Inter-domain Routing Security Workshop in 秋

Yoshida Tomoya – yoshida@ocn.ad.jp
1E7D 79AD C610 B5F2 A94E 7FF4 F4AC A722 329C 3DE8

uRPFとDNS Anycastは仲良しか？

石田慶樹　yoshiki@mex.ad.jp

本日の内容

- はじめに
- uRPF とは？
- DNS anycast とは？
- uRPF と DNS anycast は仲良しか？
- 考察
- おわりに

はじめに

- 発表の目的
  - 問題意識の共有のため
    - DNS Anycast が大流行
    - Inter-domain における Filtering 技術としての uRPF がそろそろ来そう
    - 2つの技術の干渉について誰か考えているのか？
      - Routing, DNS, SecurityのエンジニアのAND!
  - 本当に問題が起こるのか？
  - 問題が起こるとすればこれからどうすべきか？

uRPF とは？

uRPF とは

- 経路情報を利用した Ingress Filter の手法
  - unicast Reverse Path Forwarding
- 利点
  - 静的な設定の整合性を保つ必要がない
  - 経路情報の変化に応じて動的に対応可能
- 欠点
  - 経路制御変更の際に細心の注意が必要

uRPF とは(続き)

- RFC3704(BCP84)
  - Ingress Filtering for Multi-homed Networks
- RPF(Reverse Path Forwarding)
  - Strict Reverse Path Forwarding
  - Feasible Reverse Path Forwarding
  - Loose Reverse Path Forwarding w/o Default

uRPF とは(続き)

Strict Reverse Path Forwarding

Strict Reverse Path Forwarding (Strict RPF) is a simple way to implement an ingress filter. It is conceptually identical to using access lists for ingress filtering, with the exception that the access list is dynamic. This may also be used to avoid duplicate configuration (e.g., maintaining both static routes or BGP prefix- list filters and interface access-lists). The procedure is that the source address is looked up in the Forwarding Information Base (FIB) - and if the packet is received on the interface which would be used to forward the traffic to the source of the packet, it passes the check.

Strict Reverse Path Forwarding is a very reasonable approach in front of any kind of edge network: in particular, it is far superior to Ingress Access Lists when the network edge is advertising multiple prefixes using BGP. It makes for a simple, cheap, fast, and dynamic filter.

But Strict Reverse Path Forwarding has some problems of its own. First, the test is only applicable in places where routing is symmetrical - where IP datagrams in one direction and responses from the other deterministically follow the same path. While this is common at edge network interfaces to their ISP, it is in no sense common between ISPs, which normally use asymmetrical 'hot potato' routing. Also, if BGP is carrying prefixes and some legitimate prefixes are not being advertised or not being accepted by the ISP under its policy, the effect is the same as ingress filtering using an incomplete access list: some legitimate traffic is filtered for lack of a route in the filtering router's Forwarding Information Base.

There are operational techniques, especially with BGP but somewhat applicable to other routing protocols as well, to make strict RPF work better in the case of asymmetric or multihomed traffic. The ISP assigns a better metric which is not propagated outside of the router, either a vendor-specific 'weight' or a protocol distance to prefer the directly received routes. With BGP and sufficient machinery in place, setting the preferences could even be automated, using BGP Communities [2]. That way, the route will always be the best one in the FIB, even in the scenarios where only the primary connectivity would be used and typically no packets would pass through the interface. This method assumes that there is no strict RPF filtering between the primary and secondary edge routers: in particular, when applied to multihoming to different ISPs, this assumption may fail.

uRPF とは(続き)

- Strict Reverse Path Forwarding
  - パケットのソースアドレスについてFIBを参照
  - パケットを受け取ったインタフェースがforwardするべきインタフェースなら、そのパケットは通過可能
  - Strict Reverse Path Forwarding ではパスの対称性があることを前提としているがそれはありえない
  - ただしこの問題は運用技術的に解決可能である

uRPF とは(続き)

Feasible Path Reverse Path Forwarding

Feasible Path Reverse Path Forwarding (Feasible RPF) is an extension of Strict RPF. The source address is still looked up in the FIB (or an equivalent, RPF-specific table) but instead of just inserting one best route there, the alternative paths (if any) have been added as well, and are valid for consideration. The list is populated using routing-protocol specific methods, for example by including all or N (where N is less than all) feasible BGP paths in the Routing Information Base (RIB). Sometimes this method has been implemented as part of a Strict RPF implementation.

