Technical Debt: An Anycast Story

Tom Strickx
Cloudflare, London

HKNOG 7.0

Hong-Kong

Tom Strickx

- Network Hooligan at Cloudflare (Network Software Engineer)
- Contributor at NAPALM Automation and Saltstack
- https://tom.strickx.com

Ichabond

@tstrickx

Agenda

- Anycast introduction
- Our technical debt
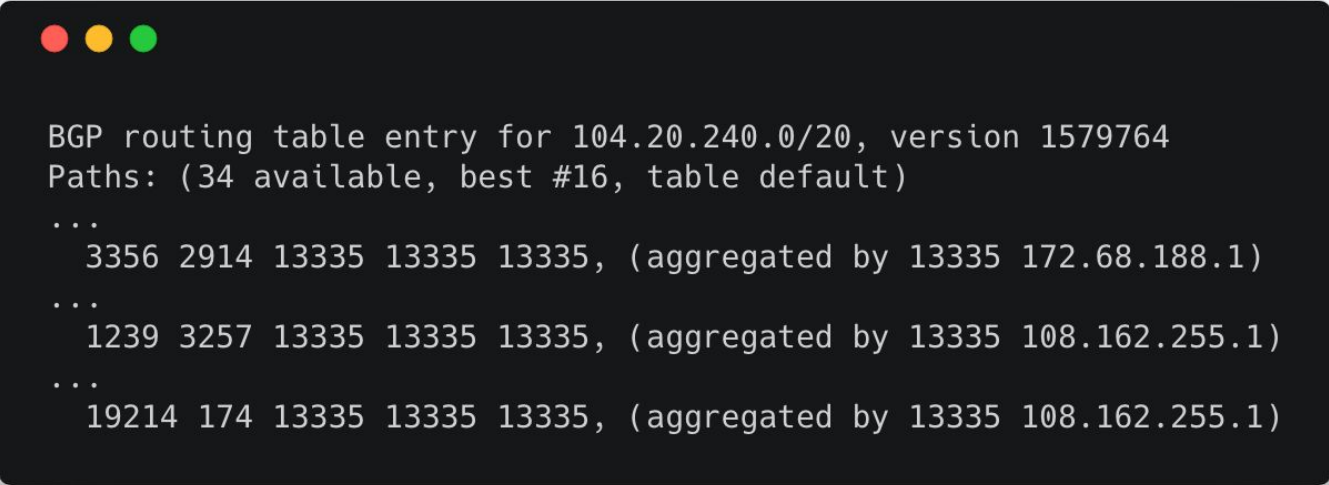- Configuration changes using Saltstack
- Change monitoring

ANYCAST ALL THE THINGS

Our Anycast Network

- ± 250 IPv4 prefixes

- ± 15 IPv6 prefixes

- Announced globally (150+ locations)

Technical Debt

```
BGP routing table entry for 104.20.240.0/20, version 1579764
Paths: (34 available, best #16, table default)
...
  3356 2914 13335 13335 13335, (aggregated by 13335 172.68.188.1)
...
  1239 3257 13335 13335 13335, (aggregated by 13335 108.162.255.1)
...
  19214 174 13335 13335 13335, (aggregated by 13335 108.162.255.1)
```

Technical Debt
History

- Few Tier 1 transit providers
- Prepends to steer traffic to proper location (± 10 PoPs)
- Eventually normalized globally

