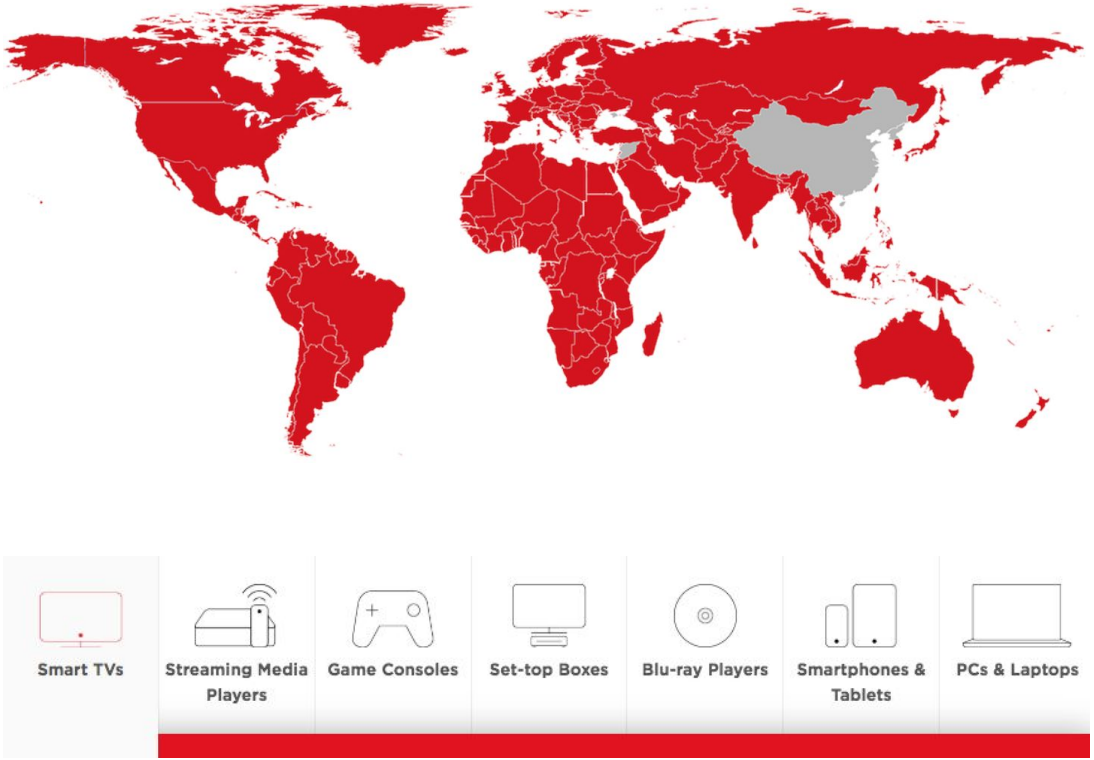
Mastering Near-Real-Time Telemetry and Big Data: Invaluable Superpowers for Ordinary SREs

