

Designing a workflow to respond to BGP Incidents

Job Snijders

NTT Ltd

job@ntt.net

Agenda

What is the problem space?

Steps

Evidence collection

Analysis

Action

Walk-through of a training scenario

Q&A

Problem space

When you receive a call “You are propagating a hijack!”

..... then what?

- If the reporter is right, you must act quickly
- If the reporter is wrong, and you act trustingly, and disconnect the wrong entity...

Why discuss process around this problem?

We all benefit if we all can respond quickly and consistently to requests for help

Evidence collection usually is a good EBGP filter inspection exercise, could this have been prevented?

When Theo calls you

Hey NTT NOC!
Your customer "Job Snijders / 15562" is
hijacking my 198.58.2.0/24 prefix!
Stop!



Confirm your relation to the reporter

Is the caller / e-mailer an existing customer?

Is their identity known to your organization?

Get their person name, company, phone & email address for follow up! (In exchange give them a ticket ID?)



Question template

- Expected Origin ASN, and authorized upstreams
- Expected prefix length
- Bonus: a website that resides inside the prefix for testing purposes

State collection (on a UNIX shell)

```
$ date
```

```
$ whois -h rr.ntt.net '!r198.58.2.0/24,L'
```

```
$ whois -h rr.ntt.net '!r198.58.2.0/24,M'
```

The purpose of the above commands is to store the current state of NTT's ACL generation input. The `,L` and `,M` options look for less-specific and more specific route objects related to the resource.

Others may want to query their local IRR cache, or RADB.

Example output of state of IRR

```
job@vurt ~$ whois -h rr.ntt.net '!r198.58.2.0/24,L'  
A898  
route:          198.58.2.0/24  
descr:          route object for 198.58.3.0/24  
origin:         AS15562  
mnt-by:         MAINT-JOB  
changed:        job@instituut.net 20191026  
source:         NTTCOM  
  
route:          198.58.2.0/24  
descr:          Theos IP block  
origin:         AS22512  
mnt-by:         MAINT-DERAADT  
changed:        deraadt@openbsd.org 20190731  
source:         NTTCOM  
  
route:          198.58.2.0/24  
descr:          RPKI ROA for 198.58.2.0/24  
..... . . .
```

Getting an overview of the steady state

For the following URLs perform a "Save webpage as PDF" (or "print to PDF"):

- http://lg.ring.nlnog.net/prefix_detail/lg01/ipv4?q=198.58.2.0/24
- <https://stat.ripe.net/198.58.2.0%2F24#tabId=at-a-glance>
- <https://stat.ripe.net/198.58.2.0%2F24#tabId=routing>
- <https://rpki-validator.ripe.net/roas?q=198.58.2.0%2F24>
- <http://irrexplorer.nlnog.net/search/198.58.2.0/24>

Capture the hijack

Try to capture the actual alleged hijack in your own network, please collect from an APAC, EU, and USA router:

```
'show route 198.58.2.0/24 all'
```

```
'traceroute 198.58.2.1'
```

Example BGP output

```
RP/0/RSP0/CPU0:r04.londen05.uk.bb#show bgp ipv4 uni 198.58.2.0/24
```

```
BGP routing table entry for 198.58.2.0/24
```

```
Versions:
```

```
Process          bRIB/RIB  SendTblVer
```

```
Speaker          947857407 947857407
```

```
Last Modified: Oct  4 11:55:16.608 for 1y03w
```

```
Paths: (3 available, best #2)
```

```
Advertised to update-groups (with more than one peer):
```

```
0.2 0.11 0.12
```

```
Advertised to peers (in unique update groups):
```

```
77.67.98.53    202.97.52.49
```

```
Path #1: Received by speaker 0
```

```
Not advertised to any peer
```

```
3257 22512
```

```
77.67.98.53 from 77.67.98.53 (213.200.87.51)
```

```
Origin IGP, metric 0, localpref 100, valid, external, group-best
```

```
Received Path ID 0, Local Path ID 0, version 0
```

```
Community: 2914:390 2914:1203 2914:2201 2914:3200 3257:3257 65504:3257
```

```
Origin AS validity: not found
```

```
Path #2: Received by speaker 0
```

```
Advertised to update-groups (with more than one peer):
```

```
0.2 0.11 0.12
```

```
Advertised to peers (in unique update groups):
```

```
77.67.98.53    202.97.52.49
```

```
15562
```

```
192.147.168.225 (metric 20334) from 129.250.0.130 (129.250.0.130)
```

```
Origin IGP, localpref 120, valid, confed-internal, best, group-best
```

```
Received Path ID 0, Local Path ID 0, version 947857407
```

```
Community: 2914:370 2914:1004 2914:2000 2914:3000
```

