BREAKING HTTPS WITH BGP HIJACKING

Artyom Gavrichenkov
Qrator Labs
ag@qrator.net

QRATOR

BGP Hijacking at a glance

- In the Internet, routing announcements are accepted without almost any validation
- This opens a possibility for a network operator to announce someone else's network prefixes without permission

BGP Hijacking, a problem

- In the Internet, routing announcements are accepted without almost any validation
- This opens a possibility for a network operator to announce someone else's network prefixes without permission
  - The prefix may be announced with the same origin
  - The prefix may be leaked
  - A malicious operator can steal prefixes and blackhole them or intercept and modify traffic in transit
  - A good operator can also steal someone's network occasionally, by an error

BGP Hijacking, a problem

- In the Internet, routing announcements are accepted without almost any validation
- This opens a possibility for a network operator to announce someone else's network prefixes without permission
  - The prefix may be announced with the same origin
  - The prefix may be leaked
  - A malicious operator can steal prefixes and blackhole them or intercept and modify traffic in transit
  - A good operator can also steal someone's network occasionally, by an error
    - A malicious *employee* of a good operator is then able to read and modify incoming traffic as well

BGP Hijacking, a problem

- In the Internet, routing announcements are accepted without almost any validation
- This opens a possibility for a network operator to announce someone else's network prefixes without permission
  - The prefix may be announced with the same origin
  - The prefix may be leaked
  - A malicious operator can steal prefixes and blackhole them or intercept and modify traffic in transit
  - A good operator can also steal someone's network occasionally, by an error
    - A malicious *employee* of a good operator is then able to read and modify incoming traffic as well
  - Unauthorized access to operator's equipment can also be used for hijacking

BGP Hijacking, a problem

- ~30000 IPv4 prefixes leaked during last 2 weeks
  - ~5000 of them in US
  - ~2000 in Australia (far from US)
- ~5000 IPv4 prefixes leaking **right now**
- Almost all this is likely to be caused just by human missteps

BGP Hijacking, a problem

- ~30000 IPv4 prefixes leaked during last 2 weeks
  - ~5000 of them in US
  - ~2000 in Australia (far from US)
- ~5000 IPv4 prefixes leaking **right now**
- Almost all this is likely to be caused just by human missteps
  - Why attackers don’t steal prefixes?

Detection of a hijacking

- Bogus AS Path at Routeviews or some providers' looking glasses
- Change in TTL
- Increased RTT

Detection of a hijacking: hardly possible

- Bogus AS Path at Routeviews or some providers' looking glasses
  – hard to discover without an advanced monitoring system
- Change in TTL
  – easy for a MitM to hide
- **Increased RTT**

“Global Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/23 to its upstream AS B.
3. ?

“Global Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/23 to its upstream AS B.
3. More specific route wins the battle (except IXs, where it may lose), and all traffic to X.Y.Z.1 starts to flow into AS M via AS B.
4. All users of X.Y.Z.1 immediately notice **increased latency**.
5. A bell rings, AS A and AS B figure out the problem and solve it somehow together during next 4-5 business days

Detection of a hijacking: hardly possible

- Bogus AS Path at Routeviews or some providers' looking glasses
  – hard to discover without an advanced monitoring system
- Change in TTL
  – easy for a MitM to hide
- **Increased RTT**

Detection of a hijacking: hardly possible

- Bogus AS Path at Routeviews or some providers' looking glasses
  – hard to discover without an advanced monitoring system
- Change in TTL
  – easy for a MitM to hide
- **Increased RTT**
  – between what?

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. ??

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. It depends on the relations between B and C
   - If B is C’s *customer:*
     - B will prefer the route originating from M
     - C will prefer the route originating from A or B(M)

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. It depends on the relations between B and C
   - If B is C’s *customer:*
     - B will prefer the route originating from M
     - C will prefer the route originating from A or B(M)

=> A global hijacking is *possible*

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. It depends on the relations between B and C
   - If B is C’s *customer:*
     - B will prefer the route originating from M
     - C will prefer the route originating from A or B(M)

     => A global hijacking is *possible*
   - If B is C’s *provider:*
     - C will prefer the route originating from A
     - B will prefer the route originating from C(A) or M

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. It depends on the relations between B and C
   - If B is C’s *customer:*
     - B will prefer the route originating from M
     - C will prefer the route originating from A or B(M)

     => A global hijacking is *possible*
   - If B is C’s *provider:*
     - C will prefer the route originating from A
     - B will prefer the route originating from C(A) or M

     => Hijacking is local to B (at best)

That was an easy part.