Ivan Ivanov
Sr. CDN Reliability Engineer
Netflix Open Connect

NETFLIX

- 137M subscribers (Q3 2018)
- 190 countries
- 1,000s of devices types







NETFLIX DEVICE

STREAM

DATA PLANE

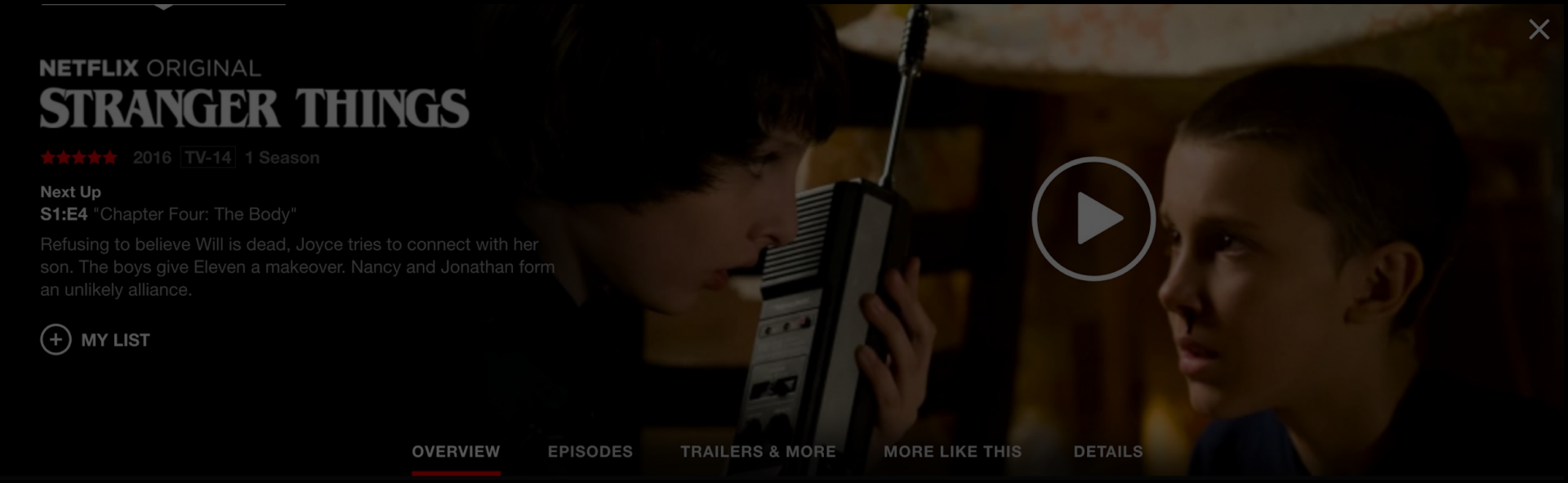
Open Connect

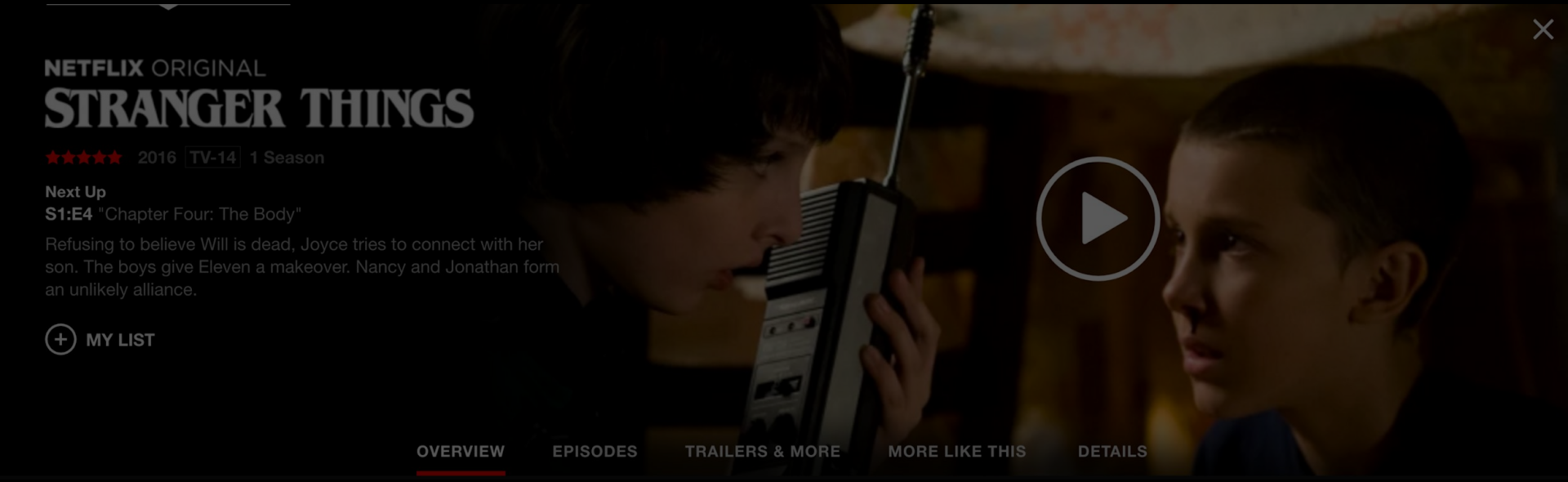
Content Delivery Network

CONTROL PLANE

powered by amazon web services™

before streaming starts = control plane = AWS





streaming = data plane = Open Connect CDN

Q: What is a Content Delivery Network?

A: Geographically distributed content servers attached to networks

+

a way of routing requests to the closest (and/or best performing) server / network path


Open Connect
CDN
A NETFLIX ORIGINAL
RESTRICTED AREA
NO TRESPASSING
U.S. GOVERNMENT PROPERTY

Netflix Open Connect CDN

- Purpose built for Netflix use case
- Globally distributed sites/pops (1,000s of locations)
- Connecting directly to residential ISP networks (>95% of our traffic is directly connected)
- Highly optimized commodity hardware
- Purpose built Open Connect backbone

Internet Exchange Participation

Country	IXP	IPv4	IPv6	Speed
AE	UAE-IX Dubai	185.1.8.52 185.1.8.53	2001:7f8:73::b5a:0:1 2001:7f8:73::b5a:0:2	20G 20G
AT	VIX Vienna	193.203.0.9 193.203.0.203	2001:7f8:30:0:1:1:0:2906 2001:7f8:30:0:2:1:0:2906	30G 30G
AU	MegaIX Melbourne Melbourne	103.26.71.40 103.26.71.65	2001:dea:0:30::28 2001:dea:0:30::41	10G 10G
AU	Equinix Sydney Sydney	45.127.172.200 45.127.172.237	2001:de8:6::2906:1 2001:de8:6::2906:2	20G 20G
AU	MegaIX Sydney Sydney	103.26.68.96 103.26.68.97	2001:dea:0:10::60 2001:dea:0:10::61	20G 20G
AU	IX Australia NSW Sydney/NSW	218.100.52.113 218.100.52.114	2001:7fa:11:4:0:b5a:0:1 2001:7fa:11:4:0:b5a:0:2	20G 20G
BE	BNIX Brussels	194.53.172.31 194.53.172.32	2001:7f8:26::a500:2906:1 2001:7f8:26::a500:2906:2	10G 10G
BR	IX.br (PTT.br) Fortaleza Fortaleza/CE	200.219.146.55 200.219.146.56	2001:12f8:0:9::55 2001:12f8:0:9::56	70G 70G
US	DE-CIX New York New York	206.130.10.81 206.130.10.122	2001:504:36::b5a:0:1 2001:504:36::b5a:0:2	10G 10G
US	Equinix New York New York	198.32.118.176	2001:504:f::2906:1	20G
US	NYIIX New York	198.32.160.90 198.32.160.189	2001:504:1::a500:2906:1 2001:504:1::a500:2906:2	20G 20G
US	Digital Realty New York New York/New Jersey	206.126.115.225	2001:504:17:115::225	20G
US	NWAX Portland	198.32.195.56 198.32.195.66	2001:478:195::56 2001:478:195::66	30G 30G
US	Equinix San Jose San Jose	206.223.116.133	2001:504:0:1::2906:2	20G
US	SIX Seattle Seattle	206.81.80.222 206.81.80.223	2001:504:16::b5a 2001:504:16::223:0:b5a	100G 100G
US	TPAIX Tampa	206.108.114.10	2001:504:3c::a	10G
ZA	NAPAfrica IX Johannesburg Johannesburg	196.60.8.80 196.60.8.100	2001:43f8:6d0::80 2001:43f8:6d0::100	60G 60G







