Why is Securing the Internet's Routing System so Hard?

Geoff Huston
APNIC
January 2019

Internet Issues

Why are some issues so challenging to solve, while others seem to be effortless?

Why was the IPv4 Internet an unintended runaway success in the 90’s, yet IPv6 has been a protracted exercise in industry-wide indecision?

What makes a problem "hard"?

It might be **technically challenging**: While we understand what we might want that does not mean we know how to construct a solution

It might be **economically hard**: The costs of a solution are not directly borne by the potential beneficiaries of deploying the solution

It might be **motivated by risk mitigation**: We are notorious for undervaluing future risk!

Internet Successes:

- IPv4 (and datagram packet switching)
- Network Address Translators (perversely!)
- TCP evolution and adaptation
- DNS scaling
- Web Security (possibly)
- Content Distribution Systems
- Streaming Services

What are Internet success factors?

- Piecemeal deployment without the requirement for central orchestration
- Competitive advantages to early adopters
- Economies of scale as adopter numbers increase
- Alignment of common benefit with individual benefit

Failure Factors

- Orchestrated actions
- Technologies that require universal or near universal adoption
- Where there are common benefits but not necessarily individual benefits
- Where there is no clear early adopter advantage

Internet ~~Non-Successes~~ Failures!

- SPAM
- DDOS defence
- BCP 38 deployment
- Secure end systems
- Secure networks
- Internet of Things
- IPv6 adoption (well maybe this will change soon!)

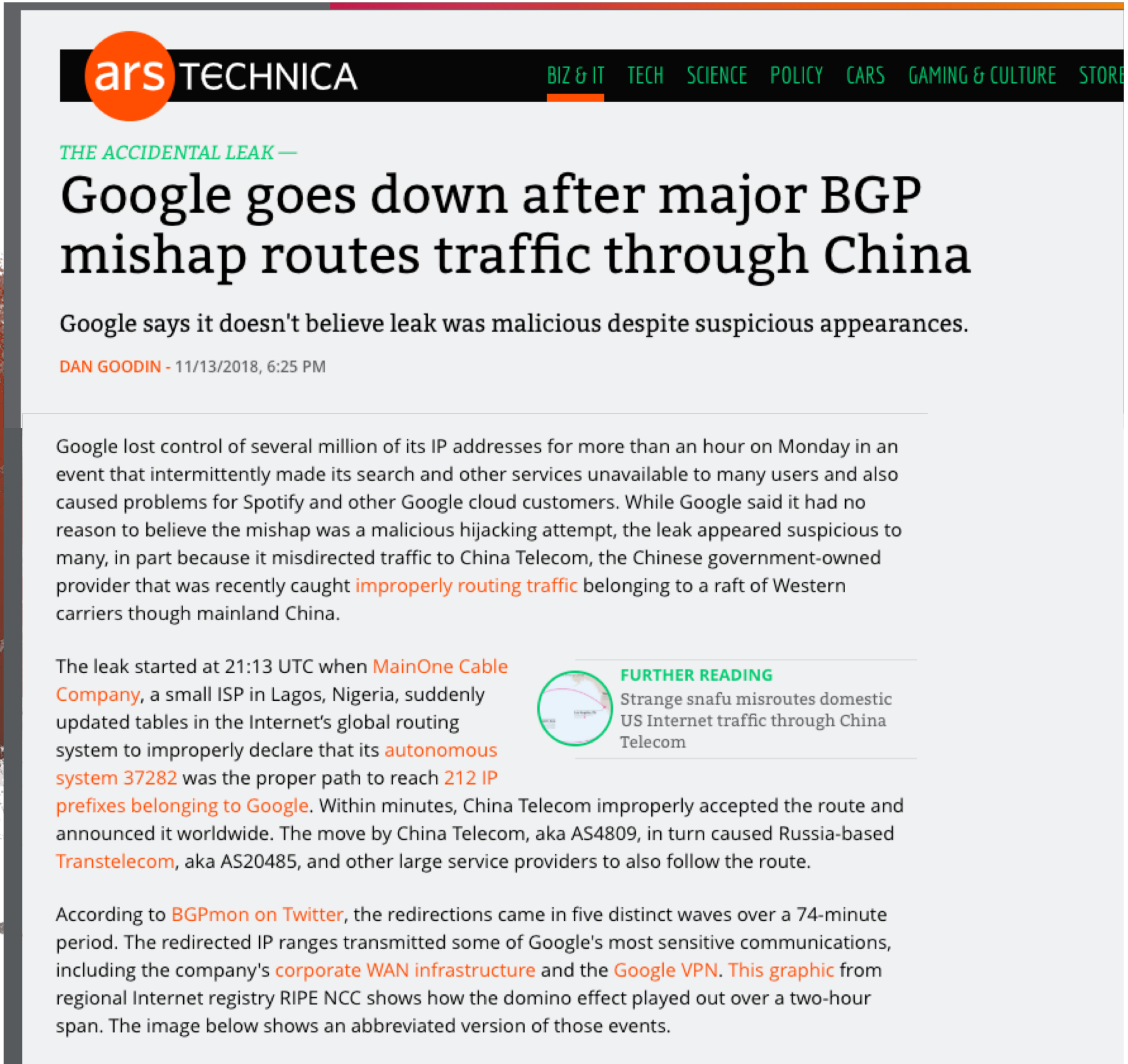
Internet ~~Non-Successes~~ Failures!

- SPAM
- DDOS defence
- BCP 38 deployment
- Secure end systems
- Secure networks
- Internet of Things
- IPv6 adoption (well maybe this will change soon!)

Lets look at this in more detail

Why is Secure Routing so hard?

Why do we keep seeing these headlines?



THE ACCIDENTAL LEAK —

Google goes down after major BGP mishap routes traffic through China

Google says it doesn't believe leak was malicious despite suspicious appearances.

DAN GOODIN - 11/13/2018, 6:25 PM

Google lost control of several million of its IP addresses for more than an hour on Monday in an event that intermittently made its search and other services unavailable to many users and also caused problems for Spotify and other Google cloud customers. While Google said it had no reason to believe the mishap was a malicious hijacking attempt, the leak appeared suspicious to many, in part because it misdirected traffic to China Telecom, the Chinese government-owned provider that was recently caught improperly routing traffic belonging to a raft of Western carriers though mainland China.