Technical Debt

Issues
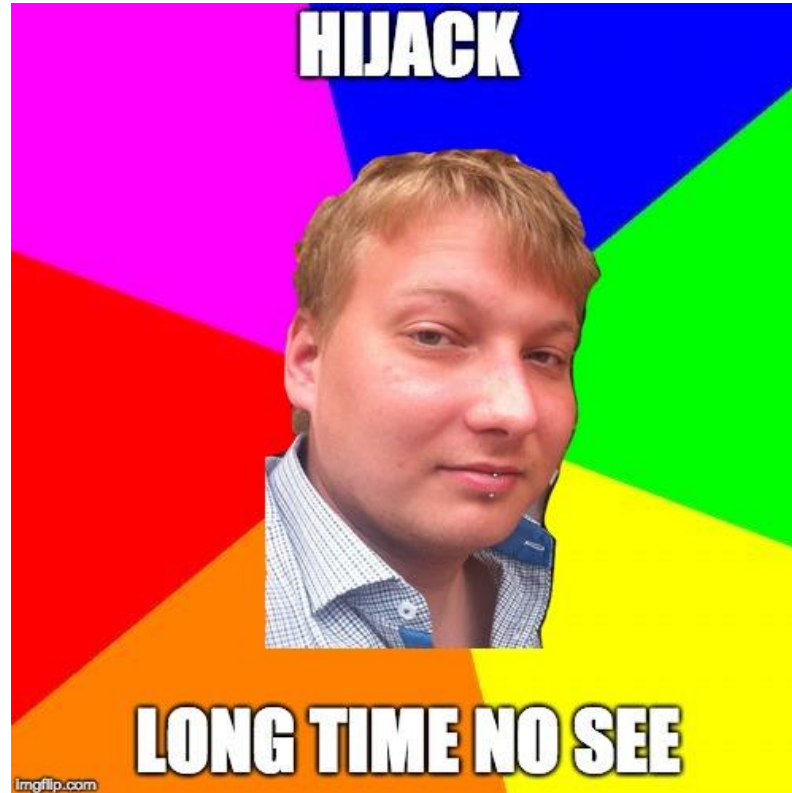
Technical Debt
Incidents

- Authoritative DNS targeted (eg. 173.245.58.0/24)

- Single location attracts all traffic due to missing prepends

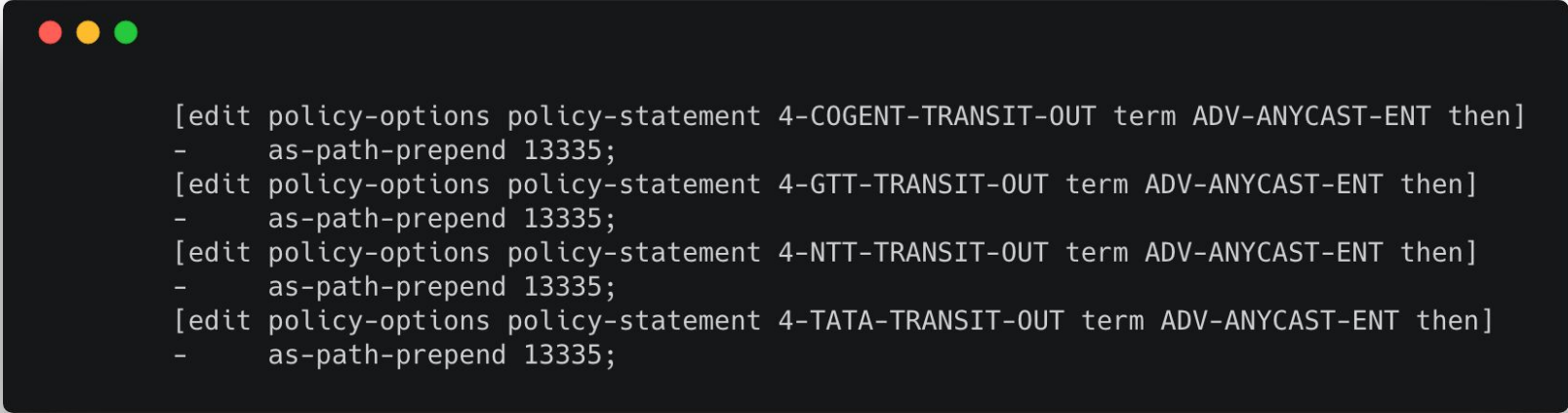# Technical Debt
Solutions

- RPKI
- Shorter AS-path
- /24 everything

Technical Debt
Resolution

- Staggered deployment (6 stages)
- As quickly as possible globally
- Extensive internal and external monitoring