https://openconnect.netflix.com/

Open Connect Appliance - Software



Design goals

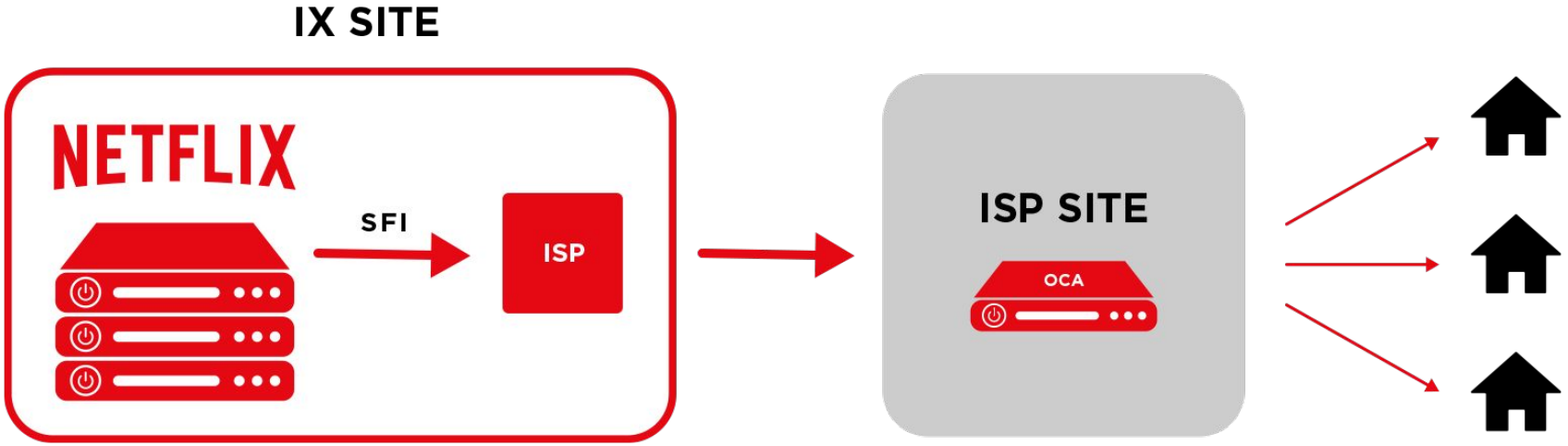
- Use of open source software
- Ability to efficiently read from disk and write to network sockets
- High-performance HTTP(S) delivery
- Ability to gather routing information via BGP

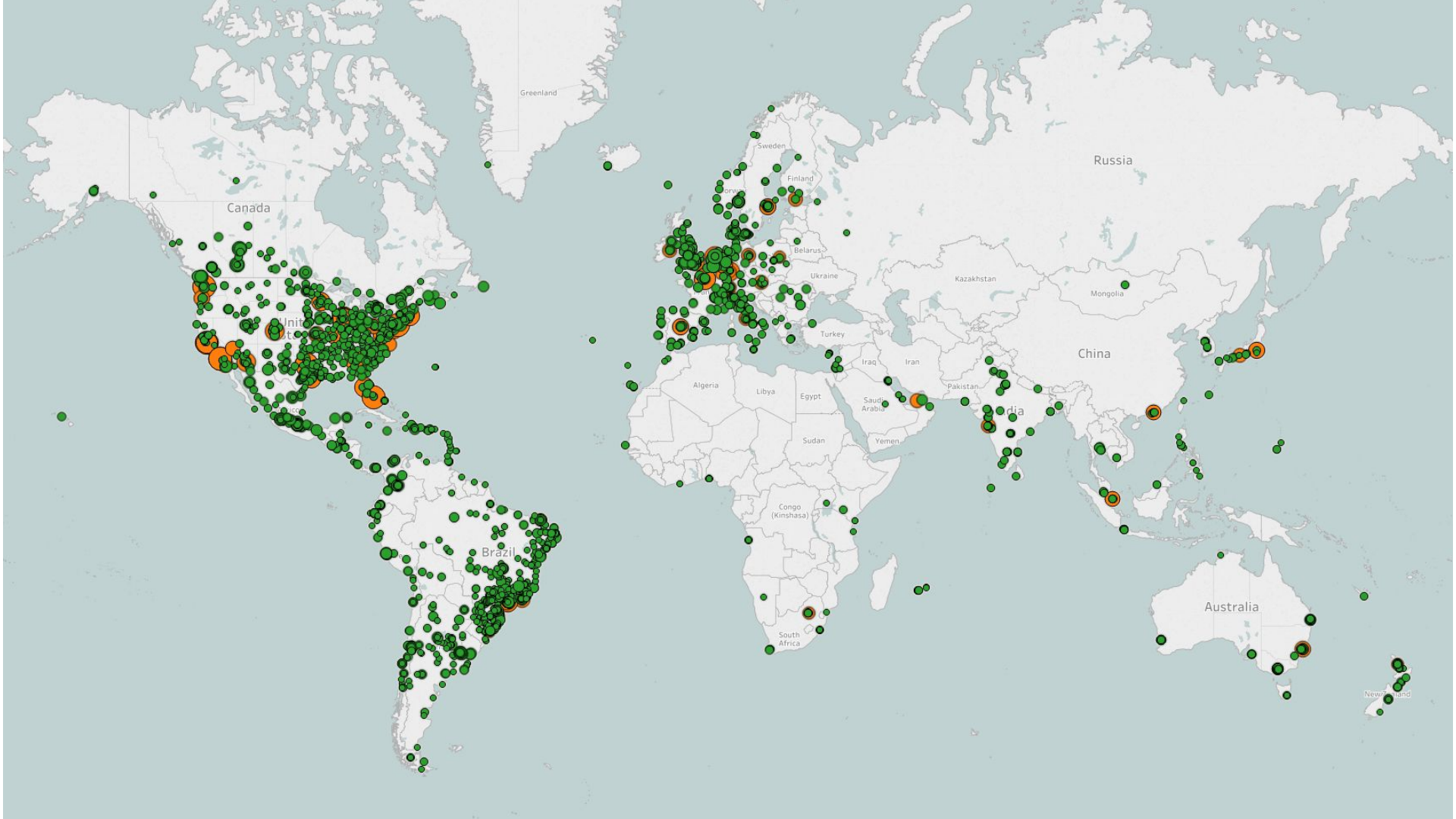
Components

- Operating System: FreeBSD-CURRENT - https://www.freebsd.org
- Web Server: NGINX - http://nginx.org
- Routing Intelligence Proxy: BIRD internet routing daemon - https://bird.network.cz