In the case of asymmetric routing and/or multihoming at the edge of the network, this approach provides a way to relatively easily address the biggest problems of Strict RPF.

It is critical to understand the context in which Feasible RPF operates. The mechanism relies on consistent route advertisements (i.e., the same prefix(es), through all the paths) propagating to all the routers performing Feasible RPF checking. For example, this may not hold e.g., in the case where a secondary ISP does not propagate the BGP advertisement to the primary ISP e.g., due to route-maps or other routing policies not being up-to-date. The failure modes are typically similar to "operationally enhanced Strict RPF", as described above.

As a general guideline, if an advertisement is filtered, the packets will be filtered as well.

In consequence, properly defined, Feasible RPF is a very powerful tool in certain kinds of asymmetric routing scenarios, but it is important to understand its operational role and applicability better.

uRPF とは(続き)

- Feasible Reverse Path Forwarding
  - パケットのソースアドレスについてRIBを参照
  - パケットを受け取ったインタフェースがbestでなくとも、代替経路として利用されるインタフェースなら、そのパケットは通過可能
  - パスが非対称でも大丈夫

uRPF とは(続き)

Loose Reverse Path Forwarding

Loose Reverse Path Forwarding (Loose RPF) is algorithmically similar to strict RPF, but differs in that it checks only for the existence of a route (even a default route, if applicable), not where the route points to. Practically, this could be considered as a "route presence check" ("loose RPF is a misnomer in a sense because there is no "reverse path" check in the first place).

The questionable benefit of Loose RPF is found in asymmetric routing situations: a packet is dropped if there is no route at all, such as to "Martian addresses" or addresses that are not currently routed, but is not dropped if a route exists.

Loose Reverse Path Forwarding has problems, however. Since it sacrifices directionality, it loses the ability to limit an edge network's traffic to traffic legitimately sourced from that network, in most cases, rendering the mechanism useless as an ingress filtering mechanism.

Also, many ISPs use default routes for various purposes such as collecting illegitimate traffic at so-called "Honey Pot" systems or discarding any traffic they do not have a "real" route to, and smaller ISPs may well purchase transit capabilities and use a default route from a larger provider. At least some implementations of Loose RPF check where the default route points to. If the route points to the interface where Loose RPF is enabled, any packet is allowed from that interface: if it points nowhere or to some other interface, the packets with bogus source addresses will be discarded at the Loose RPF interface even in the presence of a default route. If such fine-grained checking is not implemented, presence of a default route nullifies the effect of Loose RPF completely.

One case where Loose RPF might fit well could be an ISP filtering packets from its upstream providers, to get rid of packets with "Martian" or other non-routed addresses.

If other approaches are unsuitable, loose RPF could be used as a form of contract verification: the other network is presumably certifying that it has provided appropriate ingress filtering rules, so the network doing the filtering need only verify the fact and react if any packets which would show a breach in the contract are detected. Of course, this mechanism would only show if the source addresses used are "martian" or other unrouted addresses -- not if they are from someone else's address space.

uRPF とは(続き)

- Loose Reverse Path Forwarding
  - パケットのソースアドレスがルーティングテーブルにあるかどうかのみを確認
  - 厳密に言うと Reverse Path Forwarding ではない
  - Default経路の処理をどうするかでさらに扱いが分かれる

uRPF とは(続き)