Technical Debt

Change

```
[edit policy-options policy-statement 4-COGENT-TRANSIT-OUT term ADV-ANYCAST-ENT then]
-       as-path-prepend 13335;
[edit policy-options policy-statement 4-GTT-TRANSIT-OUT term ADV-ANYCAST-ENT then]
-       as-path-prepend 13335;
[edit policy-options policy-statement 4-NTT-TRANSIT-OUT term ADV-ANYCAST-ENT then]
-       as-path-prepend 13335;
[edit policy-options policy-statement 4-TATA-TRANSIT-OUT term ADV-ANYCAST-ENT then]
-       as-path-prepend 13335;
```
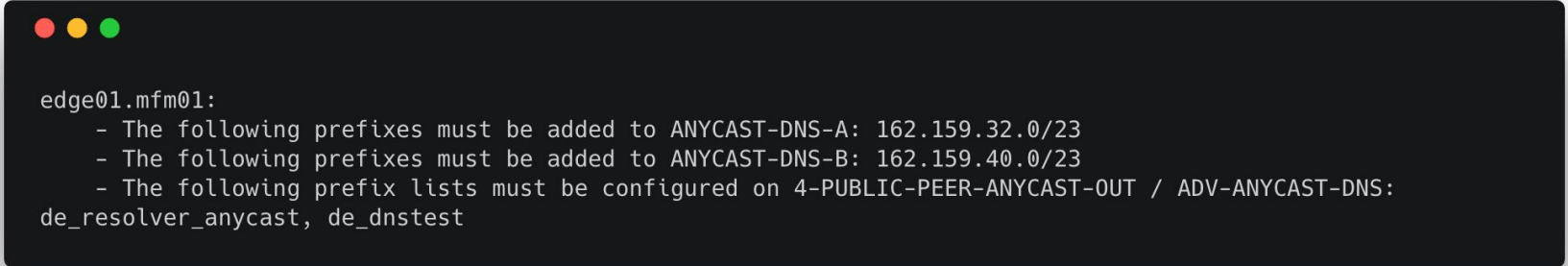
Global Rollout

Global Rollout
Saltstack

- Automation and orchestration
- Open source
- Python, Jinja2 & YAML
- Highly scalable
- Very fast
- Vendor neutral
- Across our fleet: servers and network equipment

Global Rollout
Prechecks

- Make sure we know what we're changing

- Adjust configuration if needed

- Add confidence

```
edge01.mfm01:
    - The following prefixes must be added to ANYCAST-DNS-A: 162.159.32.0/23
    - The following prefixes must be added to ANYCAST-DNS-B: 162.159.40.0/23
    - The following prefix lists must be configured on 4-PUBLIC-PEER-ANYCAST-OUT / ADV-ANYCAST-DNS:
de_resolver_anycast, de_dnstest
```

Global Rollout
Actual Change

- All in Python (config generation)
- Both Junos and EOS
- Concurrently
- Done globally within
  ± 2 minutes

```
edge01.ams01:
    ----------
    diff:
        [edit policy-options policy-statement 4-XXX-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
        [edit policy-options policy-statement 4-YYY-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
        [edit policy-options policy-statement 4-ZZZ-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
        [edit policy-options policy-statement 4-AAA-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
        [edit policy-options policy-statement 4-BBB-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
        [edit policy-options policy-statement 4-CCC-TRANSIT-OUT term ADV-ANYCAST-ENT then]
        -       as-path-prepend 13335;
    loaded_config:

        delete policy-options policy-statement 4-XXX-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
        delete policy-options policy-statement 4-YYY-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
        delete policy-options policy-statement 4-ZZZ-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
        delete policy-options policy-statement 4-AAA-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
        delete policy-options policy-statement 4-BBB-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
        delete policy-options policy-statement 4-CCC-TRANSIT-OUT term ADV-ANYCAST-ENT then as-path-prepend
    result:
        True
```

Global Rollout
Metrics

- Internal metrics
- External metrics

Global Rollout
Internal metrics

- Stored in Clickhouse or Prometheus

- Visualized with Grafana

- Flows

- SNMP data

- Request data

Global Rollout
Clickhouse

- Developed at Yandex

- Column-oriented DBMS

- Open source

- 3 PB on disk

- 100 Gbps insertion

Global Rollout
Clickhouse

- Stores flow data

- Stores request data

Clickhouse query

Result

```sql
SELECT coloId,
       dictGetString('colo', 'airport', CAST(coloId AS UInt64)) AS colo,
       count(*) AS numFlows,
       sum(packets*samplingRate) AS sumPkts,
       sum(bytes*samplingRate*8) AS sumbits,
       round(sum(packets*samplingRate/(301*1000)),1) AS rateKpps,
       round(sum(bytes*samplingRate*8/(301*1000000)),2) AS rateMbps
  FROM netflows
 WHERE date <= toDate('2018-08-24 21:32:11')
   AND timeFlow <= toDateTime('2018-08-24 21:32:11')
   AND date >= toDate('2018-08-24 21:27:11')
   AND timeFlow >= toDateTime('2018-08-24 21:27:11')
 GROUP BY coloId,
       colo
 ORDER BY sumbits DESC
 LIMIT 10
```

coloId	colo	numFlows	sumPkts	sumbits	rateKpps	rateMbps
	SJC01					
	PDX01					
	LAX01					
	FRA03					
	AMS01					
	SIN02					
	HKG01					
	LHR01					
	ORD02					
	SEA01					