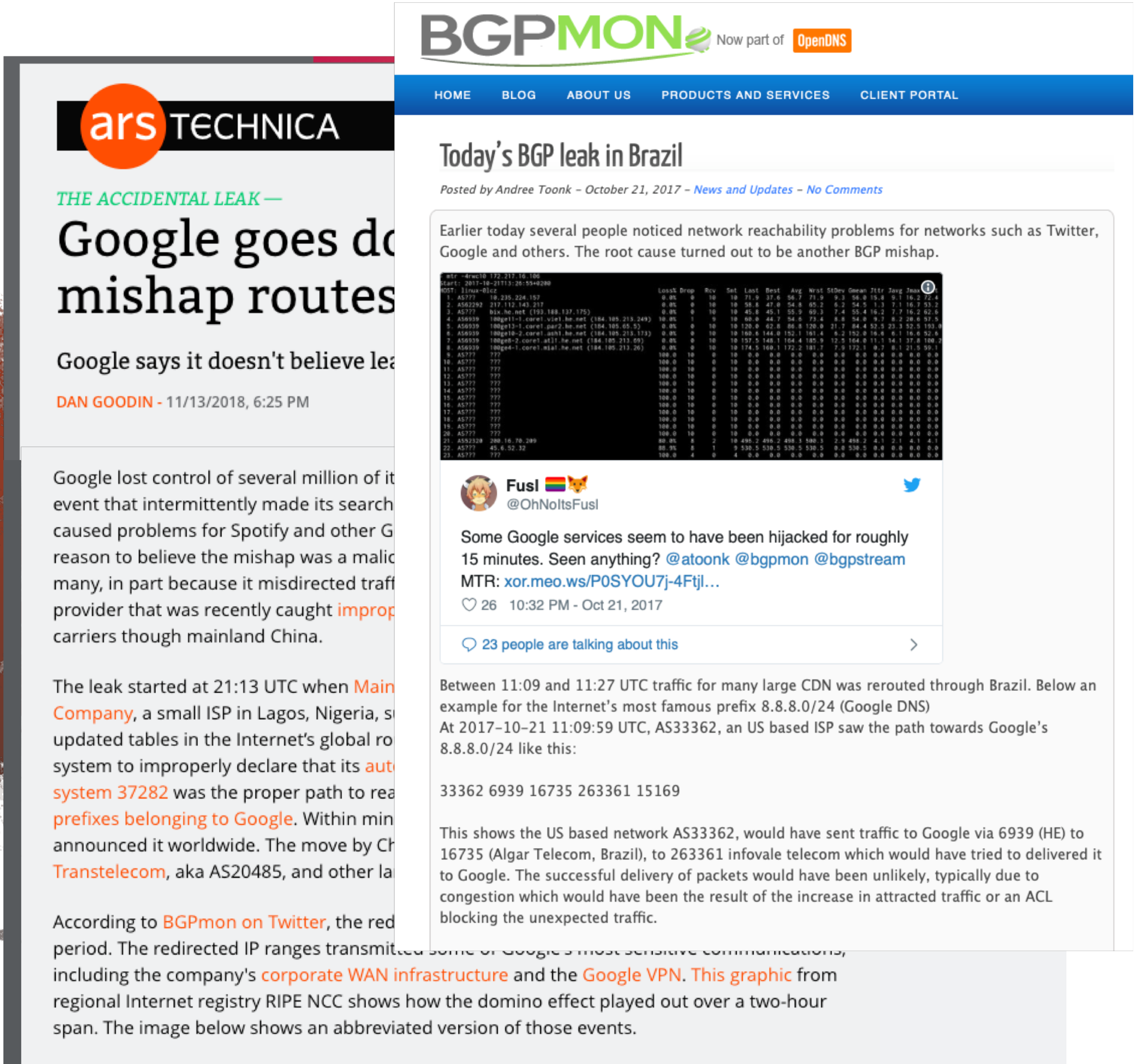
The leak started at 21:13 UTC when MainOne Cable Company, a small ISP in Lagos, Nigeria, suddenly updated tables in the Internet's global routing system to improperly declare that its autonomous system 37282 was the proper path to reach 212 IP prefixes belonging to Google. Within minutes, China Telecom improperly accepted the route and announced it worldwide. The move by China Telecom, aka AS4809, in turn caused Russia-based Transtelecom, aka AS20485, and other large service providers to also follow the route.

FURTHER READING

Strange snafu misroutes domestic US Internet traffic through China Telecom

According to BGPmon on Twitter, the redirections came in five distinct waves over a 74-minute period. The redirected IP ranges transmitted some of Google's most sensitive communications, including the company's corporate WAN infrastructure and the Google VPN. This graphic from regional Internet registry RIPE NCC shows how the domino effect played out over a two-hour span. The image below shows an abbreviated version of those events.

Why do we keep seeing these headlines?



ars TECHNICA

THE ACCIDENTAL LEAK —

Google goes do mishap routes

Google says it doesn't believe lea

DAN GOODIN - 11/13/2018, 6:25 PM

Google lost control of several million of it event that intermittently made its search caused problems for Spotify and other G reason to believe the mishap was a malic many, in part because it misdirected traff provider that was recently caught improp carriers though mainland China.

The leak started at 21:13 UTC when Main Company, a small ISP in Lagos, Nigeria, s updated tables in the Internet's global ro system to improperly declare that its aut system 37282 was the proper path to rea prefixes belonging to Google. Within min announced it worldwide. The move by Ch Transtelecom, aka AS20485, and other la

According to BGPmon on Twitter, the red period. The redirected IP ranges transmitt some of Google's most sensitive communications, including the company's corporate WAN infrastructure and the Google VPN. This graphic from regional Internet registry RIPE NCC shows how the domino effect played out over a two-hour span. The image below shows an abbreviated version of those events.