“Local Hijacking”

1. Prefix X.Y.Z.0/22 belongs to AS A, which announces it to its upstream AS C
2. One day, AS M announces X.Y.Z.0/22 to its upstream AS B.
3. What happens in B and C, depends on the relations between B and C
4. **What if B and C aren’t directly connected?**

   Things get more complicated in other AS all over the world

“Local Hijacking”

- Things get more complicated in other AS all over the world
- It is possible to steal a prefix “locally” – in a part of the Internet, perfectly isolated by inter-AS relations
  - In fact, that’s why BGP Anycast works
  - RTT will not increase significantly, so no one will notice
  - Looking glasses of *major* network operators will show valid announces

“Local Hijacking”

- Things get more complicated in other AS all over the world
- It is possible to steal a prefix “locally” – in a part of the Internet, perfectly isolated by inter-AS relations
  - In fact, that’s why BGP Anycast works
  - RTT will not increase significantly, so no one will notice
  - Looking glasses of *major* network operators will show valid announces
  - But why would we need that?

Obtaining a TLS certificate from CA

- The procedure is generally as follows:
  1. An account is created at the Web site of a certificate authority
  2. A CSR is created and uploaded
  3. CA offers plenty of options to verify domain ownership:
     - WHOIS records
     - A specific HTML page under a specific URL
     - Custom token in DNS TXT Record
     - …
  4. After the ownership is verified, you get your signed TLS certificate for your money (or sometimes for free)

Stealing a valid TLS certificate, pt. 1

Prerequisite: you need to find a CA close to your AS in topological sense

1. A prefix hosting an IP for the victim's Web site is hijacked *locally,* so that the following conditions apply:
   - At this time victim's AS should notice nothing
   - The chosen CA's traffic is routed to the hijacker
2. Go on: register with the chosen CA, upload a CSR, get an HTML page, upload HTML to your own server, pay and obtain the signed certificate

Stealing a valid TLS certificate, pt. 2

Prerequisite: you need to find a CA close to your AS in topological sense

1. A prefix hosting an authoritative DNS for the victim's Web site is hijacked *locally,* so that the following conditions apply:
   - At this time victim's AS should notice nothing
   - The chosen CA's traffic is routed to the hijacker
2. Go on: register with the chosen CA, upload a CSR, get a token, set up DNS TXT on your own server, pay and obtain the signed certificate

Stealing a valid TLS certificate, pt. 3

Prerequisite: you need to find a CA close to your AS in topological sense

1. A prefix hosting a WHOIS server for the victim's domain registrar is hijacked *locally,* so that the following conditions apply:
   - At this time victim's AS should notice nothing
   - The chosen CA's traffic is routed to the hijacker
2. …

Stealing a valid TLS certificate

- The hijack is local: victim's AS should notice nothing or almost nothing
  - Haha, some guy in Kerbleckistan experiences problems connecting to our site!

- However, the resulting TLS certificate is perfectly global: Kerbleckistanian CA is not *that* worse than GoDaddy or Comodo, the certificate would be valid anywhere

- The resulting TLS certificate can be used for MitM attacks anywhere in the world

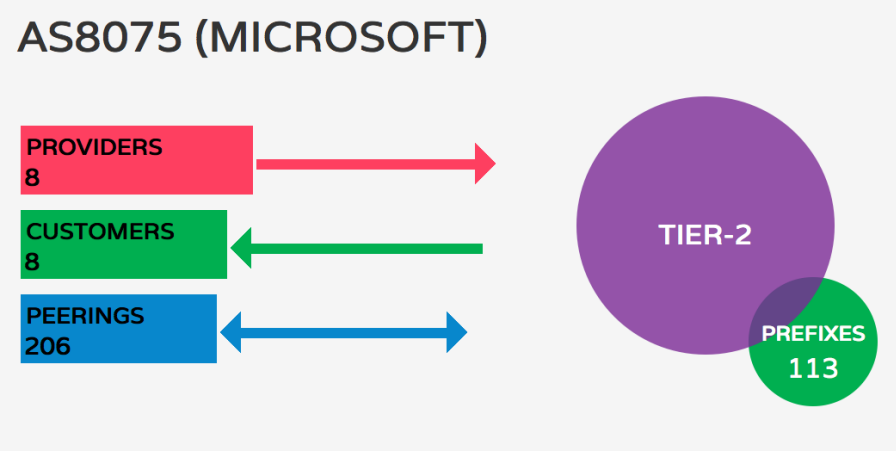
Certificate Authority Hijacking

Vice versa:

- We can steal victim's prefix near selected CA's AS
- We can steal CA's prefix near victim's AS as well
  - The implementation is just a bit more complex

Stealing a valid TLS certificate

- It's not very hard to do a local hijacking. You only need this:
  - A border router under your control
  - Information about your BGP peers: their customers, providers, peerings.
    This is not a top secret: http://radar.qrator.net/ figures out this information on a hourly basis, using public data only: traceroute, AS Paths, etc.
  - That's all



Mitigating the problem.