Ok. 10 rows in set. Elapsed: 3.469 sec. Processed: 0 rows, 0.0B (0 rows/s, 0.0B/s)
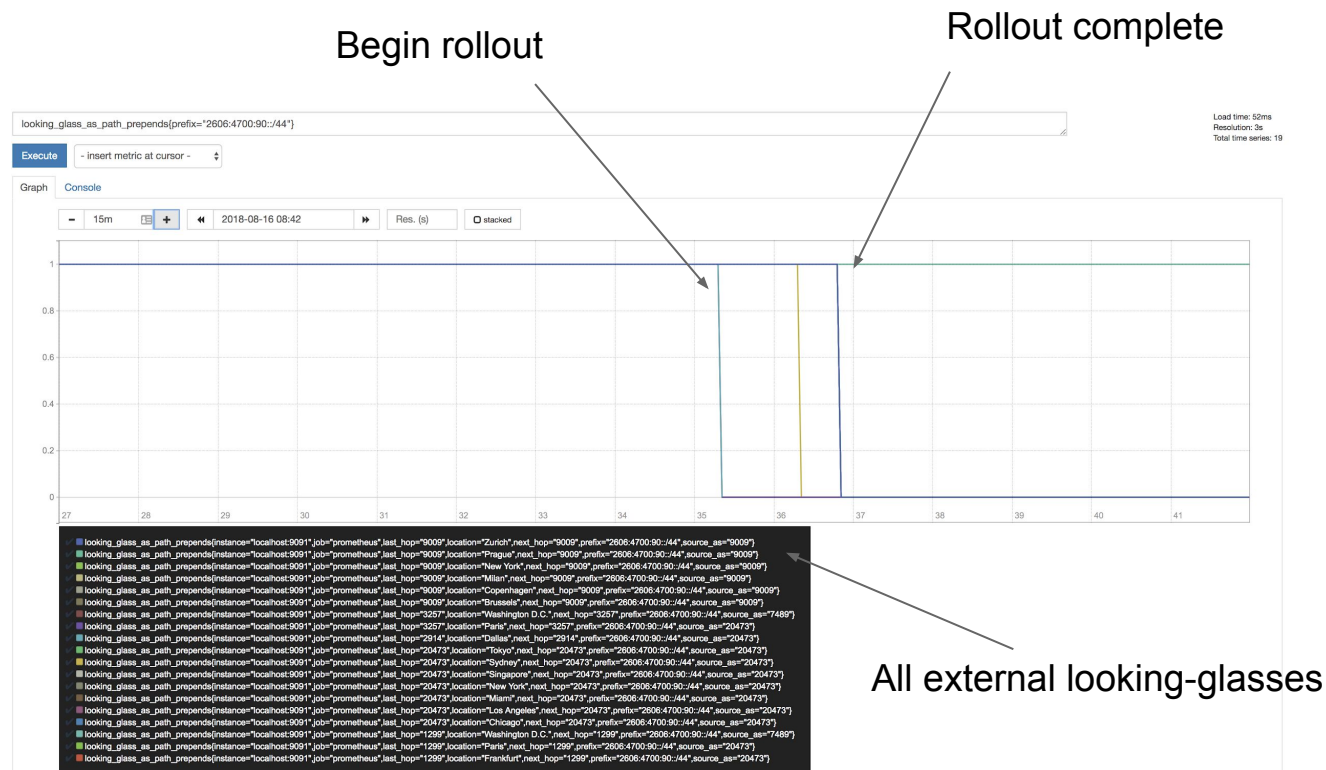
Global Rollout
Prometheus