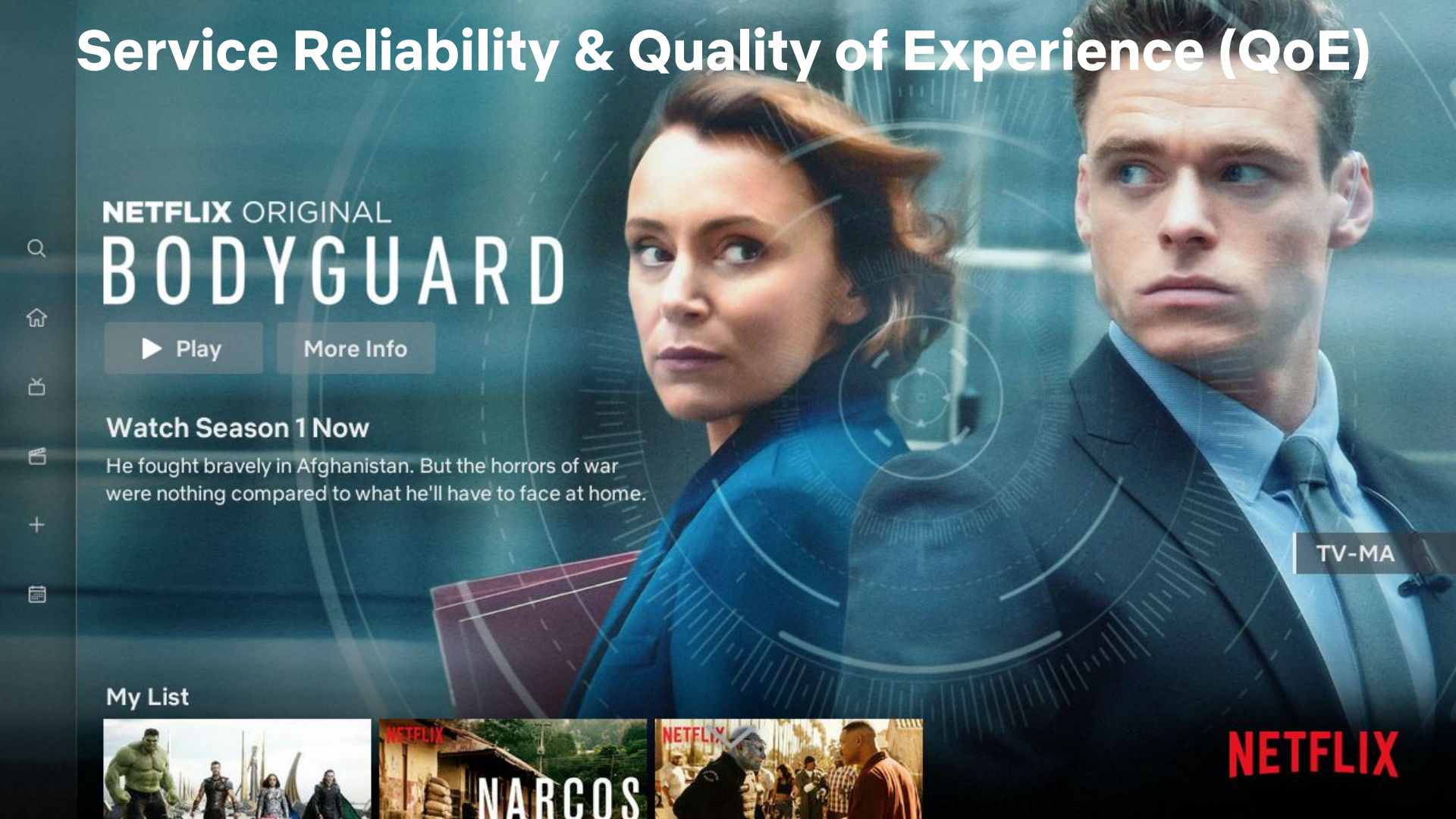
Performance

- “Serving 100 Gbps from an Open Connect Appliance” https://medium.com/netflix-techblog/serving-100-gbps-from-an-open-connect-appliance-cdb51dda3b99


IX SITE
NETFLIX
SFI
ISP
ISP SITE
OCA


Greenland
Canada
Russia
Sweden
Finland
Belarus
Ukraine
Kazakhstan
Mongolia
Turkey
China
Iraq
Iran
Pakistan
Algeria
Libya
Egypt
Saudi Arabia
Sudan
Yemen
Congo (Kinshasa)
Brazil
South Africa
Australia

Service Reliability & Quality of Experience (QoE)

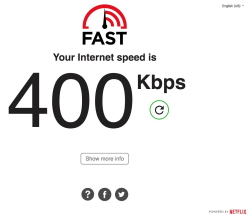


Quality of Experience (QoE)