BGPMON Now part of OpenDNS

Today's BGP leak in Brazil

Posted by Andree Toonk – October 21, 2017 – News and Updates – No Comments

Earlier today several people noticed network reachability problems for networks such as Twitter, Google and others. The root cause turned out to be another BGP mishap.

```
$ mtr -4rwc10 172.217.16.106
Start: 2017-10-21T13:26:55+0200
HOST: linux-0lcz                                        Loss% Drop  Rcv  Snt  Last  Best   Avg  Wrst StDev Gmean Jttr Javg Jmax Jint
  1. AS???    10.235.224.157                             0.0%    0   10   10  71.9  37.6  56.7  71.9   9.3  56.0 15.8  9.1 16.2 72.4
  2. AS62292  217.112.143.217                            0.0%    0   10   10  58.8  47.0  54.8  65.2   6.2  54.5  1.3  7.1 16.7 53.2
  3. AS???    bix.he.net (193.188.137.175)               0.0%    0   10   10  45.8  45.1  55.9  69.3   7.4  55.4 16.2  7.7 16.2 62.6
  4. AS6939   100ge11-1.core1.vie1.he.net (184.105.213.249) 10.0% 1   9   10  60.0  44.7  54.6  73.4   8.8  54.0  9.7  8.2 20.6 57.5
  5. AS6939   100ge13-1.core1.par2.he.net (184.105.65.5)  0.0%    0   10   10 120.0  62.8  86.8 120.0  21.7  84.4 52.5 23.3 52.5 193.0
  6. AS6939   100ge10-2.core1.ash1.he.net (184.105.213.173) 0.0%  0   10   10 160.6 144.0 152.1 161.4   6.2 152.0 16.6  6.1 16.6 52.6
  7. AS6939   100ge8-2.core1.atl1.he.net (184.105.213.69) 0.0%    0   10   10 157.5 148.1 164.4 185.9  12.5 164.0 11.1 14.1 37.8 100.2
  8. AS6939   100ge4-1.core1.mia1.he.net (184.105.213.26) 0.0%    0   10   10 174.5 160.1 172.2 181.7   7.9 172.1  0.7  8.1 21.5 59.1
  9. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 10. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 11. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 12. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 13. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 14. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 15. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 16. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 17. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 18. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 19. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 20. AS???    ???                                       100.0   10    0   10   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
 21. AS52320  200.16.70.209                             80.0%    8    2   10 496.2 496.2 498.3 500.3   2.9 498.2  4.1  2.1  4.1  4.1
 22. AS???    45.6.52.32                                88.9%    8    1    9 530.5 530.5 530.5 530.5   0.0 530.5  0.0  0.0  0.0  0.0
 23. AS???    ???                                       100.0    4    0    4   0.0   0.0   0.0   0.0   0.0   0.0  0.0  0.0  0.0  0.0
```

Fusl 🏳️‍🌈🦊
@OhNoItsFusl

Some Google services seem to have been hijacked for roughly 15 minutes. Seen anything? @atoonk @bgpmon @bgpstream MTR: xor.meo.ws/P0SYOU7j-4Ftjl...

♡ 26 10:32 PM - Oct 21, 2017

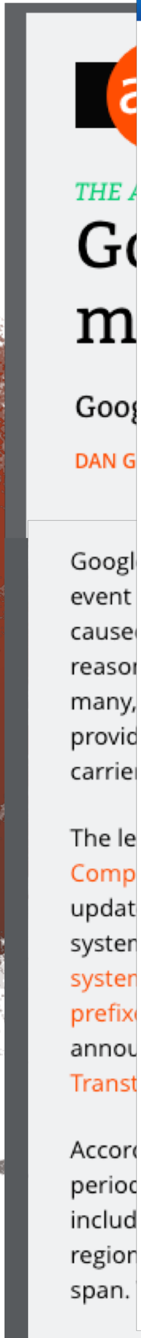
23 people are talking about this

Between 11:09 and 11:27 UTC traffic for many large CDN was rerouted through Brazil. Below an example for the Internet's most famous prefix 8.8.8.0/24 (Google DNS)
At 2017-10-21 11:09:59 UTC, AS33362, an US based ISP saw the path towards Google's 8.8.8.0/24 like this:

33362 6939 16735 263361 15169

This shows the US based network AS33362, would have sent traffic to Google via 6939 (HE) to 16735 (Algar Telecom, Brazil), to 263361 infovale telecom which would have tried to delivered it to Google. The successful delivery of packets would have been unlikely, typically due to congestion which would have been the result of the increase in attracted traffic or an ACL blocking the unexpected traffic.

Why do we keep seeing these headlines?