- 使えるか？
  - BGPカスタマに対してエッジで利用可能
  - トランジットサービスでも利用可能かも。。。
- どれくらい利用されているか？
  - JP内での利用はあまり聞かないが。。。
    - こっそり使ってますか？
  - USのトランジットISPは前から利用している模様
- BGPルータへのインプリ具合
  - USのメーカ系はほぼすべてでインプリ済か
    - 一部L3スイッチでは未実装か。。。
    - 現状はStrict ModeとLoose Modeのみか
    - JPのメーカ系は未実装も多い？

DNS anycast とは？

DNS Anycast とは

- RFC3258
  - Distributing Authoritative Name Servers via Shared Unicast Addresses
- 13個(しかない)ルートネームサーバ
  - 負荷分散
  - 耐障害性

  のための技術
- DNSの特性を利用
  - UDPで1パケット
  - 必ずQueryに対してのAnswerとなる
- 地理的に分散した複数の箇所から同一のIPアドレスブロックをアナウンスする

DNS Anycast とは(続き)

- 欠点
  - 監視ができない
- 利点
  - サーバ処理能力の増強
  - DDoS発生時の影響の制約
- どれくらいDeployしているか？
  - http://www.root-servers.org/
  - 様々な理由から積極的に進行してしまっている
    - 地理的、技術的、政治的等

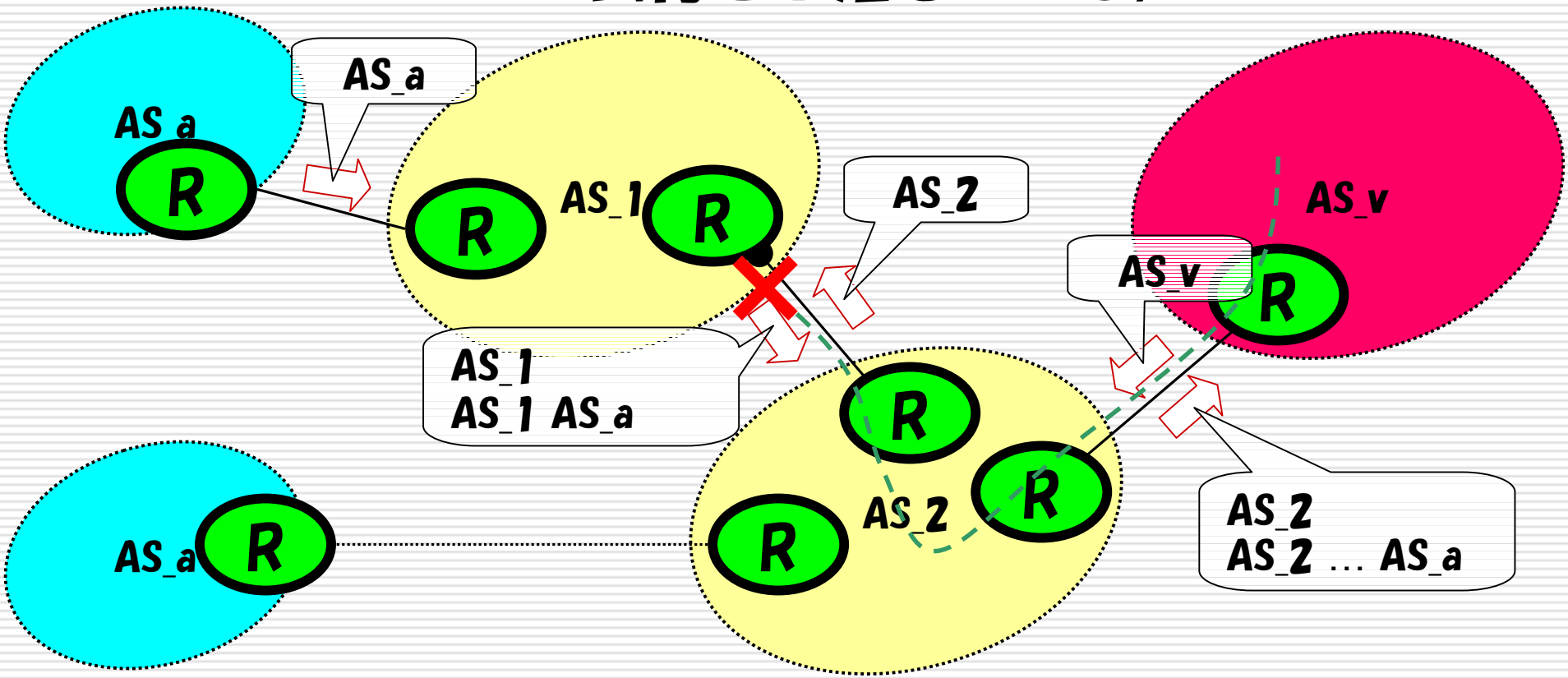
DNS Anycast とは(続き)