Video Quality

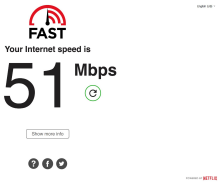
<200kbps video bitrate





>2Mbps video bitrate

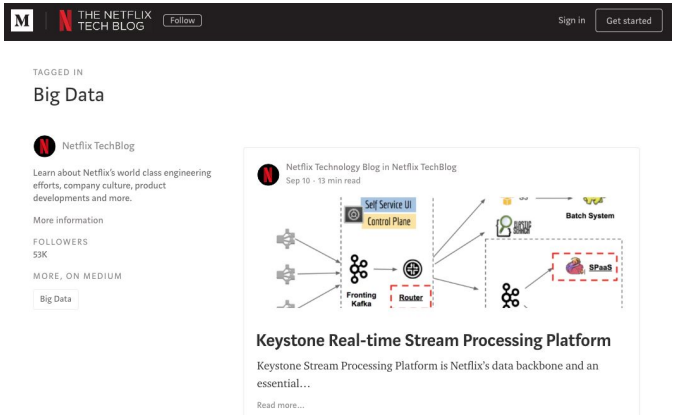




Netflix Data Platform

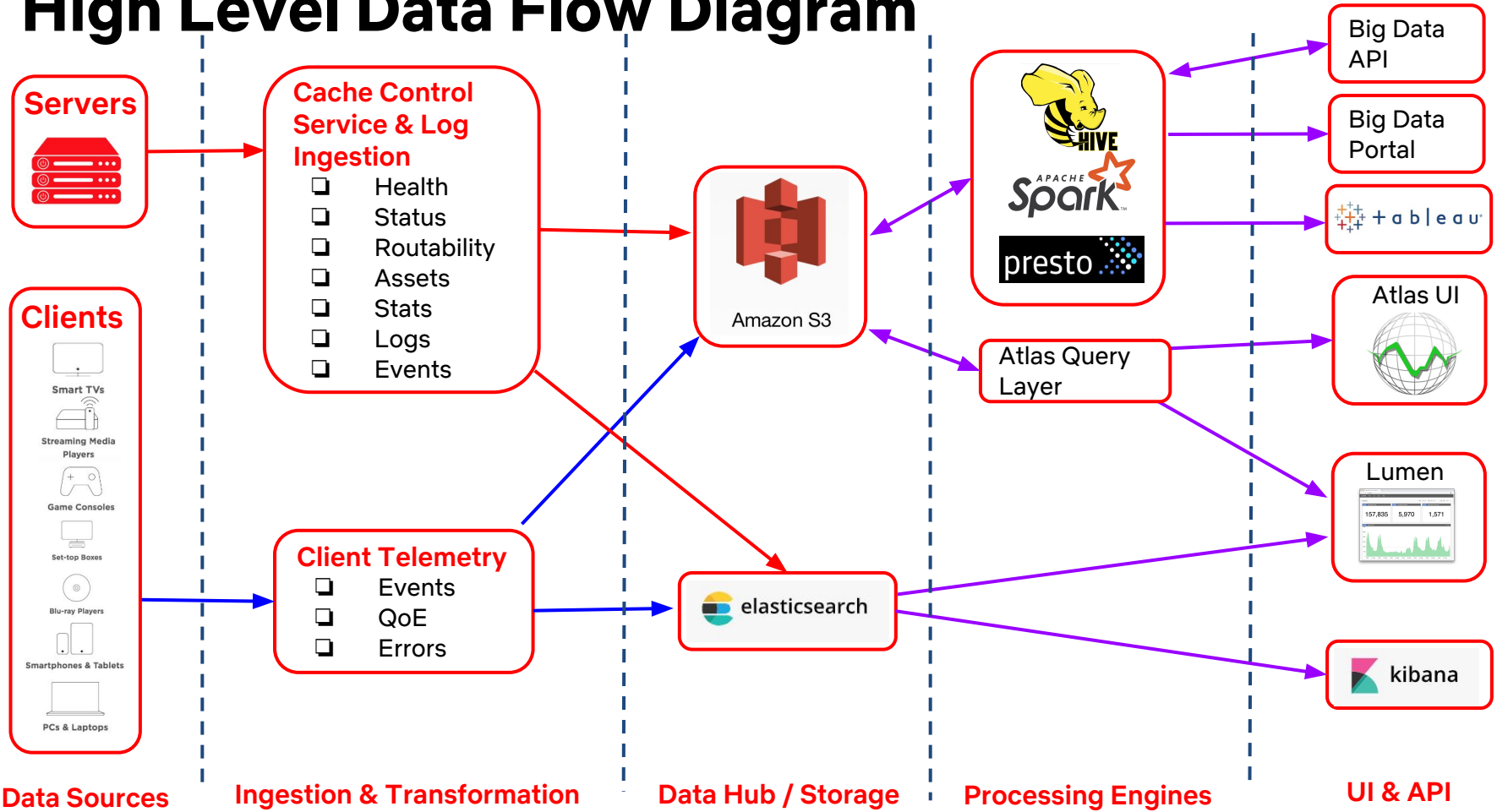
CDN telemetry system implemented on top of Netflix Data Platform:

- Data warehouse with many PB (petabytes) of data
- Hundreds of billions of events/day
- Available for use as self-service internally
- More detailed information available online in presentations and blog posts...



https://medium.com/netflix-techblog/tagged/big-data

High Level Data Flow Diagram



Atlas as a Platform for Time-Series Data

- Multi-dimensional
- Near-real-time (latency in the range of couple of minutes)
- Self-service
- Engineered for performance and scale
  - In-memory storage (fast)
  - Data stored in S3
- UI for browsing data and constructing queries
- URL friendly stack language
- Dashboards & Alerts

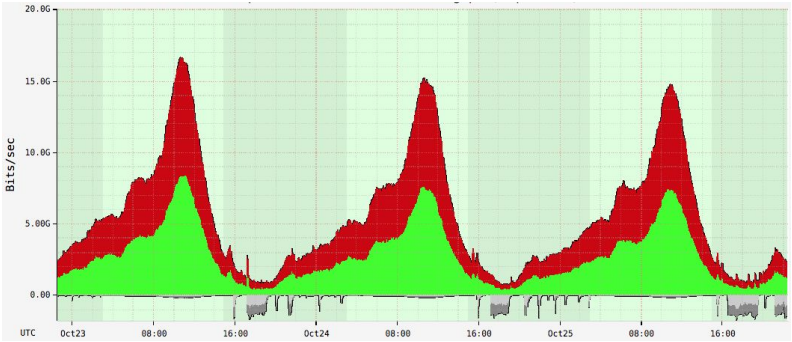


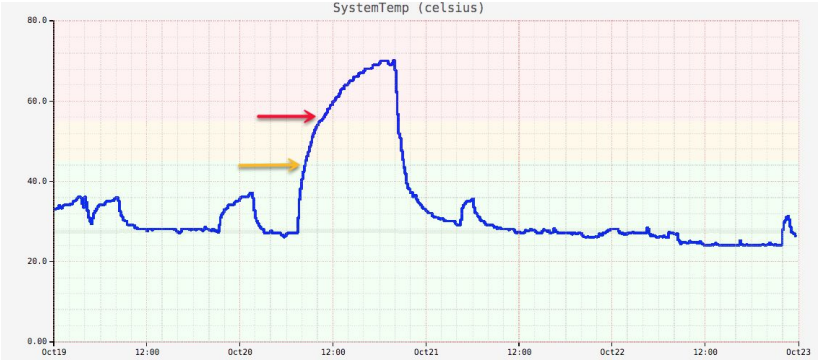
https://medium.com/netflix-techblog/introducing-atlas-netflixs-primary-telemetry-platform-bd31f4d8ed9a