THE A

G

m

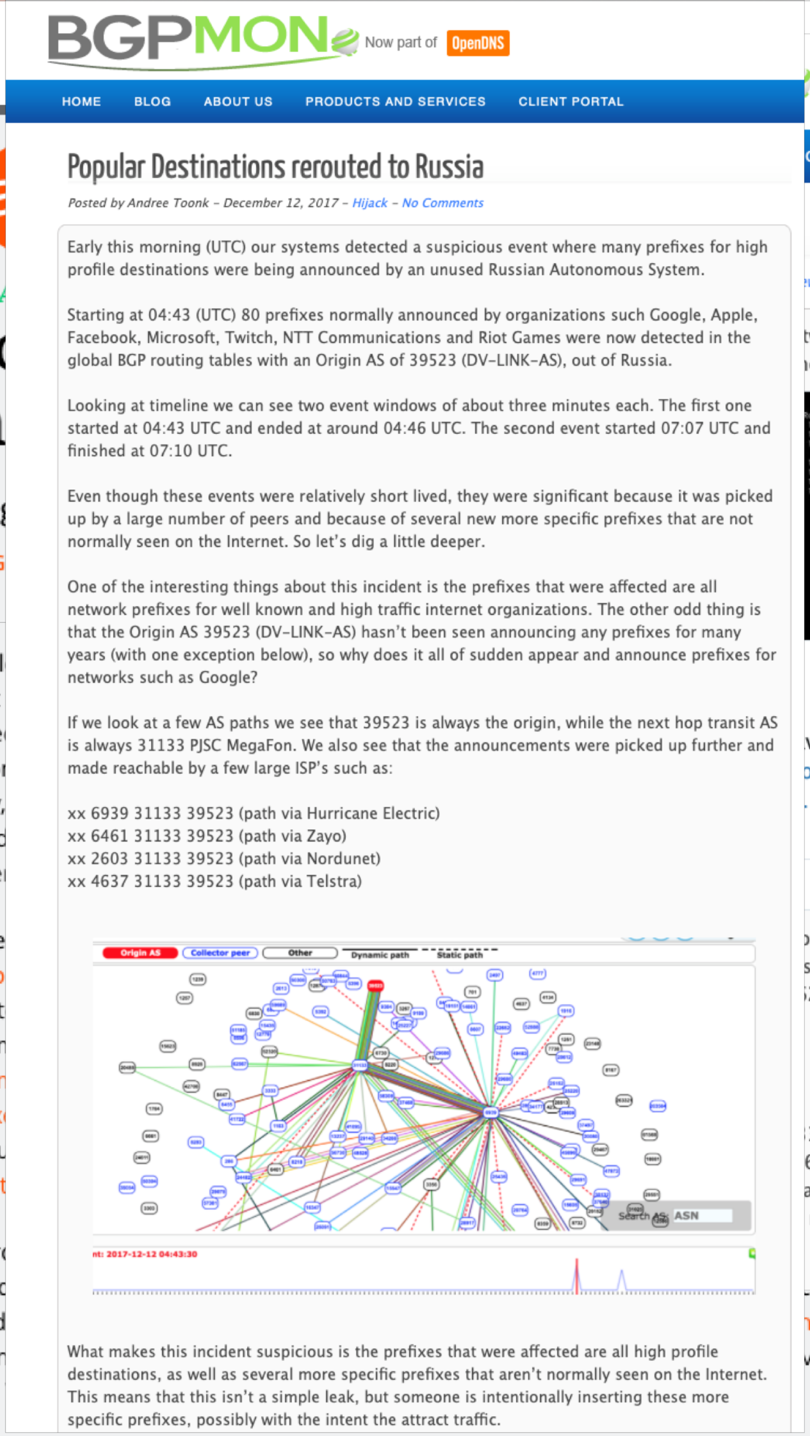
Goog

DAN G

Googl
event
cause
reasor
many,
provid
carrier

The le
Comp
updat
systen
systen
prefix
annou
Transt

Accord
period
includ
region
span.



BGPMON Now part of OpenDNS

Popular Destinations rerouted to Russia

Posted by Andree Toonk – December 12, 2017 – Hijack – No Comments

Early this morning (UTC) our systems detected a suspicious event where many prefixes for high profile destinations were being announced by an unused Russian Autonomous System.

Starting at 04:43 (UTC) 80 prefixes normally announced by organizations such Google, Apple, Facebook, Microsoft, Twitch, NTT Communications and Riot Games were now detected in the global BGP routing tables with an Origin AS of 39523 (DV-LINK-AS), out of Russia.

Looking at timeline we can see two event windows of about three minutes each. The first one started at 04:43 UTC and ended at around 04:46 UTC. The second event started 07:07 UTC and finished at 07:10 UTC.

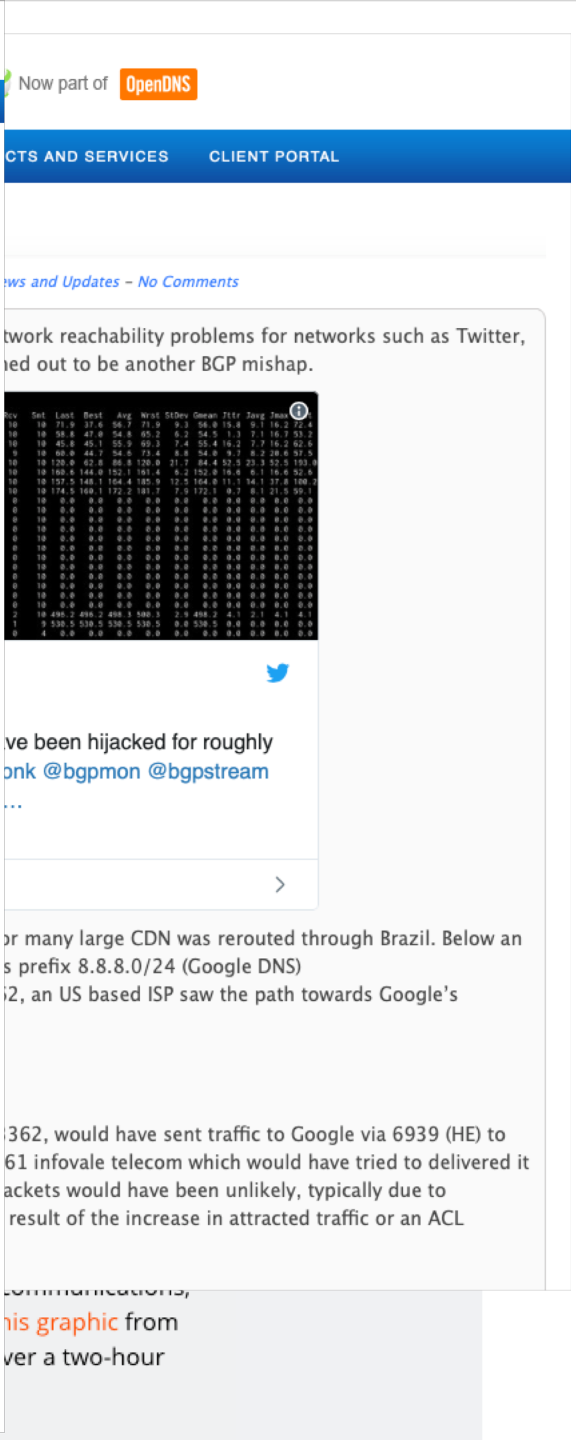
Even though these events were relatively short lived, they were significant because it was picked up by a large number of peers and because of several new more specific prefixes that are not normally seen on the Internet. So let's dig a little deeper.

One of the interesting things about this incident is the prefixes that were affected are all network prefixes for well known and high traffic internet organizations. The other odd thing is that the Origin AS 39523 (DV-LINK-AS) hasn't been seen announcing any prefixes for many years (with one exception below), so why does it all of sudden appear and announce prefixes for networks such as Google?

If we look at a few AS paths we see that 39523 is always the origin, while the next hop transit AS is always 31133 PJSC MegaFon. We also see that the announcements were picked up further and made reachable by a few large ISP's such as:

xx 6939 31133 39523 (path via Hurricane Electric)
xx 6461 31133 39523 (path via Zayo)
xx 2603 31133 39523 (path via Nordunet)
xx 4637 31133 39523 (path via Telstra)

What makes this incident suspicious is the prefixes that were affected are all high profile destinations, as well as several more specific prefixes that aren't normally seen on the Internet. This means that this isn't a simple leak, but someone is intentionally inserting these more specific prefixes, possibly with the intent the attract traffic.