Mitigating the problem.

…yuck.

Mitigating the problem.

…yuck.

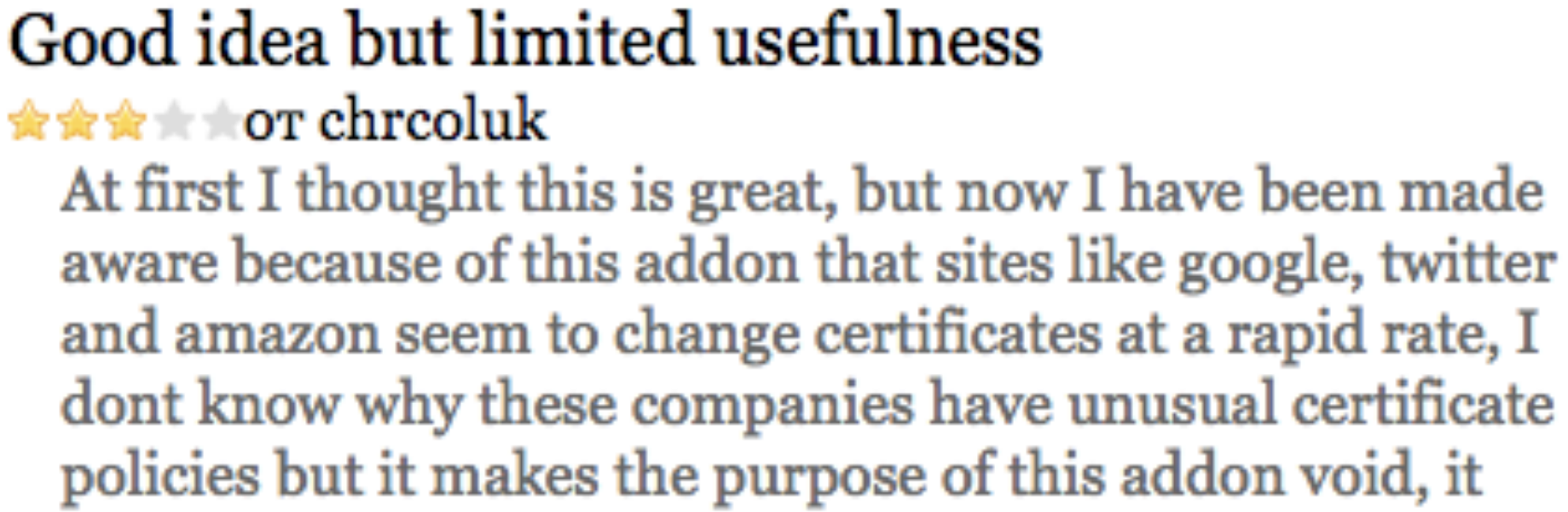
- There's obviously a problem with current SSL/TLS PKI
  - But that's not something we can fix tomorrow
- There's obviously a problem with Internet routing
  - But that's not something we can fix in a decade

Mitigating the problem.

- We have to stick to workarounds:
  - BGP monitoring, able to detect hijacking in Kerbleckistan
    - http://radar.qrator.net/ (it's free, by the way)
    - http://research.dyn.com/
    - http://www.bgpmon.net/
  - Watch your prefixes!
  - RFC 7469 [draft]
  - Browser plug-ins restricting certificate updates (Certificate Patrol etc.)
  - DANE?
  - …

Mitigating the problem.

- We have to stick to workarounds:
  - Browser plug-ins restricting certificate updates (Certificate Patrol etc.)



Mitigating the problem

- There’s obviously a problem with current SSL/TLS PKI
- There’s obviously a problem with Internet routing
- Maybe it’s high time to discuss and fix those problems

Black Hat Sound Bytes

- There are flaws in Internet routing and in TLS PKI concept. There are also corresponding risks
- Those risks could be mitigated. However, the better PKI design will help to do it easier
- BGP monitoring systems are really useful! If you are in charge of network security in a large ISP, please start using them right away

Thank you!

mailto: Artyom Gavrichenkov <ag@qrator.net>