https://github.com/Netflix/atlas

Atlas: CDN Related Data

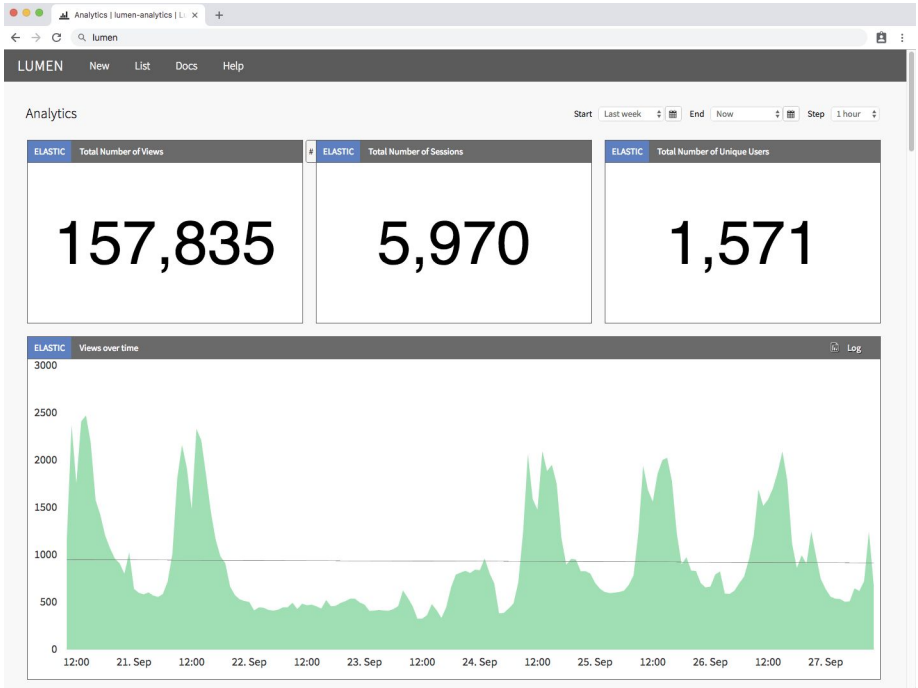
- Stats collected and uploaded by all servers
- Hundreds of counters
  - Network / memory / CPU Utilization...
  - Request per second (RPS)
  - HTTP Status Codes
  - Interface Errors
  - Bytes per protocol type (IPv4 vs IPv6)
  - Disk related counters
  - ....
- Main platform for all NRT stats collection & visualization
- UI - alerts & dashboards
- API - programmatic access





Lumen: Custom, Self-Service Dashboarding

- Easy to use: simple self-service dashboards in minutes!
- Consumes various data sources
- Variety of visualization options
- Support for variables & filtering
- Very useful for aggregation of data on various layers



https://medium.com/netflix-techblog/lumen-custom-self-service-dashboarding-for-netflix-8c56b541548c

Longer Term Multi-Dimensional Data: Aggregation, Queries & Visualization

Main NRT (near-real-time) data platform usage:

- Multi-dimensional (limited cardinality) time series in Atlas + Lumen
- *“Event-like”* data in Elasticsearch + Kibana - event data with high level of verbosity available for search & visualization
- Limited time frame (purely practical reasons)

Big Data platform:

- Large quantities of verbose data from server and client logs
- Longer term storage needs for some of this data (months, years, or “forever”)
- The need to aggregate and join data from various sources

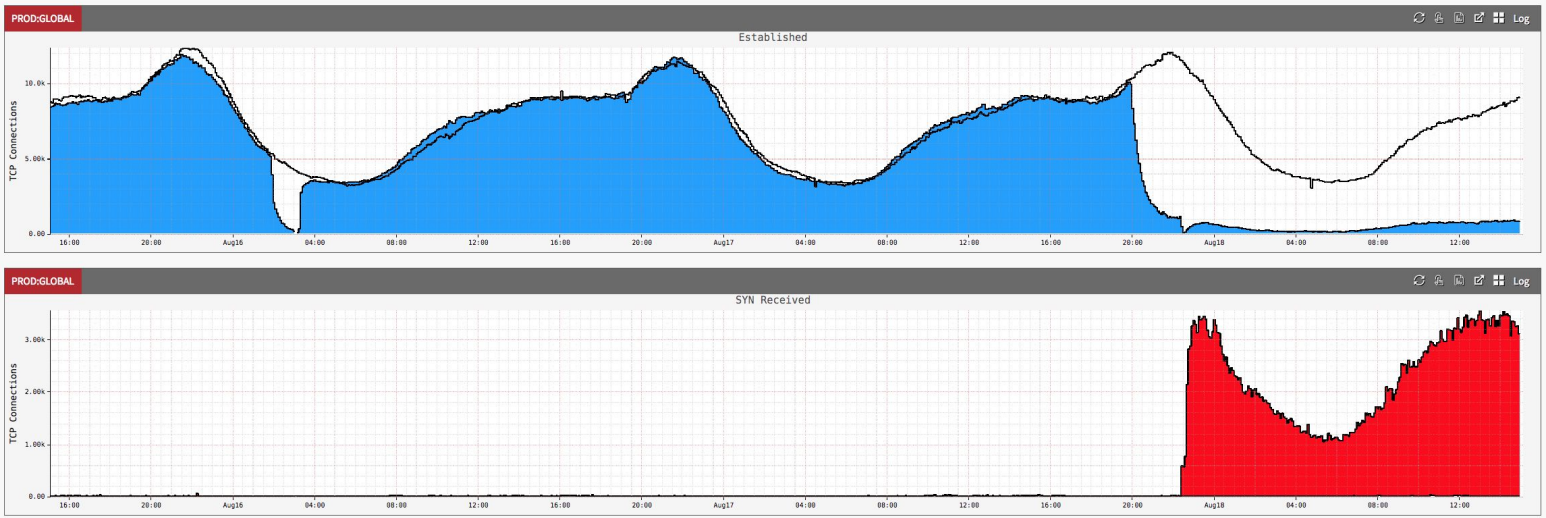
Longer Term Multi-Dimensional Data: Aggregation, Queries & Visualization

- Main data hub and source of truth in S3 + Hive metastore
- Big data transformations and queries via:
  - Scheduled background asynchronous jobs; ETL; etc… - in Hive and Spark
  - Presto - mostly for interactive queries and data extracts
- Tableau - visualization layer; scheduled reports; dashboards
- Let’s take a look on some examples next...