Now part of OpenDNS

ews and Updates – No Comments

twork reachability problems for networks such as Twitter, ned out to be another BGP mishap.

ve been hijacked for roughly
onk @bgpmon @bgpstream
...

or many large CDN was rerouted through Brazil. Below an s prefix 8.8.8.0/24 (Google DNS) 52, an US based ISP saw the path towards Google's

362, would have sent traffic to Google via 6939 (HE) to 61 infovale telecom which would have tried to delivered it ackets would have been unlikely, typically due to result of the increase in attracted traffic or an ACL

communications,
his graphic from
ver a two-hour

Why is Secure Routing so hard?

Here is my ‘top ten’ list of why this has proved to be an extremely hard problem

1. Noone is in charge!

- There is no single authority model for the Internet
- It’s a decentralized distributed environment
- Which means there is no reference point for what is “right” in routing
- Equally there is no clear way of understanding what is “wrong”
- There is no authority to direct anyone to do anything

2. Routing is by Rumour

- We use self-learning routing protocols that discover the network’s topology
  - We tell our neighbors what we know
  - And learn from our neighbors what they know
- We assume that everyone is honest and everyone is ‘correct’
- And we find it challenging to determine if a rumour is incorrect and even hard to determine who originated it!

3. Routing is Variable

- The routing system generates a view of the network that is local to where you are
- If you and I are in different locations on the net then we will have different routing outcomes
- There is no single ‘reference’ routing outcome that we can use to compare with the local view

4. Routing is Backwards!

- Outbound advertisements influence inbound traffic
- Accepted inbound advertisements influence outbound traffic

5. Routing is a Negotiation

- Routing has two roles:
  - Topology discovery
  - Policy negotiation
- Policy is a negotiation:
  - I have local import preferences
  - You have local export preferences
- Routing attempts to negotiate a stable outcome between export and import preferences

6. Routing is not Deterministic

- The routing system will not always generate the same outcomes from the same inputs
- The negotiation process depends on the interactions between information flows and timers to perform the negotiation
- Subtle changes in the environment may produce different stable outcomes

7. There is no Evil Bit

- In security system a “bad” data time does not identify itself as bad!
- All we can do is identify “good” data
- If we identify all the good data then what’s left is bad
- But if we only identify some good data then…

8. Because risk is hard

- Taking measures to mitigate risk is like a lottery:
  - Everyone spends money to buy a ticket
  - But there is only one winner!
- But in this case the “win” is that the attack is deflected
  - And the potential victim does not even know that an attack was launched at them
- So everyone pays, but there are no visible winners!

9. Because Business

- Each network is motivated by conventional business economics
  - Each network attempts to balance factors of risk and opportunity
  - Spending resources on security must be seen to either reduce business risk or increase an enterprises competitive advantage
- Network operators under-appreciate risk
  - Which is a problem as security normally results in a system-wide outcome (i.e. there is little in the way of competitive advantage to be had here)
- But lawyers over-appreciate liabilities
  - Which places operators of critical trust infrastructure in a potentially difficult position

10. Because we don't know what we want!

- It is challenging to identify a ‘correct’ routing system
- It is far easier to detect when problems arise
- So we know what we don’t want when we see it, but that does not mean that we can recognize what we actually want
    - The absence of a recognizable ‘bad’ does not mean that its ‘good’!

Why is Securing Routing so Hard?

- Because no single entity is in charge
- Because we can’t audit BGP, as we have no standard reference route set to compare with
- Because we can’t arbitrate between conflicting BGP updates (because there is no standard reference point)
- Because there are no credentials that allow a BGP update to be compared against the original route injection (because BGP is a hop-by-hop protocol)
- Because routing is based on opaque local decisions

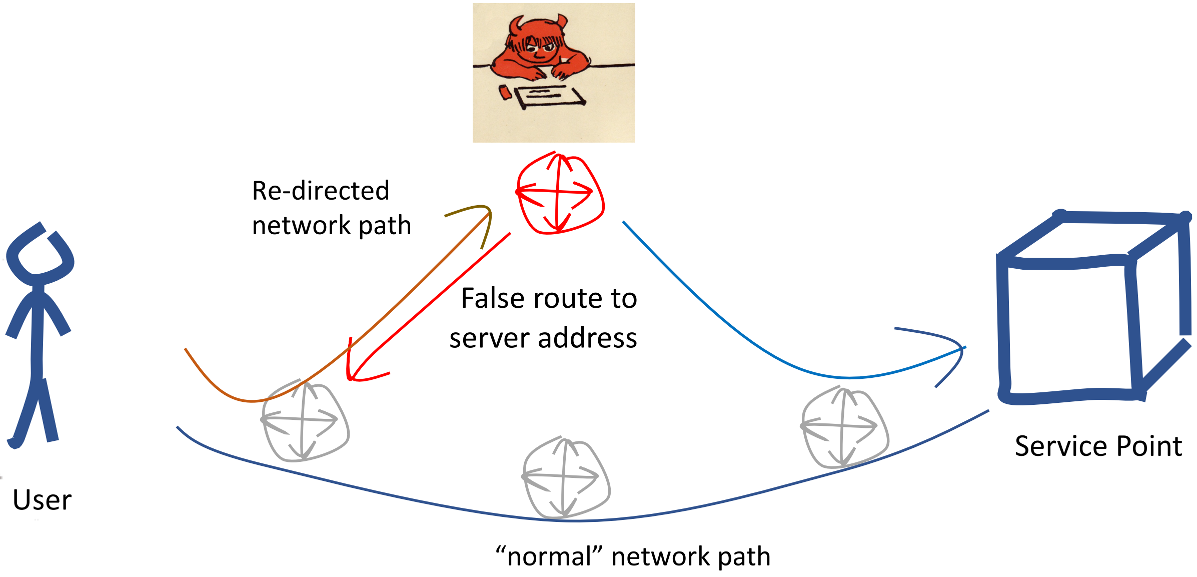
Routing Security is a Failure so far

Why should we worry?

Because it's just too easy to be bad in routing!

What's the risk?

User traffic gets diverted enabling a Man-in-the-Middle attack on a service



What's the risk?

DOS Attack

Divert the traffic to a sinkhole

- Deny users access to the site
- Crude, but effective!

What's the risk?

DNS Attacks

Divert DNS traffic to fake DNS servers and provide fake answers

- Very few domains are DNSSEC-signed and not enough resolvers perform DNSSEC validation
- So the faked answer can pass unchallenged

What's the risk?