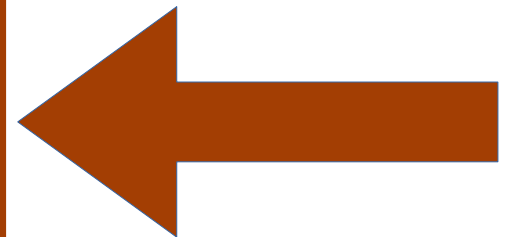
```
Path #3: Received by speaker 0
```

```
Not advertised to any peer
```

```
15562
```

```
192.147.168.227 (metric 20345) from 129.250.0.145 (129.250.0.145)
```

```
Origin IGP, localpref 120, valid, confed-internal
```



Collecting traceroutes is important

Should consider it a priority, can't replay it retroactively

Are you dealing with a ghost route? Where is the data path actually taking folks?

If we are “too late” (aka hijack is over)

If we are too late, the issue can be deferred for later analysis. NTT will assess on a case by case basis what help can be offered.

If we proceed to produce a post-mortem, we’d use our internal MRT IBGP archive to analyze whether we accepted or propagated the hijack, supplemented with RIPE RIS, Routeviews, etc.

Note: setting up EBGP sessions to dfzwatch, routeviews, ripe ris, isolario, etc, helps everyone!

Back to those URLs

The purpose of collecting information from these websites is to figure out whether the reported hijack announcement has any validity or not.

If the <http://lg.ring.nlnog.net/> website indicates that the announcement is RPKI invalid, we can more quicker move to a conclusion.

NLNOG RING looking glass x +

← → ↻ ⓘ Not secure | lg.ring.nlnog.net/prefix_detail/lg01/ipv4?q=198.58.2.0

Nodes: all lg01 Protocols: ipv4 ipv6 show route for 198.58.2.0

lg01: show route for 198.58.2.0/32 all

```
198.58.2.0/24
[NTT1 20:35:31 from 129.250.1.96] * (100/-) [AS15562i]
  Type: BGP unicast univ
  BGP.origin: IGP
  BGP.as_path: 2914 15562
  BGP.next_hop: 129.250.1.96
  BGP.med: 16944
  BGP.local_pref: 100
  BGP.community: (2914,410) (2914,1004) (2914,2000) (2914,3000)
  BGP.ext_community: (RPKI Origin Validation State: invalid)
[ALTIBOX1 20:35:44 from 79.160.115.253] (100/-) [AS15562i]
  Type: BGP unicast univ
  BGP.origin: IGP
  BGP.as_path: 29695 3356 2914 15562
  BGP.next_hop: 79.160.115.253
  BGP.local_pref: 100
```


IRR & RPKI data can easily change over time

Since IRR and RPKI data may change over time, it is prudent to store the 'current state' (as PDFs?) so that we can more easily construct a post-mortem if needed.

Impact analysis

- is the reporter the same entity as the victim?
- if the reporter is the victim, can they quantify the impact?
- is their whole company down, or was the IP space not in use?
- is the prefix "well-known" or "golden" in the sense that it is something like 1.1.1.0/24, 8.8.8.0/24 or one of the ccTLD, gTLD, or DNS root servers?
- Is the prefix in your top XYZ traffic destinations?

Follow up actions / how to stop hijacks

Call the originator of the prefix – use WHOIS / RDAP / PeeringDB / your CMS for contact information, and ask to revert their change

Especially in the case of accidental misconfigurations, people generally are happy to cooperate to resolve the issue. We should assume positive intent.

(Second question: ask if they have enabled a “BGP optimizer”)

Approach peers/upstream providers

If the entity that originates the incorrect route announcement is not directly connected to the NTT backbone, but rather through one of our competitors such as Telia or Level3, and “direct call” was not successful;

we can reach out to the originator’s upstream providers and request them to block the rogue announcement.

The reporter should participate in the chase

If the hijack is caused by a customer of NTT, contacting NTT is of course appropriate....

but if our role in that context is that of “intermediate transit network”... It may be better for the reporter to directly reach out to closer to the source.

Start by reaching out to the right most ASN in the AS_PATH!