“What happened to my traffic?”

Relatively small partner site with unexpected shift in traffic. Servers are healthy. Connectivity in general is good. Configuration and route advertisements are ok.

Quick look on Lumen dashboard with detailed TCP connection stats shows large number of connections in half-open state.



Yes, ***broken return path*** for large portion of prefixes.

Site Move

We had an earlier case of site migration in one of our facilities. Peering and traffic for multiple partners migrated from the old location to a new location in the same area.
During the migration, as part of validation we identified noticeable degradation in performance, but only for a small subset of ASNs (Autonomous System Numbers).
Hmmm… sooo, something, somewhere is not right, but what exactly?

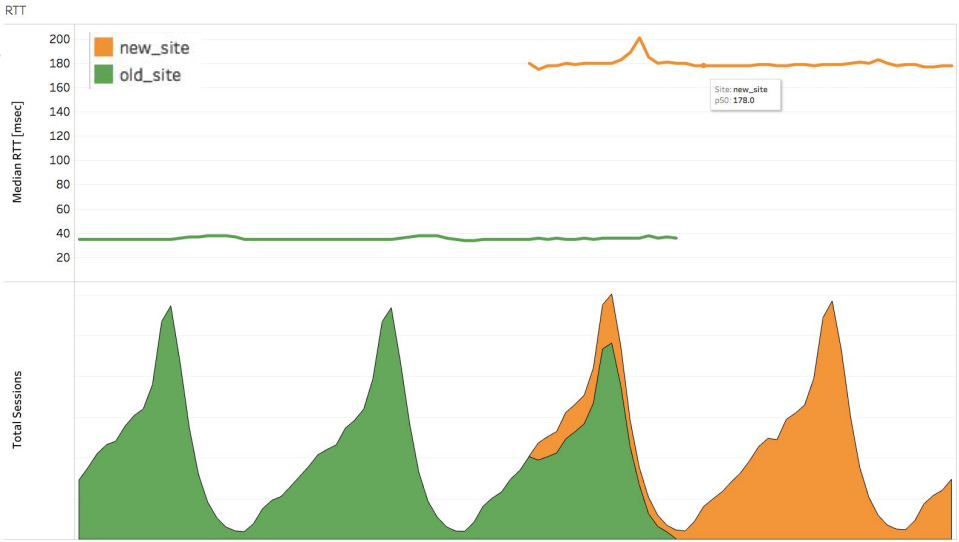
- Server related issues? - *Nope*
- Increase in retransmits? - *Nope*
- Physical connectivity or bad optics/patch panel/cables? - *Possibly; but not likely since there is no increase in interface errors*
- So, *something else?*

Site Move

How about latency?

Presto RTT

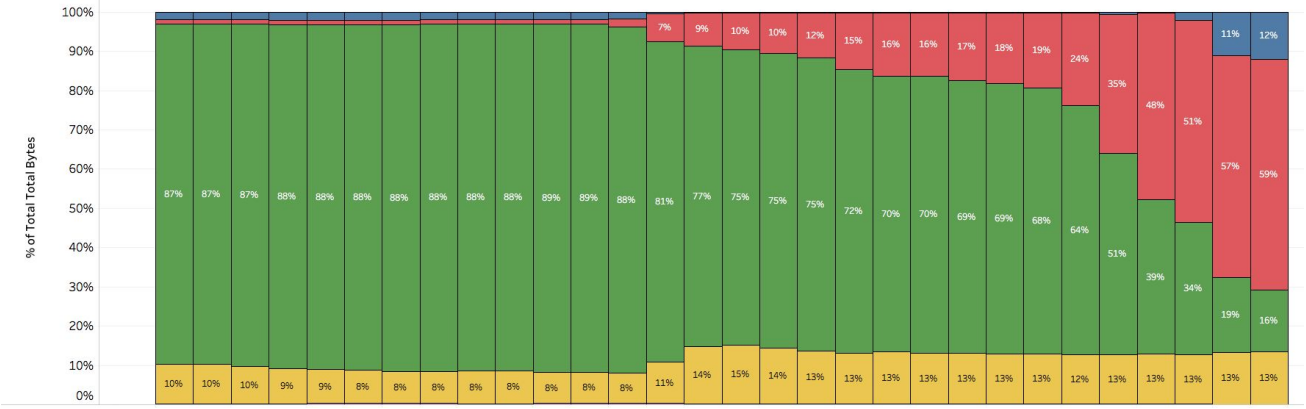
```
1  select
2      utc_dateint
3      ,utc_hour
4      ,site_name
5      ,client_asn
6      ,approx_percentile(tcp_rtt_ms,0.5) as median_RTT
7      ,count(*) as total_sessions
8
9  from db.table_name                  --placeholder db and table_name
10
11 where
12     where utc_dateint between 20181020 and 20181023
13     and site_name = 'site_name'      --placeholder site_name
14     and client_asn = '1234'          --placeholder ASN
15
16 group by 1,2,3,4
17
```



Yes, clearly. Closer look on network path shortly after that uncovered incomplete network configuration on one of the partner peering links.

Tableau + Big Data

- Dashboards and reports based on aggregated data from different sources
- Usually based on scheduled Tableau data extracts done daily/hourly (as needed)
- Very few cases (for lightweight queries) directly - e.g. Presto connection
- Very helpful for long term traffic trends, utilization, capacity, error trends, forecasting, modeling, hardware status and failures, etc…



Example - daily aggregation of data for encoding changes overtime for specific device category

Customer Errors - fictional screenplay based on past events

Fade In

You are on-call. Customer support team is pinging you about increased volume of calls from customers of *XYZ* ISP, with specific category of devices intermittently failing to start playback with error code *1234:*

> *“Are you aware of any current issues that could cause this?”*

You:

> *“Hmmm... not really, but let me take a look... Yes, so everything looks green on my side. It’s gotta be something outside of our network. Can we get some more details - sample client IPs, HTTP or possibly Wireshark captures?*

Well, this can take a while. Maybe you can proceed with Stage 2 of the investigation in parallel...