Server Attacks

Divert TCP traffic to fake servers and provide fake answers

- Collect user credentials while shadowing the actual site

An attack vector on HTTPS…

- Let’s say you can find an online trusted CA
  - that uses the DNS as proof-of-possession of a DNS name in order to mint a domain name certificate
  - And the DNS name is not DNSSEC protected
- You can mint a fake domain name certificate by:
  - Mount a routing attack on the DNS infrastructure with a fake DNS responder
  - Answer everything correctly except for *.victim DNS challenge from the CA
  - And for the *.victim challenge queries respond with your own answer
  - Which means you can answer the CA’s DNS challenge
- Now you have a trusted fake domain name certificate

- You are now able to pull off a MITM attack on a TLS ‘protected’ service

Let's Secure Routing!

Can we devise changes to operational practices, or operational tools or routing technologies that manage the inter-domain routing system that could prevent the propagation of false or artificial routing information in the Internet?

The Ideal

We want the interdomain routing system to advertise the **correct** reachability information for “**legitimately connected”** prefixes at all times

That means that we want to **avoid**:

- promulgating reachability for bogus address prefixes
- promulgating incorrect paths for reachable prefixes
- blocking paths for legitimately connected prefixes

The Problems

- While we'd like to think we understand the provenance for each and every IP address, that is not exactly the case
- And even if we did, we have no precise knowledge as to which network has the authority to originate a route object for that address
- And even then we have no exact knowledge of the inter-domain topology of the network
- And even then we have no clear knowledge of the local policy constraints that are applied to the propagation of reachability and topology information

Which means…

- we have no clear model of “truth” to compare to the information flow that we see in the routing system
- In the absence of better information network operators just accept everything that they learn from BGP
  - Which includes its share of cloudy half-truths and lies

A Goal for Routing Security

Can we devise changes to operational practices, or operational tools or routing technologies that manage the inter-domain routing system that could resist attempts to inject false or artificial routing information in the Internet?

Objectives

1. Identify whether an address is “bogus” or not
2. Assure that the address holder has given their permission for an address to be announced into the routing system
3. Identify which AS(s) have been given this permission
4. Identify if the AS Path is consistent with the ‘correct’ operation of BGP
5. Identify if the AS Path is consistent with the routing policies of the each of the Ases
6. Identify when routing information is being ‘incorrectly’ withheld

What Data would we like for Prevention?

- An (impossible) ideal data set is the “reference set” that describes a ‘correct’ route object set that should be visible at any vantage point in the network
    - And access to a set of credentials that support any such attestation of “correctness”
- As a compromise we could settle for a reference set that describes a ‘stable’ route object set that should be visible at any vantage point in the network

Internet Route Registries

- First used in the early 1990’s as the Route Arbiter Database (RADB) as part of the NSFNET program
- Describes route origination and inter-AS routing policies
- An explicit declaration of intent in routing
- Route Registries could be used as filter to be applied to BGP announcements, filtering out route advertisements that are note described in the route registry
    - Primary value in preventing neighbor route leaks
    - Can be used to prevent hijacks

Route Registry Issues

- Poor Authority Model (or the lack of one)
  - How can a user know that a RR entry is genuine and current?
  - How can a user know that a RR entry is maintained by an entity who is the authoritative “owner” of an IP address or ASN?
  - How can a user tell the difference between a current RR entry and a lapsed historical RR entry?
- Too many RRs
  - If two different RRs contain conflicting information, what are users meant to do?
- Incomplete Data
  - If a route is not described in a Route Registry is it just the registry that is missing data or is the route itself invalid?
- Scaling issues
  - No realistic way to apply IRR filters to upstreams
- RPSL got too geeky!
  - The Route Policy Language used by Route Registries got overly expressive and complex

What's missing with IRRs?

- If we want to improve the usefulness of route registries we probably need to add digital signatures with an authority model
  - The signatures can provide currency and authenticity
  - The authority model can allow RR entries to be seen as explicit authorities or permissions from address holders to network operators and from network operators to other networks
- So lets look at a *testable* authority model

PKI-derived Authority Models

Digital Certificates can be used to convey an authority or permission from one party to another

- Start with IP address certificates (X.509 public key certificates that include a set of IP addresses and ASNs)
- If an address holder A allocates an IP address to B then the action can be 'witnessed' by A issuing a public key certificate for the allocated address coupled with B's public key , all signed using A's private key. By virtue of the allocation B is now able to exercise authority over the IP address, and sign authorities using B's private key that any third party can validate
- The PKI follows the address allocation framework. Trust originates from the anchor RIR data, and then follows allocations in a verifiable path from the registry to the address holder

RFC3779: X.509 Public Key Certificates for IP addresses and AS Numbers

- An X.509 Public key certificate that includes a set of IP addresses and AS numbers
- If a certificate can be validated against a trust anchor then it indicates that:
  - The IP addresses and/or AS numbers have been validly allocated
  - The holder of the subject key pair is the current holder of the IP addresses and/or AS numbers
  - Attestations validly signed using this key can be considered as genuine authorities that cannot be repudiated
- This is the foundation of the current work in routing security

Route Origination Authority

- An address holder can convey a ‘permission’ for an AS to originate a BGP route for the address by signing a permission authority (ROA) using a signing key associated with a valid public key address certificate
- This authority:
  - can be validated by any interested party
  - is dated, so currency is known
  - cannot be repudiated