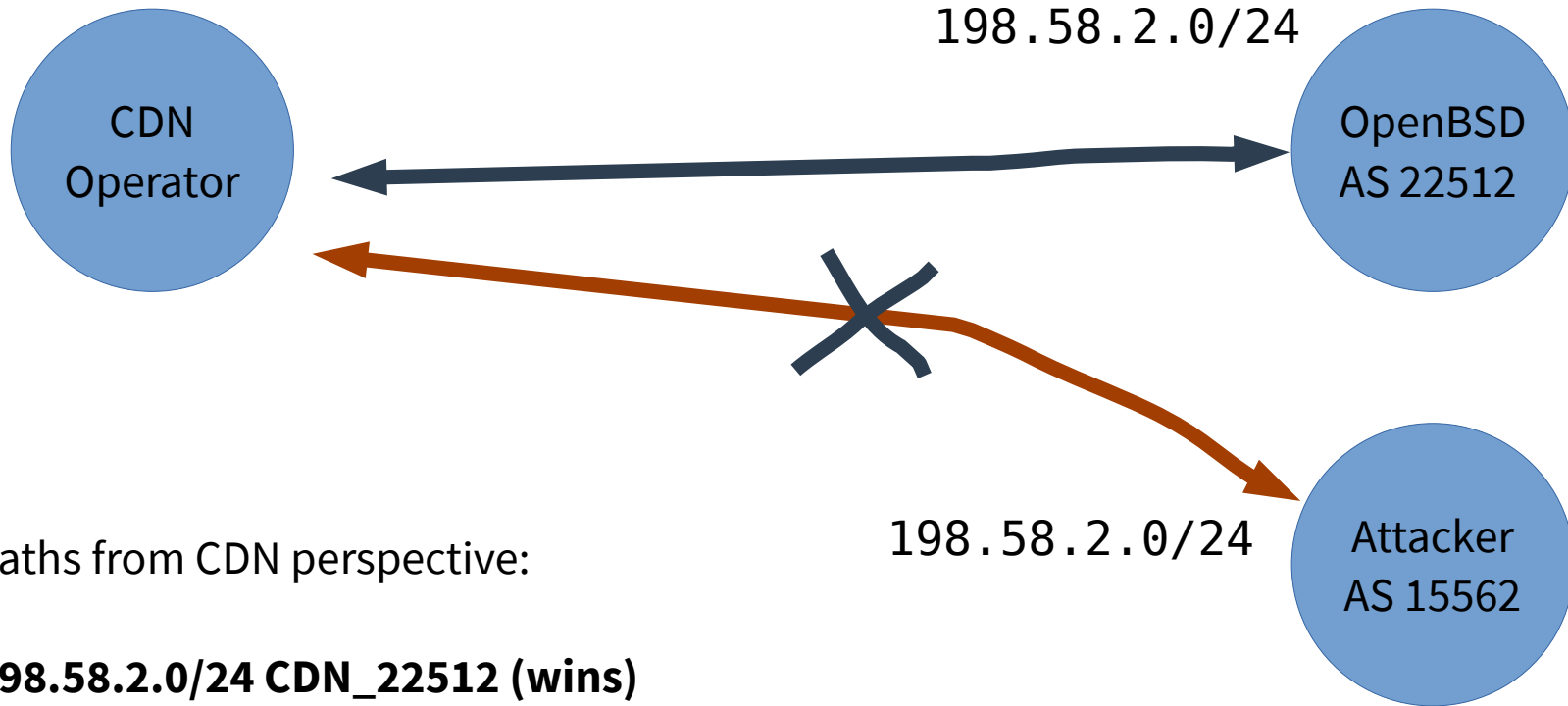
Origin Validation and lack of path validation

We peer directly in many cases if we care about the traffic.

Origin Validation - combined with direct peering - is a very powerful '1hop verified' protection

Change of tactics: announce same prefix

Cloudflare applying “invalid == reject”