- Time-series database

- Monitoring platform

- Open source

Global Rollout
Prometheus

- Prepend length
- PromQL

Begin rollout

Rollout complete

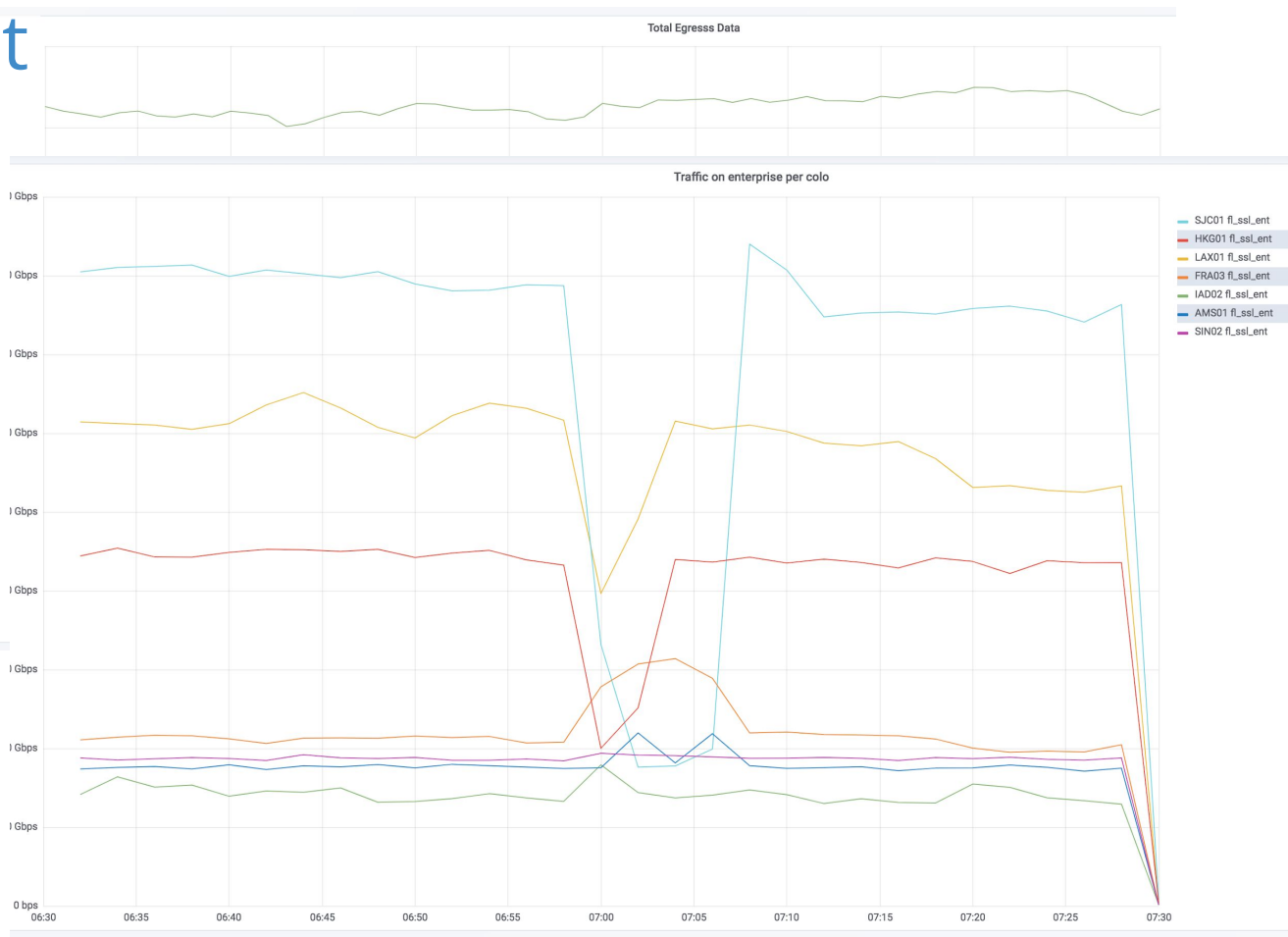All external looking-glasses

Global Rollout
Grafana

- Analytics

- Time-series visualization

- Multitude of plugins

- Open source

Global Rollout
Internal Metrics

- RPS / colo

- Traffic shift during chang

- Real-time information

Total Egresss Data

Traffic on enterprise per colo

Legend:
- SJC01 fl_ssl_ent
- HKG01 fl_ssl_ent
- LAX01 fl_ssl_ent
- FRA03 fl_ssl_ent
- IAD02 fl_ssl_ent
- AMS01 fl_ssl_ent
- SIN02 fl_ssl_ent

Global Rollout
External metrics

- Stored in Prometheus or raw ingestion
- Visualized with Grafana

- RIPE Atlas
- Looking glasses

Global Rollout
RIPE Atlas

- Global probes

- Ping, Traceroute, DNS query

- REST API

- Determine routing before and after change

Global Rollout
Looking Glasses

- Routeviews

- AS57335 (http://dfz.watch/) looking glasses (Thanks Aaron!)