Server	Operator	IP Address	AS#	Location
A	VeriSign Global Registry Services	198.41.0.4	19836	Dulles VA
B	Information Sciences Institute	IPv4: 192.228.79.201 IPv6: 2001:478:65::53	tba	Marina Del Rey CA
C	Cogent Communications	192.33.4.12	2149	Herndon VA; Los Angeles; New York City; Chicago
D	University of Maryland	128.8.10.90	27	College Park MD
E	NASA Ames Research Center	192.203.230.10	297	Mountain View CA
F	Internet Systems Consortium, Inc.	IPv4: 192.5.5.241 IPv6: 2001:500::1035	3557	Ottawa; Palo Alto; San Jose CA; New York City; San Francisco; Madrid; Hong Kong; Los Angeles; Rome; Auckland; Sao Paulo; Beijing; Seoul; Moscow; Taipei; Dubai; Paris; Singapore; Brisbane; Toronto; Monterrey; Lisbon; Johannesburg; Tel Aviv; Jakarta; Munich
G	U.S. DOD Network Information Center	192.112.36.4	568	Vienna VA
H	U.S. Army Research Lab	IPv4: 128.63.2.53 IPv6: 2001:500:1::803f:235	13	Aberdeen MD
I	Autonomica/NORDUnet	192.36.148.17	29216	Stockholm; Helsinki; Milan; London; Geneva; Amsterdam; Oslo; Bangkok; Hong Kong; Brussels; Frankfurt; Bucharest; Ankara; Chicago; Washington DC; Tokyo; Kuala Lumpur
J	VeriSign Global Registry Services	192.58.128.30	26415	Dulles VA (2 locations); Mountain View CA; Seattle WA; Amsterdam; Atlanta GA; Los Angeles CA; Miami; Stockholm; London; Tokyo; Seoul; Singapore; Sterling VA (2 locations, standby)
K	Reseaux IP Europeens -Network Coordination Centre	IPv4: 193.0.14.129 IPv6: 2001:7fd::1	25152	London(UK); Amsterdam(NL); Frankfurt(DE); Athens(GR); Doha(QA); Milan(IT)
L	Internet Corporation for Assigned Names and Numbers	198.32.64.12	20144	Los Angeles
M	WIDE Project	IPv4: 202.12.27.33 IPv6: 2001:dc3::35	7500	Tokyo; Seoul(KR); Paris(FR)

