Reverse-DNS Naming Convention for CIDR Address Blocks

draft-gersch-dnsop-revdns-cidr-01

Joe Gersch
Secure 64 SW Corp

Dan Massey
Colorado State University

Eric Osterweil
Verisign

How Do You Represent a CIDR Prefix in Reverse DNS?

- Conventions exist for representing IP addresses
  - Both IP v4 and IPv6
  - 129.82.138.2 => 2.138.82.129.in-addr.arpa
- Conventions for representing a prefix?
  - Neither IPv4 or IPv6
  - 129.82.138/24 => ????
  - No guidance limits innovation and leads to fractures

What Doesn't Change

- No request for a new record type
- No change in any DNS resolver
- No change in any DNS server
- No change in any DNS cache
- No change in DNSSEC processing
- Not mandatory to use it

We seek only a prefix naming convention

But Doesn't This Already Work?

- Why not just treat a prefix like an address?
  - 129.82.0.0/16 => 16/0.0.82.129.in-addr.arpa
- Authority for data is in the wrong zone
  - 129.82.0.0/16 owned by Colorado State Univ
  - 129.82.0.0/24 delegated to small department
  - 16/0.0.82.129.in-addr.arpa belongs to small department
- Does not handle CIDR masks:
  - 129.82.128.0/17 => ???
  - RFC 2317 and others don't apply to prefixes

Prefix Chaining: The Main Idea

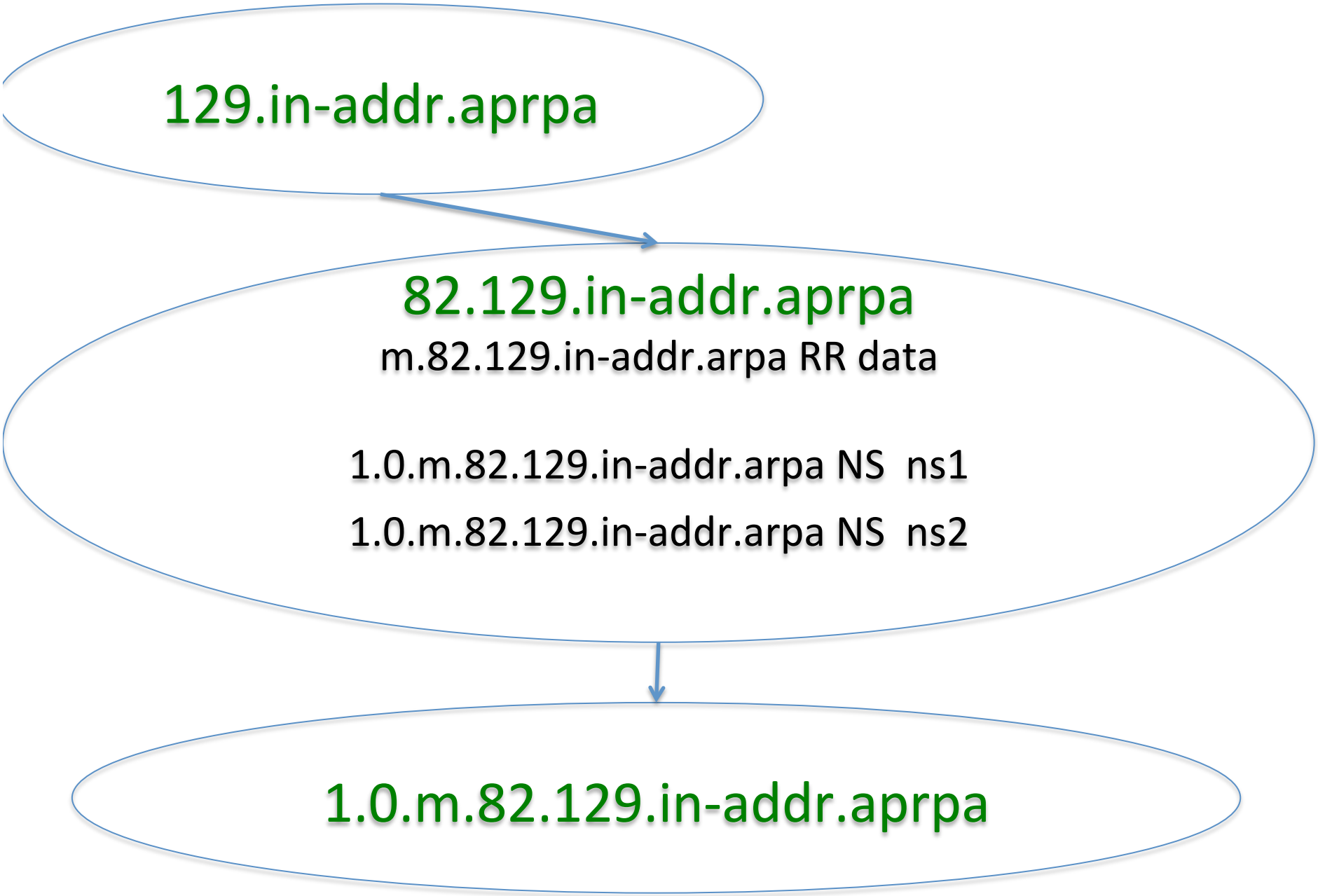
- Use the existing reverse tree whenever possible
- Switch to binary when you are off an octet boundary
- Result adds to existing reverse DNS tree while also preserving prefix hierarchical structure

Prefix Chaining – Step by Step

- Step 1: Drop the unnecessary octets
  - 129.82.***0.0***/16 → 129.82
- Step 2: Calc the prefix length mod 8
  - 16 mod 8 = 0
- Step 3: If N=0, reverse and append
  - m.82.129.in-addr.arpa

Prefix Chaining – Step by Step

- Step 1: Drop unnecessary octets
  - 129.82.64***.0***/18 → 129.82.64
- Step 2: Calculate the prefix length mod 8
  - 18 mod 8 = 2
- Step 3: If N != 0, expand bits
  - Insert m before last octet
    - 129.82.64 → 129.82***.m***.64
  - Convert last octet to binary
    - 129.82.m.64 → 129.82.m.***0.1.0.0.0.0.0.0***
  - Truncate to L mod 8 bits, reverse, and append
    - ***129.82.m.0.1***.0.0.0.0.0.0 → 1.0.m.82.129.in-addr.arpa


129.in-addr.aprpa
82.129.in-addr.aprpa
m.82.129.in-addr.arpa RR data
1.0.m.82.129.in-addr.arpa NS ns1
1.0.m.82.129.in-addr.arpa NS ns2
1.0.m.82.129.in-addr.aprpa

Summary

- No convention for naming a CIDR prefix
  - A convention would help innovation and prevent fracture (classic standards case)
  - Should be application agnostic
  - No protocol or implementation changes
- Proposed such a naming convention
  - No major changes suggested
  - Many minor changes suggested. Thanks!