- IX looking glasses (We need more! With APIs!)

- Collect / scrape into Prometheus

- Visualize with Grafana

Jerome Fleury @Jerome_UZ · Jul 5
Planning simultaneous change of 32000 BGP sessions across multiple vendors. What could go wrong ?

💬 13   🔁 4   ❤️ 36   ✉️

Aaron A. Glenn
@networkservice

Following

Replying to @Jerome_UZ

I put up some looking glasses on some of my anycast instances for one of your coworkers to watch what happens from the outside 🙃

Godspeed, good luck, and all that.

Global Rollout
Looking Glasses

- Scrape metrics

- BeautifulSoup for scraping

- Aggregate data

```python
def check_lg(lg_name, looking_glass, params):
    prefixinfo = {}
    lg_prefixes_seen = 0
    r = requests.get(looking_glass['url'].replace('https', 'http'), params=params, timeout=10)
    if r.status_code != 200:
        return {}
    soup = BeautifulSoup(r.text, 'html.parser')
    data = soup.body.div.pre.string.strip()
    # remove extraneous header content
    prefixes = data.splitlines()[4:]
    compiled = re.compile(r"([ ]{0,1}\d*)+?([ ]+"+params['req']+r"+)+")
    for prefix in prefixes:
        prefix_raw = prefix
        prefix = prefix.replace('*>', '').replace('success.', '').strip().split()
        prefix_old = prefix
        prefix = prefix[0:4]
        if prefix:
            # FORMAT:
            # [PREFIX, GATEWAY, LOCALPREF, MED, AS-PATH]
            prefix.append(" ".join(prefix_old[4:-1]))
            asn_data = compiled.search(prefix[4])
            asn_list = asn_data.group().split()
            last_hop = _get_last_hop(asn_list, params['req'])
            our_as = re.finditer(params['req'], prefix[4])
            no_private = [asn for asn in prefix_old[4:-1] if int(asn) not in range(64512, 65535)]
            prefixinfo[prefix[0]] = {'next_hop_asn': asn_data.group(1),
                                     'last_hop_asn': last_hop,
                                     'path_length': len(no_private),
                                     'prepend_length': sum(1 for _ in our_as) - 1}
    return {lg_name: {'prefix_info': prefixinfo}}
```
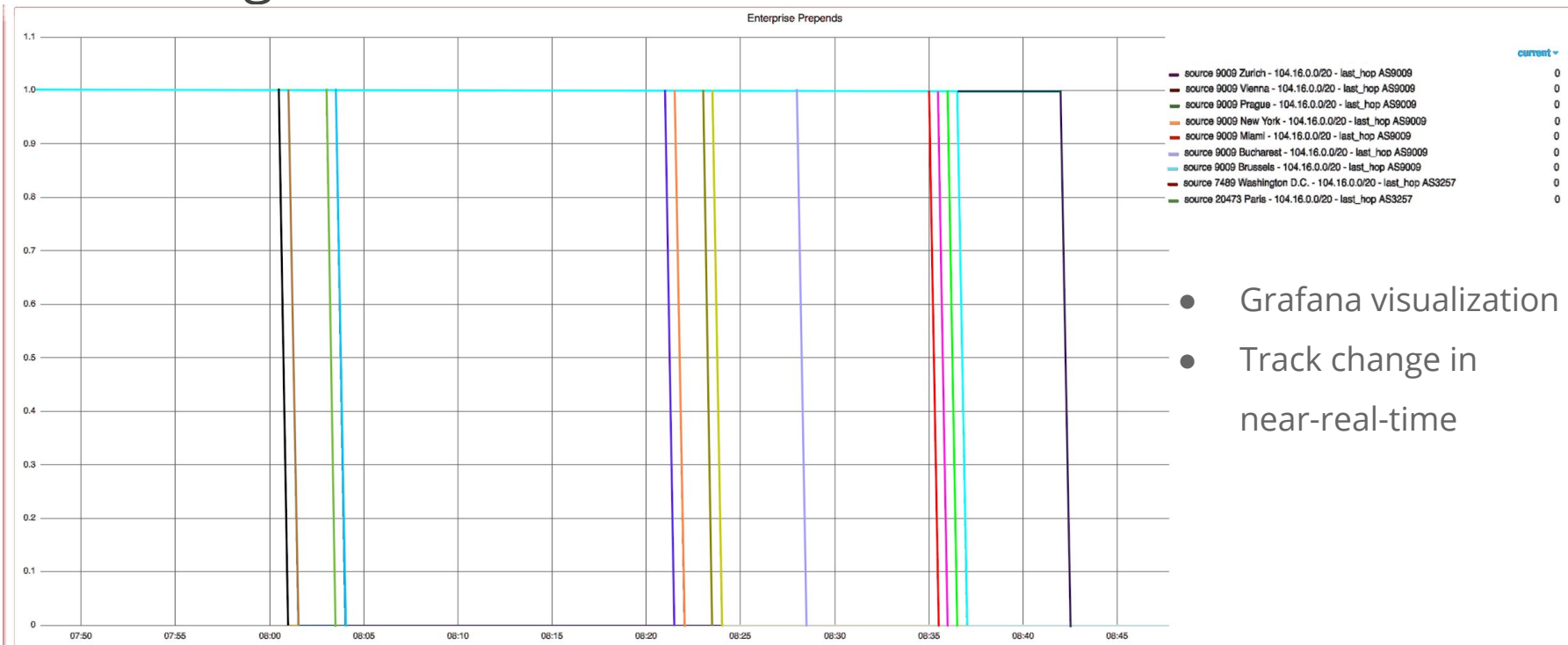
Global Rollout
Looking Glasses