uRPF と DNS anycast は仲良しか？

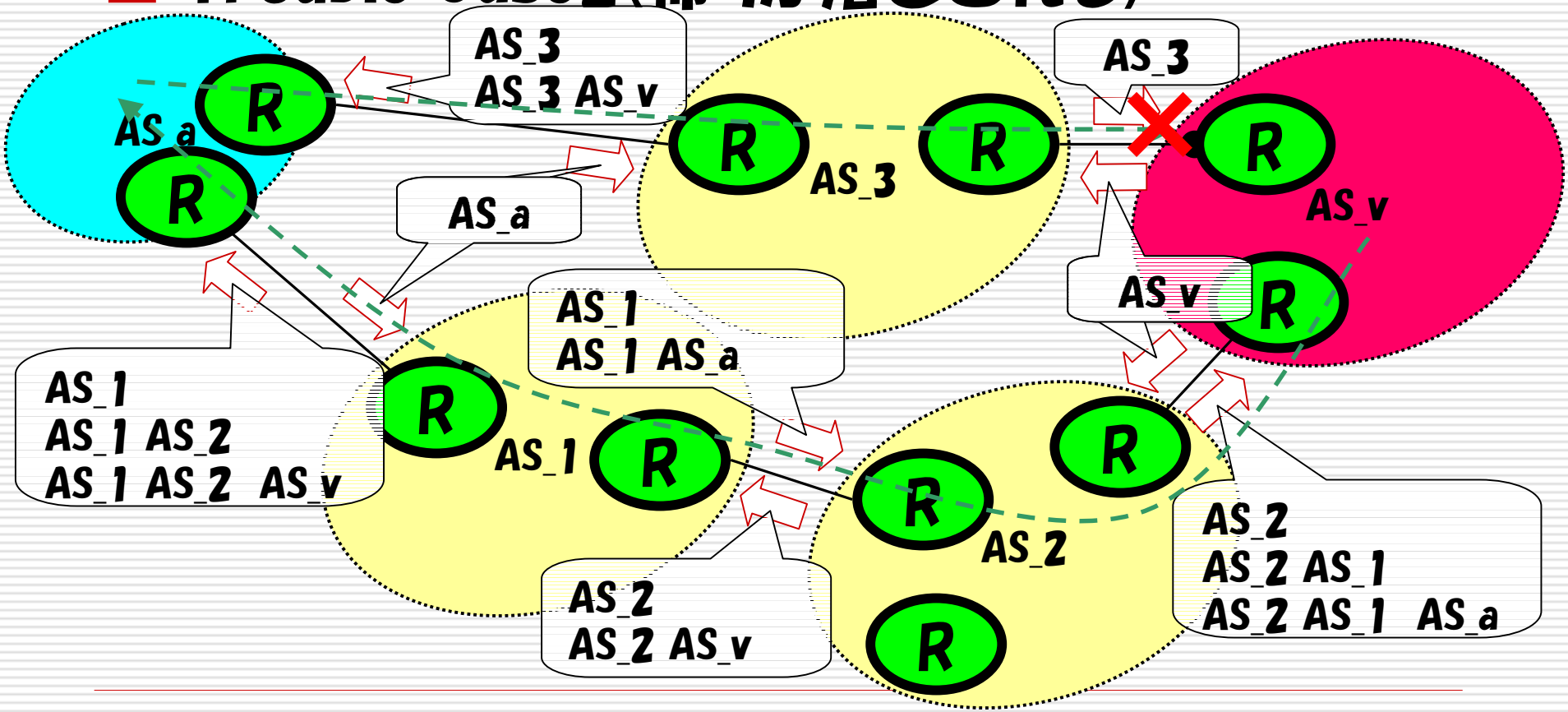
uRPF vs BGP Anycast 問題

- Trouble Case1(行きが落とされる)