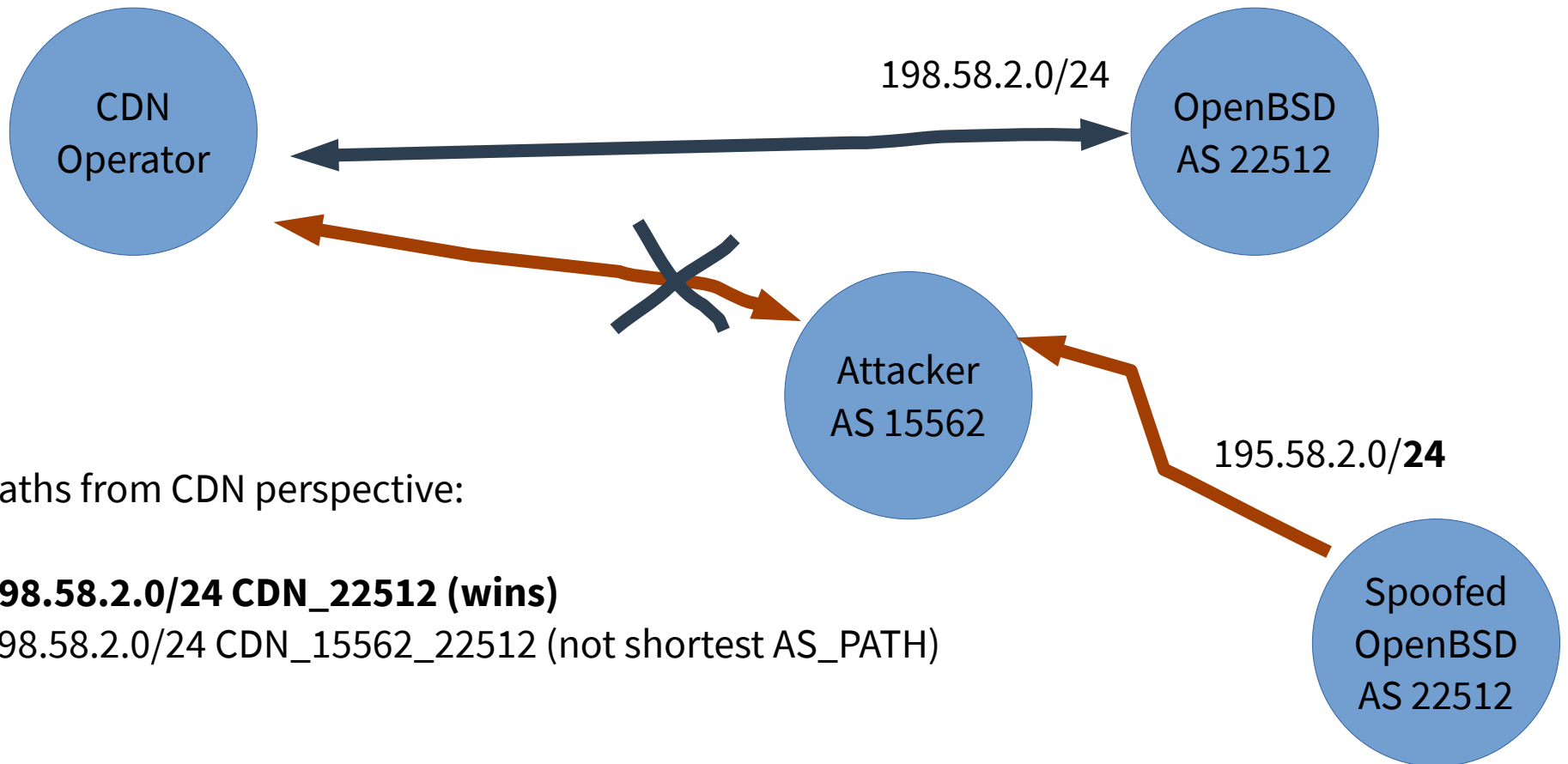
Paths from CDN perspective:

198.58.2.0/24 CDN_22512 (wins)

195.58.2.0/24 CDN_15562 (rejected, wrong Origin ASN)

Even spoofed origins or leaks are less effective

Cloudflare applying “invalid == reject”



Paths from CDN perspective:

198.58.2.0/24 CDN_22512 (wins)

198.58.2.0/24 CDN_15562_22512 (not shortest AS_PATH)

Clean up IRR entries for rogue announcements

Ideally not only the source of the hijack in the BGP Default-Free Zone is stopped, but the routing registry information that allowed it to become part of the 'allow list' ceases to be too.

Fixing IRR often is a quick way to deploy new correct filters.

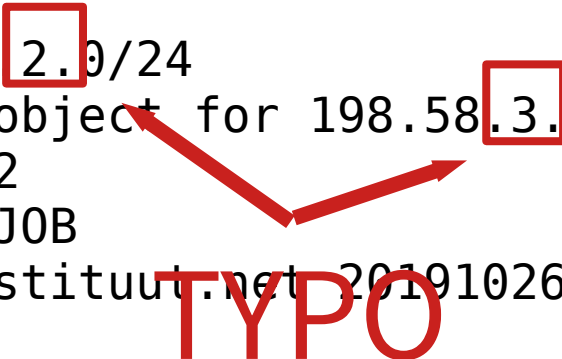
<http://www.irr.net/docs/list.html> has a list of contact details for various IRR databases

Back to the output of state of IRR, mis copy+paste

```
job@vurt ~$ whois -h rr.ntt.net '!r198.58.2.0/24,L'
```

```
A898
```

```
route:          198.58.2.0/24
descr:          route object for 198.58.3.0/24
origin:         AS15562
mnt-by:         MAINT-JOB
changed:        job@instituut.net 20191026
source:         NTTCOM
```



```
route:          198.58.2.0/24
descr:          Theos IP block
origin:         AS22512
mnt-by:         MAINT-DERAADT
changed:        deraadt@openbsd.org 20190731
source:         NTTCOM
```

```
route:          198.58.2.0/24
descr:          RPKI ROA for 198.58.2.0/24
```

```
.....
```

Questions & Answers



This presentation was created on OpenBSD 6.6