Resource Certification Using DNS + IRR's

Shane Amante
Brian Deardorff
Mike Feldpusch
Eric Hendrickson
Dan Luther
Mark Persiko
Level 3 Communications

Eric Osterweil
Verisign

3 "Horsemen" of the Internet

1. **IRR** = *Policy* Information
2. **Routing** = *Reachability* Information (BGP)
3. **DNS** = Naming

Why not (just) IRR's for resource certification?

- A: "IRR sucks"
  - But, what does that really mean?
- A: *Quality of IRR data sucks*
- Q: Can we improve IRR data quality?
- A: Yes, with a light *assist* from DNS

Why DNS?

- DNSSEC is deployed for reverse DNS
- Single root, single Trust Anchor *deployed & operational*
- Substantial resiliency & scalability to DNS, built from *decades of operational experience*
- Massive amount of monitoring, documentation, training, tool chains, etc. built around DNS

How would such a system work?

1. (IP prefix, Origin_AS) SRO RRset encoded in DNS per: **`draft-gersch-dnsop-revdns-cidr`**
2. Build *Candidate IRR Origin List* (IP prefix, Origin_AS) from IRR 'route' objects
3. Verify each Candidate IRR Origin (IP prefix, Origin_AS) using DNSSEC SRO RRset lookup
4. Results in #3 determine *Final, IRR Origin List* used for offline verification and/or pushed to routers in network

Details of DNS Lookup

1. **Valid** - DNSSEC SRO RRset matches Candidate (IP prefix, Origin_AS)
2. **Invalid** - DNSSEC SRO RRset does not match Candidate (IP prefix, Origin AS)
3. **Not Found** in DNSSEC SRO RRset -
    1. If [more-specific] (IP prefix, Origin_AS) contained in already valid aggregate/covering (IP prefix, Origin_AS), then **Valid** and issue *Warning message* to Administrator
    2. If (IP prefix, Origin AS) not in a valid aggregate, then **Invalid** and issue *Error message* to Administrator

Origin Encoding & Verification in DNSSEC

```
# 204.199.192.0/20, Origin_AS = AS 596
$ dig +dnssec -t TYPE65401 0.0.1.1.m.199.204.in-addr.arpa | egrep '(ad\;|65401 \\#)'
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1
0.0.1.1.m.199.204.in-addr.arpa.   86379 IN TYPE65401 \# 4 00000254; 0x254 = AS 596
```

```
# 204.199.240.0/22; Origin_AS = AS 596
$ dig +dnssec -t TYPE65401 0.0.1.1.1.1.m.199.204.in-addr.arpa | egrep '(ad\;|65401 \
\#)'
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
0.0.1.1.1.1.m.199.204.in-addr.arpa. 86400 IN TYPE65401 \# 4 00000254; 0x254 = AS 596
```

```
# 204.199.20.0/24; Origin_AS = AS 598
$ dig +dnssec -t TYPE65401 m.20.199.204.in-addr.arpa | egrep '(ad\;|65401 \\#)'
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
m.20.199.204.in-addr.arpa. 86400 IN   TYPE65401 \# 4 00000256; 0x256 = AS 598
$
```

Conclusions

- Using DNSSEC for resource certification of information in IRR's is straightforward
- Using DNSSEC + IRR's will leverage existing, deployed systems, tools and training
- Questions?