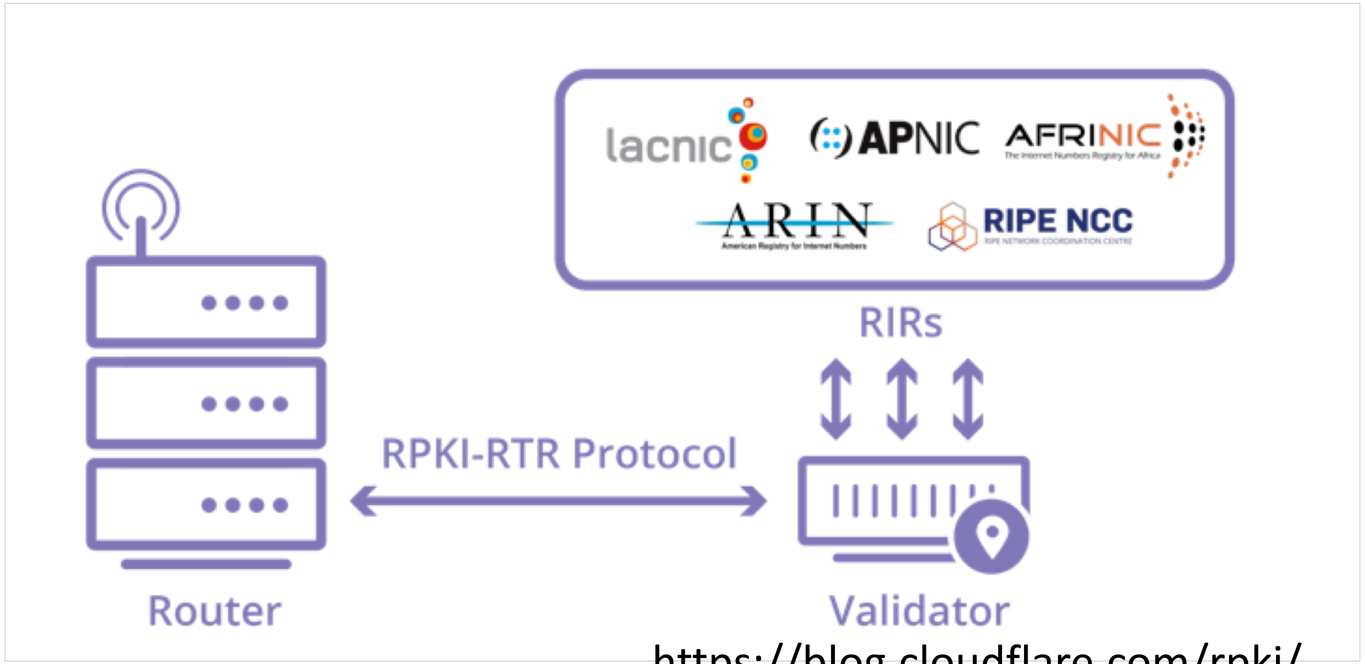
If we all used ROAs then:

1. ✓ Identify whether an address is “bogus” or not
2. ✓ Assure that the address holder has given their permission for an address to be announced into the routing system
3. ✓ Identify which AS(s) have been given this permission
4. Identify if the AS Path is consistent with the ‘correct’ operation of BGP
5. Identify if the AS Path is consistent with the routing policies of the each of the Ases
6. Identify when routing information is being ‘incorrectly’ withheld

So let's use the authority model to secure BGP route origination!

Address holders should generate ROAs

Network operators should use ROAs to filter updates


lacnic
APNIC
AFRINIC
The Internet Numbers Registry for Africa
ARIN
American Registry for Internet Numbers
RIPE NCC
RIPE NETWORK COORDINATION CENTRE
RIRs
RPKI-RTR Protocol
Router
Validator

https://blog.cloudflare.com/rpki/

From ROAs to a fully secure BGP

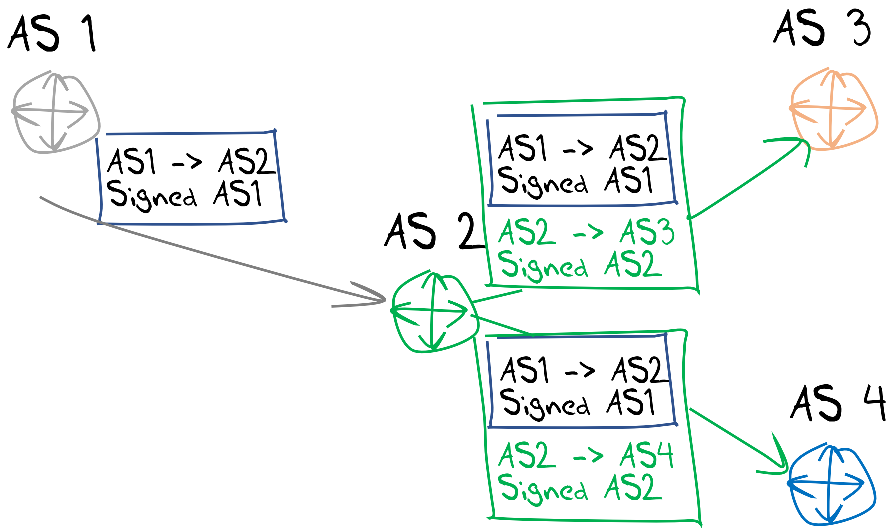
ROAs are good, but probably not enough to stop a determined routing attacker

- The attacker simply needs to replicate the BGP origination in the AS path to be accepted as "good"

So we really need to secure the BGP AS Path as well

We can do this with RPKI certs!

- Every eBGP speaker has a key that is certified by the AS
- When an update is passed to a neighbor AS, the router signs across the existing AS Path signature and the neighbor AS
- A BGPSEC speaker validates a received update by checking that
  - there is a current ROA to describe the address and origin AS
  - The received AS Path can be validated as a sequence of sign-over-sign operations by the AS keys



But ASPath protection is hard…

- BGPSEC cannot cope with partial adoption
  - It cannot jump across non-participating networks
- It has a high crypto overhead for session restarts
- It does not define how to promulgate the collection of certificates required to validate the digital signatures
- It does not necessarily identify and prevent route leaks

- Which means that BGPSEC is not looking like its going to be deployed everywhere

What's going wrong?

The economics of this situation work against it

- Apparently there are inadequate commercial drivers to undertake extensive informed route monitoring that would enable hijack suppression at source
- Probably because integrity of common infrastructure is everyone’s problem which in turn quickly becomes nobody’s problem
- And we have no ‘forcing’ authority to compel network operators

  (maybe we’re getting such a forcing authority imposed upon us, but that’s another story!)

Where to from here?

- We are pretty convinced about the value of RPKI certificates and digital signatures
  - Because we really have nothing better to offer in their place
- But the AS Path protection elements of BGPSEC are a critical problem!
- In the IETF we are working on approaches that address the issues with BGPSEC and AS Path protection
  - But that effort could take years
  - And there is no guarantee of success!

Where are we heading?