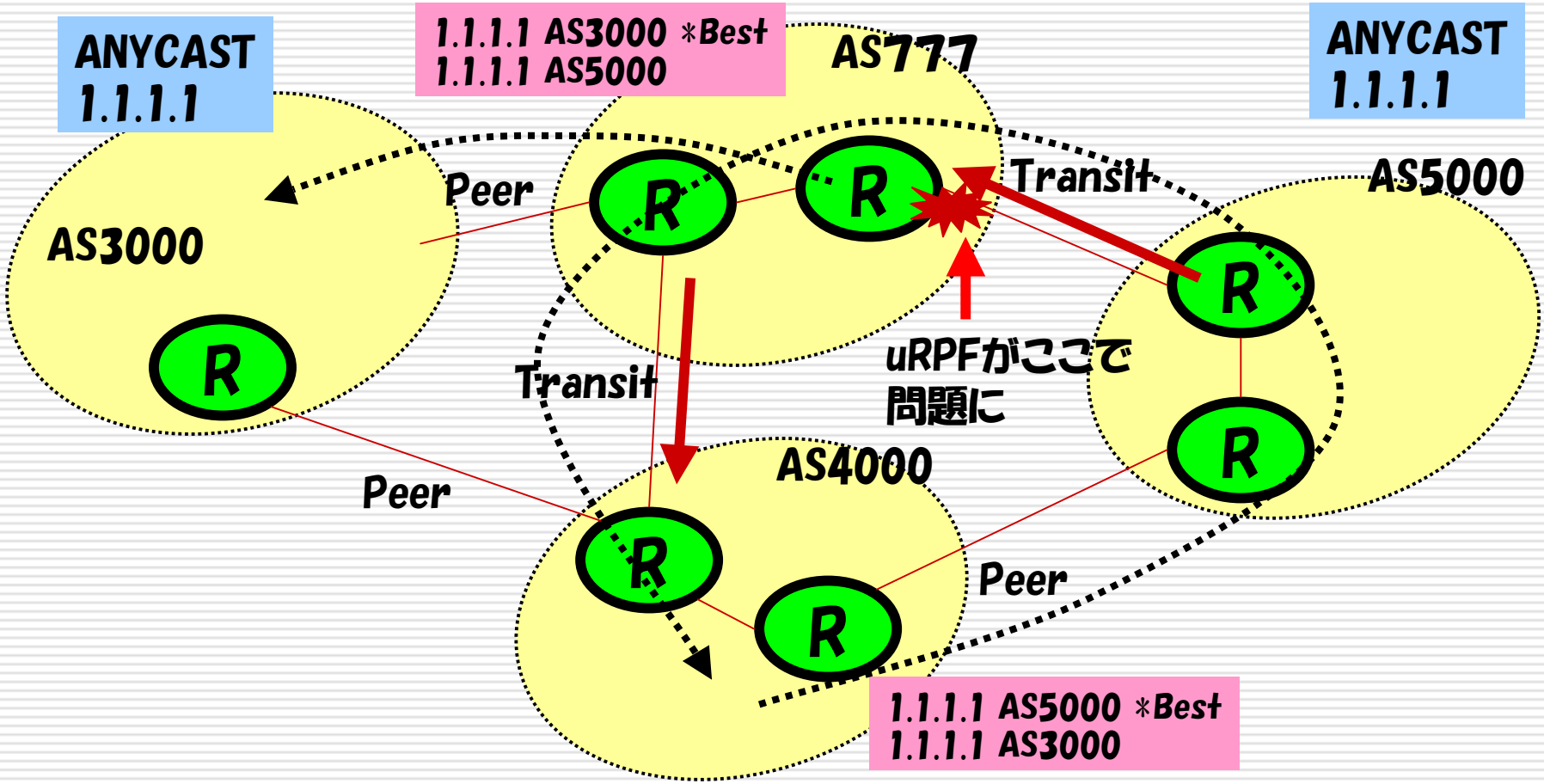


uRPF vs BGP Anycast 問題

- Trouble Case2(帰りが落とされる)