Customer Errors - Stage 2, Option 1

1. Use your ***psychic abilities*** to identify and locate impacted customers
2. ***Run (or fly) really fast***, ring the doorbell, introduce yourself, use your smartwatch/phone/tablet/laptop to take the captures.
3. Analyze the captures, correlate the data, identify the issue.
4. Mitigate the issue.
5. Fix the root problem.

   5.a. (optional) If the problem is fiber cut use your ***laser vision*** to quickly splice the damaged fiber cables.

Well… if you only had such *superpowers*. You will need to wait… or maybe there is another option…

Customer Errors - Stage 2, Option 2

1. Launch a new real-time query on top of Mantis stream-processing platform that selects the desired client events for devices from the desired category scoped to the ASN in question

```
SELECT * FROM stream WHERE e["device"] == "1111" && e["asn"] == "2222" && e["errorcode"] ==~ /.*3333.*/
```

2. Couple of minutes later you have the first verbose data in UI and Elasticsearch.
3. Five minutes later you have enough data points to see the common pattern and identify the issue - bad connectivity between specific IP ranges outside our network and subset of servers in a specific location.
4. Mitigate the issue - reroute traffic in scope away from the impacted servers.

*Mantis post: https://medium.com/netflix-techblog/stream-processing-with-mantis-78af913f51a6

Customer Errors - Epilogue

Good! We fixed the problem!

But wait... to be in this situation is not really *“good”:*

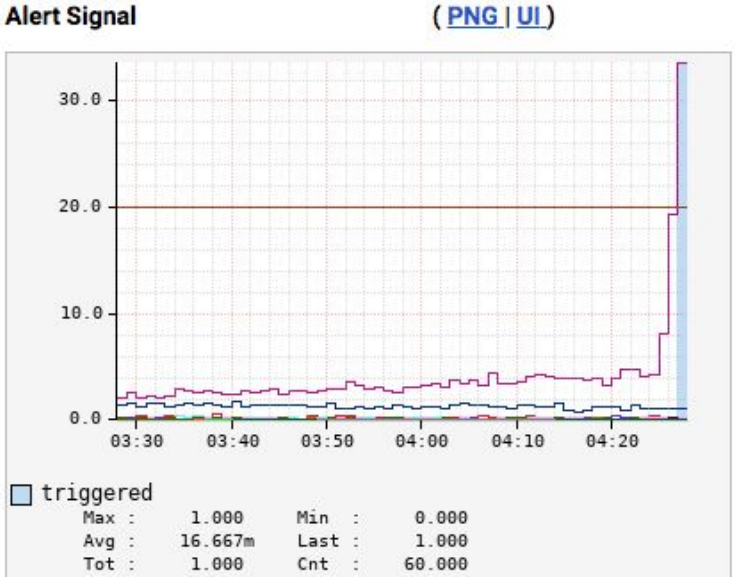
TTD (Time to Detect): ***unknown...*** possibly ***couple hours??***
TTR (Time to Recover): **~10-12 min**

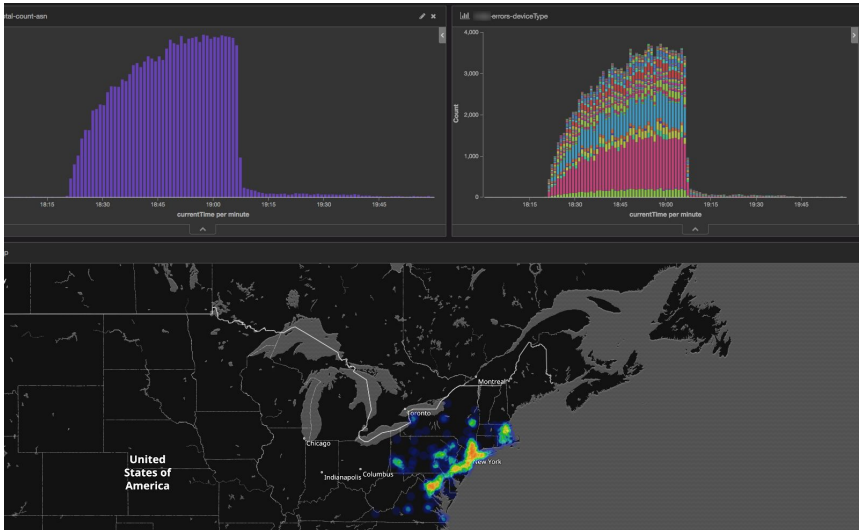
So, we will need to make the client server communications for this data flow more resilient. In parallel, for the next time when something like this happens we need to improve:

- Already have a job running all the time, that collects data for such issues and stores it in Elasticsearch
- Make a Kibana dashboard that shows the data and allows search; visualizations, etc…
- Implement an Atlas metric, and trigger alert based on it when this happens
- Consider implementing auto-mitigation

Customer Errors - Epilogue

TTD (Time to Detect): **2-3 min**
TTR (Time to Recover): **~5-6 min** *(*when actionable on service side)*





Takeaway

- Near-real-time collection and aggregation on data from all applicable layers - fundamental requirement for having the complete end-to-end picture of your environment
- Capability to correlate related events from different sources - extremely helpful for really getting the holistic view of your service
- Use of real time data processing technologies can help you minimize TTD and TTR
- Longer term trends of aggregated & correlated data over long periods of time (days, months, years) will allow you to see the big picture easier, help you identify slower moving trends (see the “Boiling frog” fable), and also plan for future events.
- In case if you already have some of these technologies available in your environment, but you are not using them as part of your toolset, please go ahead and try to start using them. They can help a lot!
- I realize that you may not have these capabilities available at your disposal today, but this could be something that you can consider implementing in the future. Cloud computing vendor of your choice can help you to get this done.

Questions?

Thank you