
protocols {
    bgp {
      group PEER {
        import [ OriginValidation ];
      }
    }
}
```

```
# ROV Policy map
policy-options {
  policy-statement OriginValidation {
      term Valid {
          from {
              family inet;
              validation-database valid;
          }
          then {
              validation-state valid;
              next policy;
          }
      }
      term Unknown {
          from {
              family inet;
              validation-database unknown;
          }
          then {
              validation-state unknown;
              next policy;
          }
      }
      term Invalid {
          from {
              family inet;
              validation-database invalid;
          }
          then {
              validation-state invalid;
              reject;
          }
      }
      then next policy;
  }
}
```

- **IOS-XR config 例**

```
!
router bgp 2497
# RTR server session
 rpki server xxx.xxx.xxx.xxx
  transport tcp port 8323
  purge-time 360
  refresh-time 600
  response-time 1300
 !
 rpki server xxx.xxx.xxx.xxx
  transport tcp port 8323
  purge-time 360
  refresh-time 600
  response-time 1300
 !

# RPKI 有効化
 bgp bestpath origin-as allow invalid
 address-family ipv4 unicast
  bgp origin-as validation enable
 !

# BGP neighborに反映
 neighbor xxx.xxx.xxx.xxx
  address-family ipv4 unicast
   route-policy importPolicy in
 !
!
```

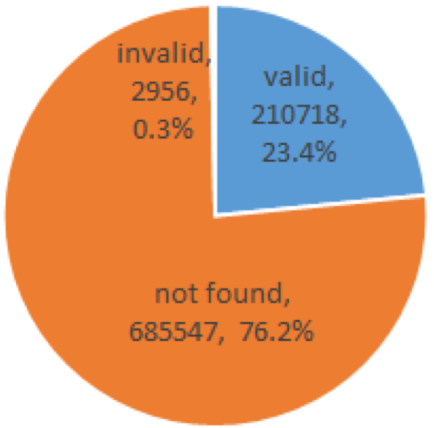
```
# ROV Policy map
route-policy importPolicy
  apply OriginValidation
  pass
end-policy
!
route-policy OriginValidation
  if validation-state is invalid then
    drop
  endif
end-policy
!
```

詳しくはIIJ Engineers Blogを御覧ください
インターネットをよりロバストに。RPKIはじめました
https://eng-blog.iij.ad.jp/archives/9320

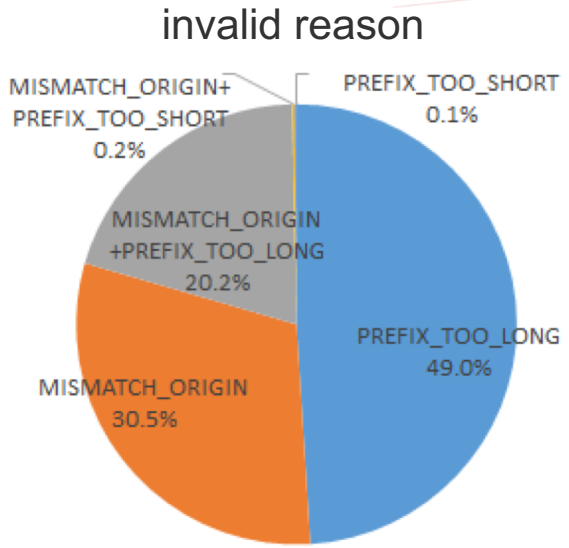
- **ルータ**
  - 機能要件
    - RFC6907 7.1 Prefix-Origin Validation Use Cases に沿った動作をするか
    - AS_SETの扱い、best path selection、監視用metric
    - 実ROA + 試験用ROA(SLRUMで作成)で検証
      - ルータで不定期にVRPが消える不具合も確認
  - 非機能要件
    - CPU・メモリ負荷、適用時のforwardingへの影響

- **Relying Party/Cache**
  - 機能要件
    - rsync/rrdpによるROA取得・検証、RTR、SLRUM、監視用metric
  - 非機能要件
    - RTRセッション数によるCPU ・メモリ負荷

- **顧客フロント(サポート、営業)が気にすること**
  - IIJの場合は、趣旨に対する反対意見は皆無だったが…
  - 導入時においては **ROVによるメリット <<< ROVによる通信影響** が懸念された
    - 理由がなんであれ(本当の不正、mis-configuration)、ROVによりこれまでできていた通信ができなくなる不安
  - ROV結果を用いて自社内の経路がどうなるか分析が必要
    - IIJの場合
      - 内部のBGP経路をROVしてみてinvalid経路を確認
      - 地道にinvalidな理由を確認する
        - Routerで持っているVRPs recordとの比較
      - flowで経路のトラフィック量を確認



- **ROVで破棄されるinvalid経路の分析** ***IIJ内で2020/9頃実施**
  - およそ3,000経路
    - フルルートの0.3%、多いような少ないような…
  - 当然、これらが本当の不正経路か、設定ミスかはわからない
  - 3,000経路/0.3%だけを見せるとさらに不安が募るので深掘りは必須



- **PREFIX_TOO_LONG**
  - Max Lengthより細かい
  - AS内の細かい経路を漏らしてる?