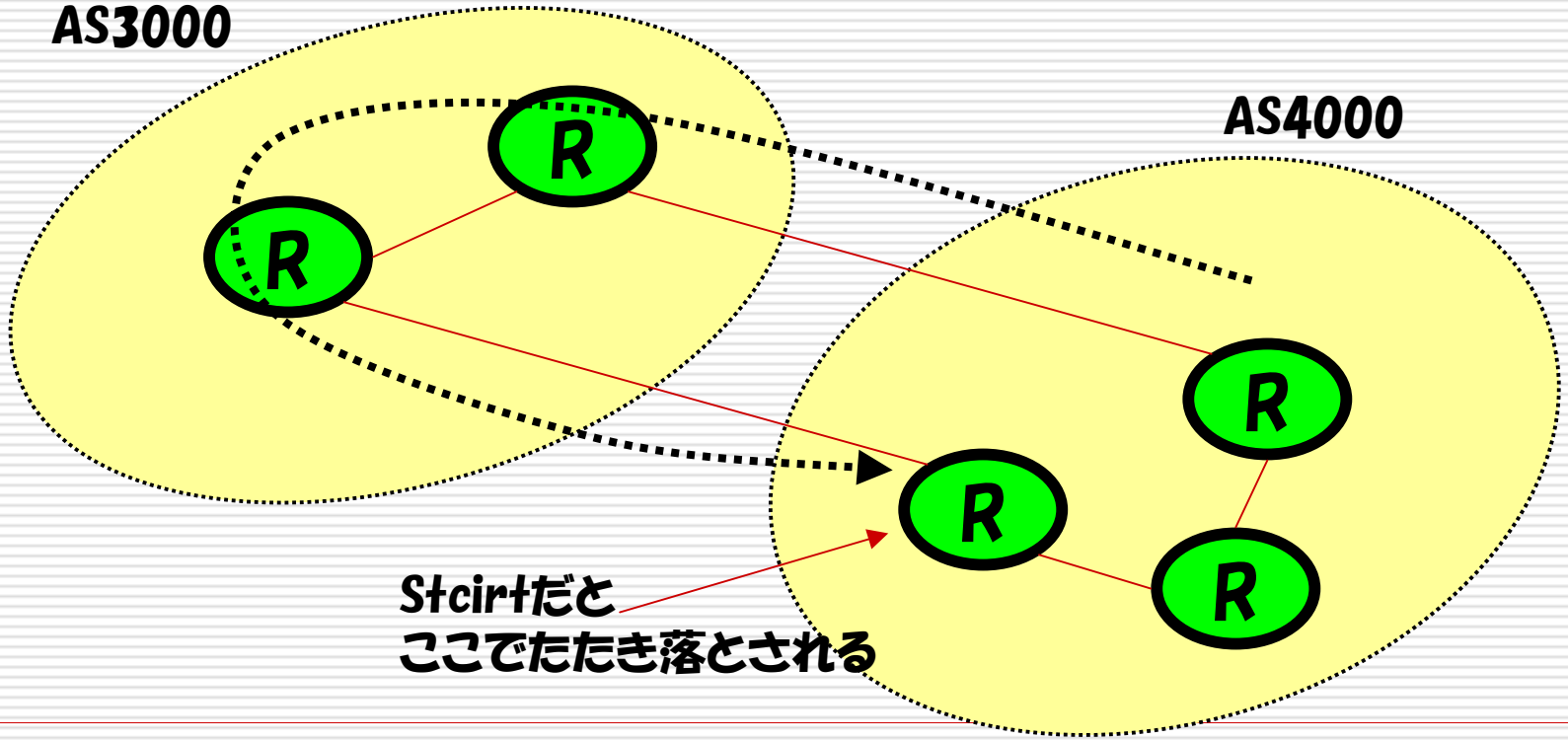


uRPF vs BGP Anycast 問題 その1



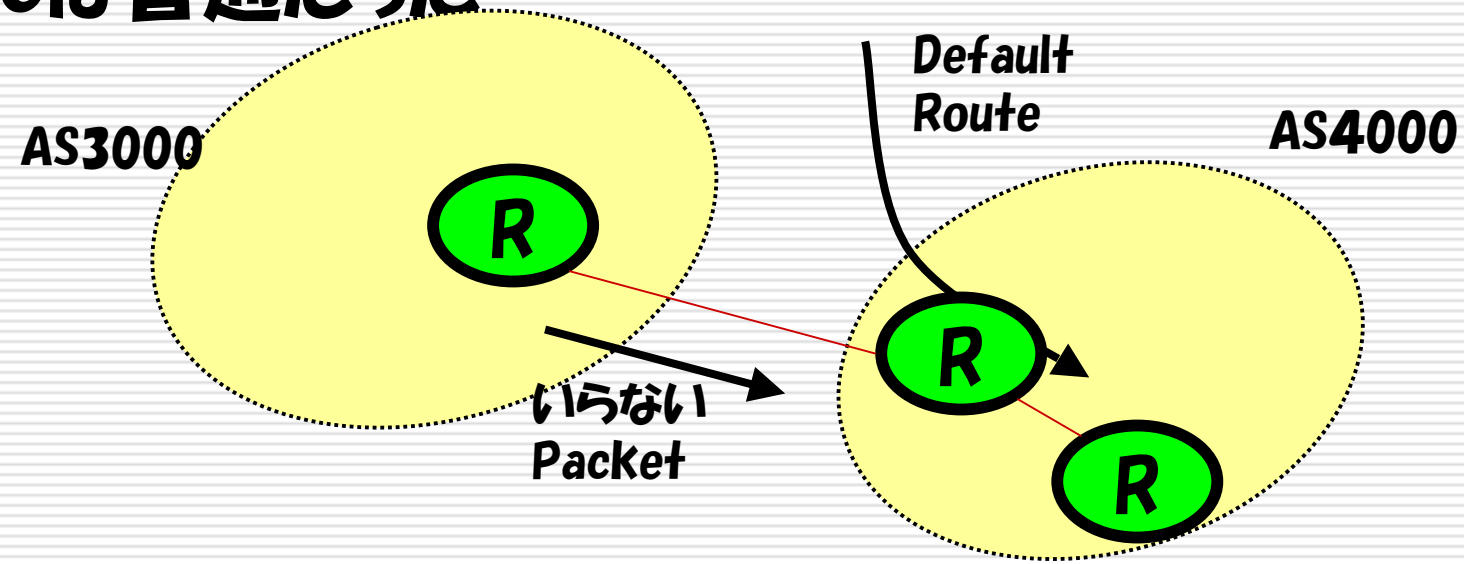
uRPF vs BGP Anycast 問題 その2

- 行きと帰りが異なる場合



uRPF vs BGP Anycast 問題 その3

- LooseでもDefaultルートはStrictなJの実装
  - 通って欲しくないのに、通ってしまう！
- Cは普通だった



考察

考察

- Multihomeで発生する問題のひきずっているのに過ぎない？
  - MultihomeでもuRPFを使うと問題は発生するものだ⇒本当でしょうか？
  - 確かにMultihomeで発生する問題をそのまま引き継いでいるであろう
  - さらにAnycastはネットワークの実体が異なるので問題を複雑にしている

考察(続き)

- 問題は起きないのか？
  - 実際には発生するだろう！
  - すでに発生したとおぼしき現象も観測済
- 気にしなくていいのか？
  - YESの立場
    - n/13の一部で発生する問題にしかすぎない
    - ルートサーバについてはどこかにたどり着ければOK
  - NOの立場
    - 13個あるすべてのルートサーバにたどりつけるべき
    - トラブルの当事者には解決できない問題である
- もっと恐ろしい問題に気づいてしまいました
  - Alternate Rootが簡単にできてしまいます！！！
  - Root-Servers Hijacking!

考察(続き)

- 思いつき
  - IPv4のuRPFではroot-severのアドレスを除外する
    - たかだかv4のホストアドレスが13個
    - めったに変わらない
  - しかし、
    - きっとルートサーバのソースアドレスを持つDoSが流行るに違いない
    - それでもいいか？？？

おわりに

- さて、この Topic は私の杞憂に過ぎないのでしょうか？
- 問題があるとすれば、今後どのようにすればいいのかな？