

PrefixLen & ExtRef

OpsAreaWG / IETF Dublin

2024.11.06

Oliver Gasser <oliver@ipinfo.io>,

Massimo Candela <massimo@us.ntt.net>,

Tom Harrison <tomh@apnic.net>,

Russ Housley <housley@vigilsec.com>,

Randy Bush <randy@psg.com>

Two Drafts

draft-gasser-opsawg-prefix-lengths

Like GeoFeed for publishing an Op's
customer assignment size

draft-ymbk-opsawg-rpsl-extref

Generalizing these external references

Initial Motivation

- Applications want to know the width of IP space used by a 'customer'
 - Blocklisting/throttling
 - Rate limiting/CAPTCHAs
 - etc.
- It benefits ISPs to publish this as it benefits our customers by reducing collateral damage

PrefixLen CSV File

2001:db8::/32,56

192.0.2.0/24,32

Publish the **width** of allocations
within a prefix

inetnum: Hack

```
inetnum: 192.0.2.0/24 # example  
prefixlen: https://my.com/prefixlen
```

or, in the interim

```
inetnum: 192.0.2.0/24 # example  
remarks: Prefixlen https://my.com/prefixlen
```

PrefixLen Files

inetnum: 192.168.0.0/16

prefixlen: <https://my.com/prefixlen>

and the prefixlen file might be

192.168.0.0/17,24

192.168.128.0/18,27

192.168.192.0/20,24

192.168.224.0/20,23

Authenticity

As with GeoFeed files,
PrefixLen files may be
signed using the RPKI of
the prefix in the referring
inetnum:

Similar Restrictions Apply

CGNAT

The 'consumer' of the PrefixLen data (a provider of services) needs to know that there is high granularity and mobility

192.0.0.0/20,36

The CGNAT uses 192.0.0.0/20 and 16 customers share each /32

That's PrefixLen

on to ExtRef

Proliferation!

Oops! Now we have

inetnum: 192.0.2.0/24

geofeed: <https://my.com/geofeed>

prefixlen: <https://my.com/prefixlen>

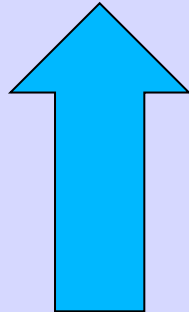
There will be more!! They will breed!!

So Generalize!

inet6num: 2001:db8::/32

extref: Geofeed <https://my.com/geofeed>

extref: Prefixlen <https://my.com/prefixlen>



SubTypes

IANA SubType Registry

- Require IANA SubType Registry
- So things do not go random and crazy
- Beware, SubTypes could refer to objects more complex than files, e.g. code!!!
- So there is a big security issue
- So the SubType Registry requires RFCs

Warning!!!

The Webbies
are Coming

whois -> RDAP

- RIRs want to move from whois to RDAP
- So the inetnum: hacks will also migrate
- RIRs already have GeoFeed in test
- This will be a long transition, we just need to be aware of it
- We will still need a bulk pull as opposed to the RDAP tweezers
- RDAP replaces whois not FTP

```

{
  "objectClassName": "ip network",
  "startAddress": "2001:db8::0",
  "endAddress": "2001:db8:0:ffff:ffff:ffff:ffff:ffff",
  ...
  "links": [
    {
      "value": "https://example.net/ip/2001:db8::/48",
      "rel": "geo",
      "href": "https://example.com/geofeed",
      "type": "application/geofeed+csv"
    },
    {
      "value": "https://example.net/ip/2001:db8::/48",
      "rel": "prefix",
      "href": "https://example.com/prefixlen",
      "type": "application/prefixlen+csv"
    },
    ...
  ],
  ...
}

```

And every simple
thing needs 42
new features!

Other Attributes

- "type" of customer (residential / business / event / 3G/4G/5G/42G, other)
- last mile properties - e.g: rough indication of latency and BW
- publisher extensible data

These smell like infinite rat holes to what end?

But cool if someone come up with a short list with rigorous definitions

B Ark

- "type" of customer (residential / business / event / other)
- last mile properties - e.g: rough indication of latency and BV
- publisher extensible data

These smell like infinite rat holes to what end?

But cool if someone come up with a short list with rigorous definitions