- Insert into Prometheus
- [python_client](#)

```python
for looking_glass, prefix_info in datadump.items():
    for prefix, prefix_data in prefix_info['prefix_info'].items():
        next_hop = prefix_data['next_hop_asn'].strip()
        last_hop = prefix_data['last_hop_asn'].strip()
        labels = [
                    prefix,
                    next_hop,
                    last_hop,
                    self.looking_glasses[looking_glass]['location'],
                    self.looking_glasses[looking_glass]['source_as'],
                ]
        path_length_asn.labels(*labels).set(prefix_data['path_length'])
        as_path_prepends_last_hop_asn.labels(*labels).set(prefix_data['prepend_length'])
        current_path_length_metrics.append(labels)
```

Global Rollout
Looking Glasses

- Grafana visualization
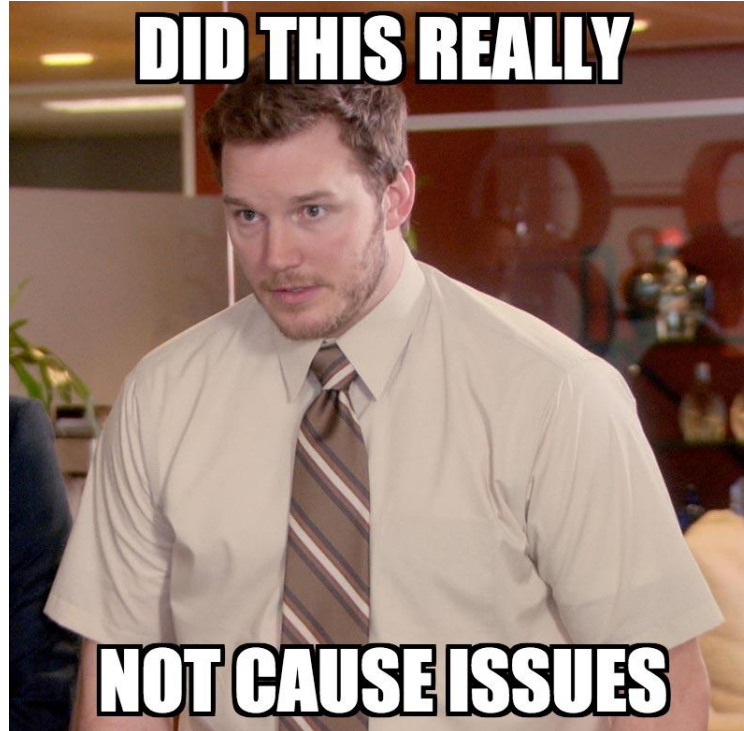- Track change in near-real-time

Global Rollout
Combined

- Traffic
- Prepends

Global Rollout
Takeaways

Global Rollout
Takeaways

- Negligible customer impact

- Route fluctuations for ± 2 minutes

- Took 1 hour to complete change, 2 days to prepare

- Instantly detected and resolved minor issues

- Heavily reliant on open source tooling and community

Questions

?

tstrickx@cloudflare.com