- The problem is not going to go away
- So we need to look at other ways to secure the propagation of routing information:
  - What if we decoupled origination, topology and policy validation?
  - What could we gain by using deliberate efforts at asymmetric partial adoption?
    - What’s more important in routing security: client routes or server routes?
    - i.e. should we concentrate on IXPs and CDN routes as points of active route policing?
  - Will open market disciplines lead us to a secure Internet environment or are we necessarily looking at regulatory imposts to force universal adoption?

What can you do today?

“Don’t let the perfect be the enemy of the good!”

- Don’t let waiting for a complete routing security framework stop you from doing something practical and helpful today

What can you do today?

We should take steps to make routing attacks easier to detect and easier to deflect

- Generating ROAs can help
  - Maybe they won’t help a lot today, but as more networks filter on ROAs then they will be more effective to protect against simple address hijacking
- Route Registry objects can help
  - Again this is not a complete answer, but its better than nothing
- You should filter your customers
  - Filter customer routing updates according to ROAs and IRR profiles
- Consider signing up to MANRS
  - https://www.manrs.org/

Thanks!

Questions?

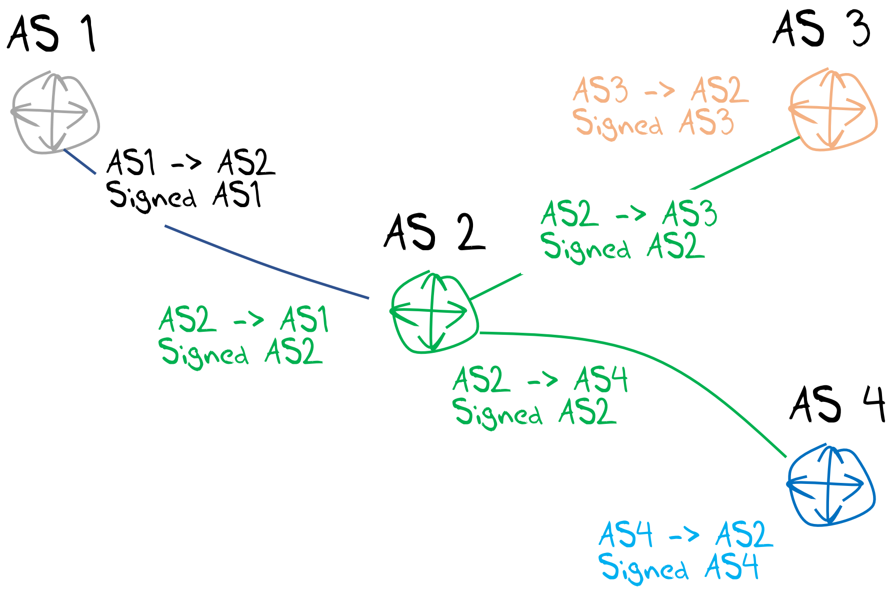
Additional Material

1. soBGP
2. General Comments
3. Opinions

1. soBGP: an alternative to BGPSEC

- Instead of the high overhead of AS Path validation we can look at secure origin BGP (soBGP) from 2003
- soBGP looked at the AS Path as a topology vector composed of a number of paired AS adjacencies
  - An AS publishes a signed adjacency attestation for all of its neighbors
  - If a signing AS appeared in an AS Path then its neighbors in the AS Path must also by described in the adjacency attestion
- This replaces strict AS Path **Validation** with AS Path **Plausibility**

soBGP and AS Adjacencies



AS Path Processing using AS Adjacency 'hints'

AS1 -> AS2 -> AS3	plausible
AS1-> AS3 -> AS2	implausible
AS1-> AS2 -> AS3 -> AS4	implausible

soBGP compared to BGPSEC

- Lower crypto overhead
- Can be used in scenarios of partial adoption
- Does not prevent a network from learning false information, but prevents a network being used in a falsified AS path
  - Unless you also include the AS's peers
  - And so on
  - Incremental deployment generates incremental benefit
- Can include directionality in the AS adjacency attestation

2. Generic Concerns

Is a ***trust hierarchy*** the best approach to use?

- The concern here is **concentration of vulnerability**

  If validation of routing information is dependent on the availability and validity of a single root trust anchor then what happens when this single digital artifact is attacked?

- But is there a viable alternative approach?

  Can you successfully incorporate robust diversity of authentication of security credentials into a supposedly highly resilient secure trust framework?

This is a very challenging question about the nature of trust in a diverse networked environment!

Web trust – 1,500 Cas

DNSSEC trust – 1 key

which is 'better'?

Concerns

A major issue here is that of ***partial use and deployment***

- This security mechanism has to cope with partial deployment in the routing system
  - The basic conventional approach of “what is not certified and proved as good must be bad” will not work in a partial deployment scenario
- In BGP we need to think about both origination and the AS Path of a route object in a partial deployed environment
  - AS path validation is challenging indeed in an environment of piecemeal use of secure credentials, as the mechanism cannot tunnel from one BGPsec “island” to the next “island”
- A partially secured environment may incur a combination of high incremental cost with only marginal net benefit to those deploying BGPsec

Concerns

Is certification the ***only way*** to achieve useful outcomes in securing routing?

- Is this form of augmentation to BGP to enforce "protocol payload correctness" over-engineered, and does it rely on impractical models of universal adoption?
- Can various forms of *routing anomaly detectors* adequately detect the most prevalent forms of typos and deliberate lies in routing with a far lower overhead, and allow for unilateral detection of routing anomalies?
- Or are such anomaly detectors yet another instance of "cheap security pantomime" that offer a thinly veiled placebo of apparent security that is easily circumvented or fooled by determined malicious attack?

3. My Opinions!

My personal view of a design compromise for secure BGP:

- Improve the robustness of RPKI certs by altering the cert validation algorithm
- Flatten the certificate hierarchy by using a single CA and distributed RAs
- Place origination signatures, ROAs and certs into the BGP protocol updates as opaque attributes
- Use AS Adjacency attestations
- Place AS Adjacency attestations into BGP protocol updates as opaque attributes
- Exploit the use of TCP in BGP to never resend already sent certs
- Flatten parts of the CA hierarchy by using RAs rather than CA delegations
- Reduce OOB credential distribution to TA material
    - For which you can use the DNS and DNSSEC if you really want to put all your eggs in one basket!

Like all the other approaches, this represents a particular set of compromises about speed, complexity, cost, deployment characteristics and robustness – it has it's weaknesses in terms of comprehensive robustness, but it attempts to reduce the number of distinct moving parts