- **MISMATCH_ORIGIN**
  - Origin ASがおかしい。パンチングホールかも?
- **MISMATCH_ORIGIN + PREFIX_TOO_LONG**
  - 細かい経路 & Origin ASおかしい
  - パンチングホールでありがち

いずれも本当の不正経路の可能性はあるが
それなら対処して長期間は続かないはず…

設定ミスや考慮漏れの可能性が高いか?

- **invalid経路を破棄した場合、実際に到達性がなくなるのはどのくらい?**
  - あるinvalid経路を破棄しても、代わりの経路があれば到達性は保たれる
  - 何を持って「代わりの経路がある」とするかにより異なる
  - Invalidでない && invalid経路より短い && invalid経路と同じOrigin AS
    - 到達性を失うのはフルルートの0.152%
  - invalidでない && invalid経路より短い
    - 到達性を失うのはフルルートの0.097%

- **これら宛先との通信量を推定**
  - samplingした経路との通信量をNetFlowで調査
  - 散発的に少量が流れているのみ

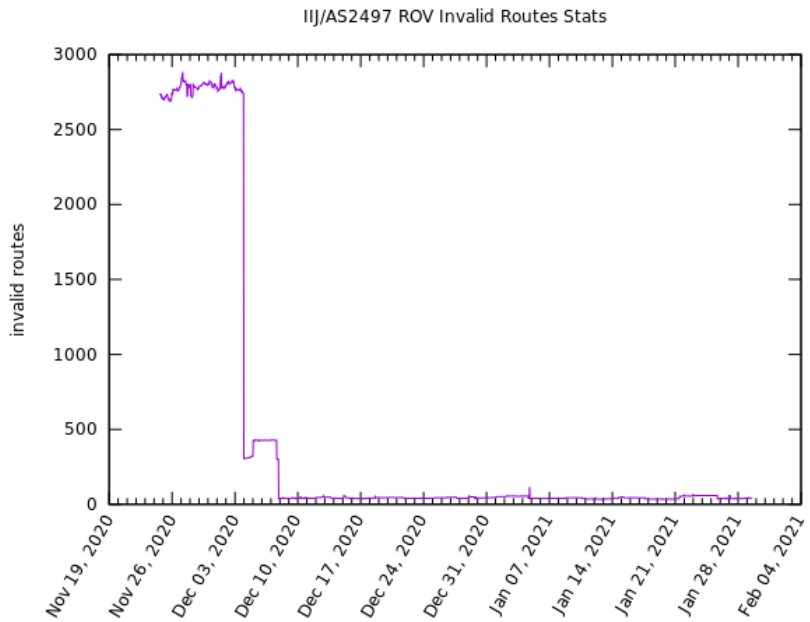
- **→ 3,000経路を破棄しても大きな通信影響はない!**
  - もちろん全くないとは言い切れないが個別対応可能なレベルと判断

- **Transit Customerへの影響**
  - 自身が配下や顧客へfull routeを広告していれば、自社からのinvalid経路の広告は無くなる
    - 自社のみがinvalid経路を広告していればROVでrejectした際に配下はinvalid経路を失う

- **どうしてもそのinvalid経路が今必要だ! と言われたら**
  - 各ソフトウェアともSLRUM(RFC8416)という仕組みがある
  - 強引にvalidにして通す (IIJではオレオレROAと呼称)
  - 用意はしたが、未来永劫使われないことを願う
    - 本当の経路ハイジャックを許すことになるかも。乱用禁止
    - 各ASでROV導入が進む中、IIJだけ通しても意味は薄い
  - IIJではまだ発動したことはなし
    - 検証などには利用できるので上手く使う

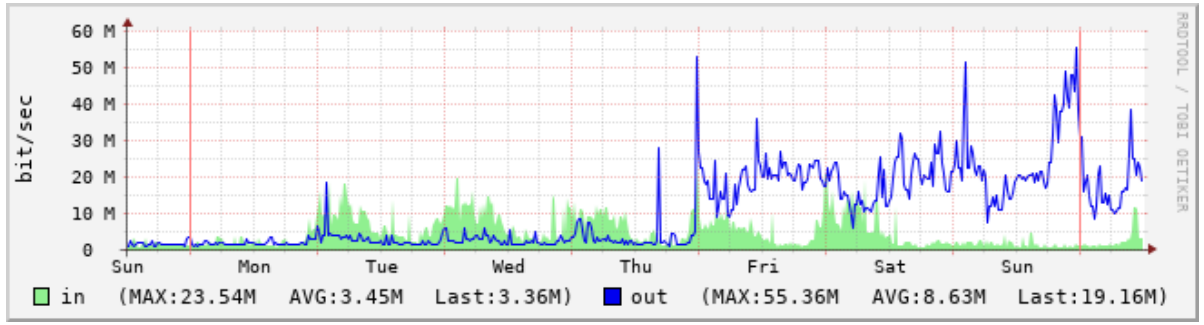
- **万が一の状況確認に備える**
  - 顧客問い合わせ時にROVの影響かどうか確認できるようにする
    - 内部経路確認用のLooking Glassを用意していく
    - RPKI ValidatorのWeb UIでの確認手法の確立
    - invalid経路dumpの取得
      - 各ルータのinvalid経路とVRPs数のdump
      - IIJの場合は内部のルータで地道に取得(telnet芸)

- **IIJ内部でもinvalid経路は0にならない**
  - 一部 AS-SET経路が正しくROVできていない
    - OSが古い事による
    - 新し目のOSだとちゃんとできる模様



- **ROVによる影響を確認しつつ、影響を最小限へ**
  - ROVを開始 → invalid経路のケア → invalidをreject

- **IIJでの設定の流れ例**
  - invalid 約3000経路の影響が未知数なのでケアする
  1. 全ノードでvalidationを開始、invalidはLocal Pref を0に設定
     - トラフィックがすごーく遷移するかもしれないことは意識
  2. 少しづつinvalid経路をrejectするように変更
     - APAC/EAMA → JP → US → Upstream
     - 最終的にUpstreamの特定接続にinvalidが集中する状況を作ってた
     - UpstreamはROV導入済みなのでこの時点ですでにinvalid経路は少ない状況ではあった
  3. 満を持してすべてreject!



- **ROA作成後にROA登録状況を変更するようなことは発生していない**
  - ROA発行後に経路広告ポリシーに大きな変更がなかった
  - 一度作ったらそうそう変更するものでもない

- **RTRセッションの不安定事象**
  - ルータ内でRTRが全断することはいまのところなし
  - RTRサーバ側のリソース問題で RTRセッションがflapする事象はたまにある
    - 基本的には1本でもあれば問題なし
    - 全部切れてもnot found/unknownになるだけとの想定 (サービス提供への影響はなし)

- **ROVがこなれてくるのはこれから**
  - 2021-07 Security Bulletin: Junos OS and Junos OS Evolved: Specific packets can trigger rpd crash when BGP Origin Validation is configured with RPKI (CVE-2021-0281)
  - 2021-10 Security Bulletin: Junos OS: Receipt of a specific BGP update may cause RPKI policy-checks to be bypassed (CVE-2021-31375)

- **Relying Party への依存**
  - 開発は何時まで継続するのか？漠然とした不安
  - 定期的な脆弱性対応
    - https://english.ncsc.nl/latest/news/2021/october/29/upcoming-announcement-of-rpki-cvd-procedure?s=03)

まとめ

- **ROA発行のポイントや作成方法、ROV導入に向けた検討事項についてお話しました**
  - 本日の発表がみなさまのRPKI導入に少しでもお役に立てれば幸いです
  - 是非、**積極的な導入検討をお願いします**

- **インターネットのBGPネットワークは各ASの高いリテラシーと継続的な安定運用によって成り立っていると思います**
- **自社の間違った振る舞いがインターネット全体へ波及することは周知の事実であり、今できる自衛手段を積極的に取り入れ、自社のセキュリティを高めることがインターネット全体の安定稼働につながると私は信じています**

- **RPKIが当たり前に運用される”明日のカタチ”をみんなで作りましょう**
  - RPKIの導入がインターネット運用の常識となる未来は直ぐそこです
  - ご不明な点やもっとここを知りたいがあればお気軽にご連絡ください
  - みなさんからのノウハウの共有も是非よろしくお願いします

日本のインターネットは1992年、IIJとともにはじまりました。以来、IIJグループはネットワーク社会の基盤をつくり、技術力でその発展を支えてきました。インターネットの未来を想い、新たなイノベーションに挑戦し続けていく。それは、つねに先駆者としてインターネットの可能性を切り拓いてきたIIJの、これからも変わることのない姿勢です。IIJの真ん中のIはイニシアティブ――IIJはいつもはじまりであり、未来です。