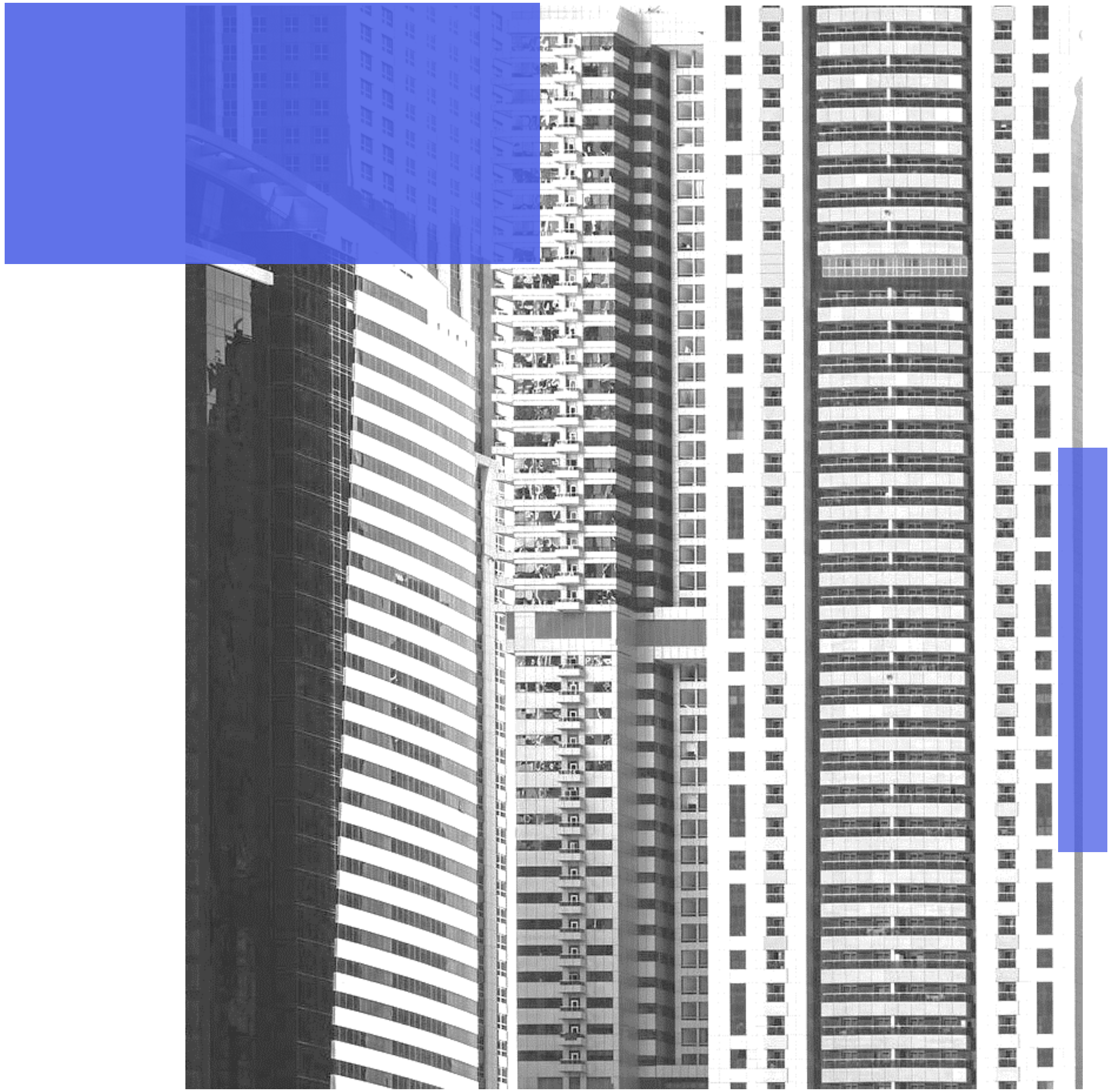
MATT(S)

GENERIC CONFERENCE PRESENTATION



CURRENT PHYSICAL DEPLOYMENT

67 ARKIME SENSORS	**6.3PB**
33 ELASTICSEARCH SERVERS	**3.1PB**
29 PACKET BROKERS	

2023

Expand

7 Arkime servers	**3.4pb**
2 Elasticsearch nodes	**0.5pb**

Upgrade

Buildout new lab environment

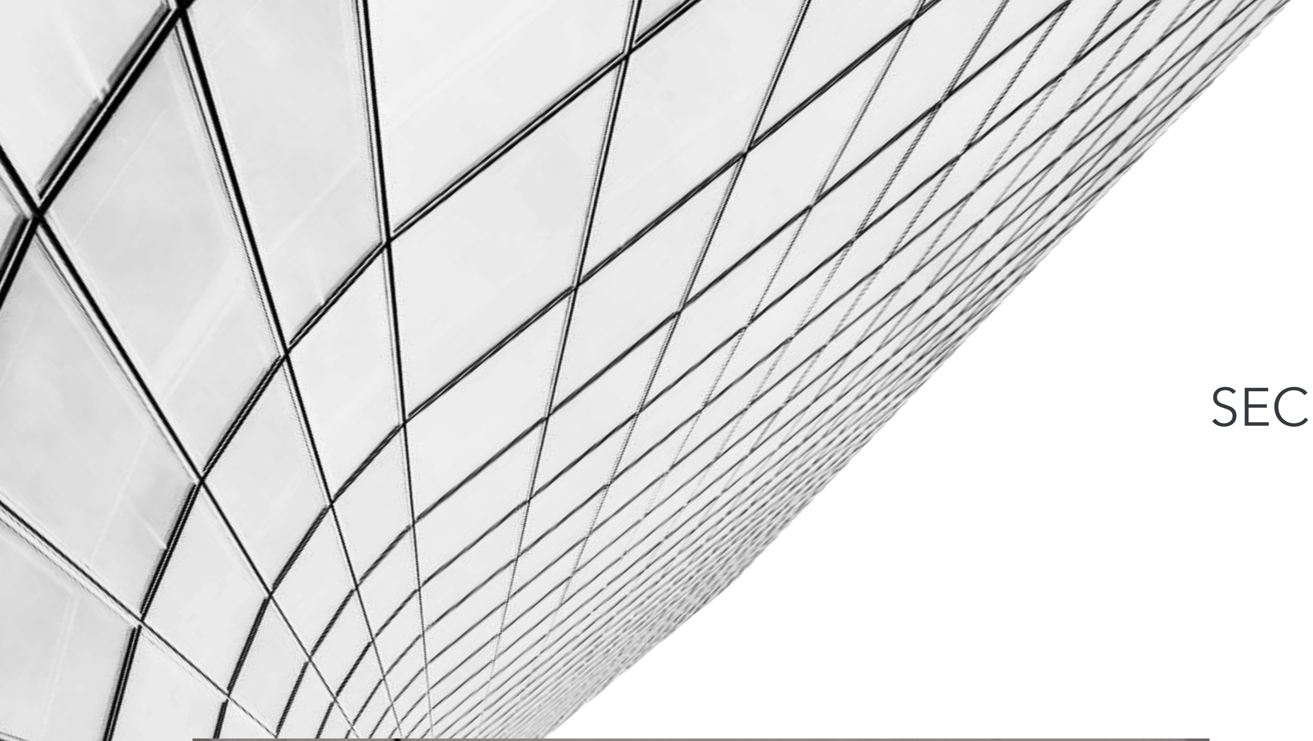
Upgrade to version 4.X

Optimize

Filtering is hard

Documentation & Automation

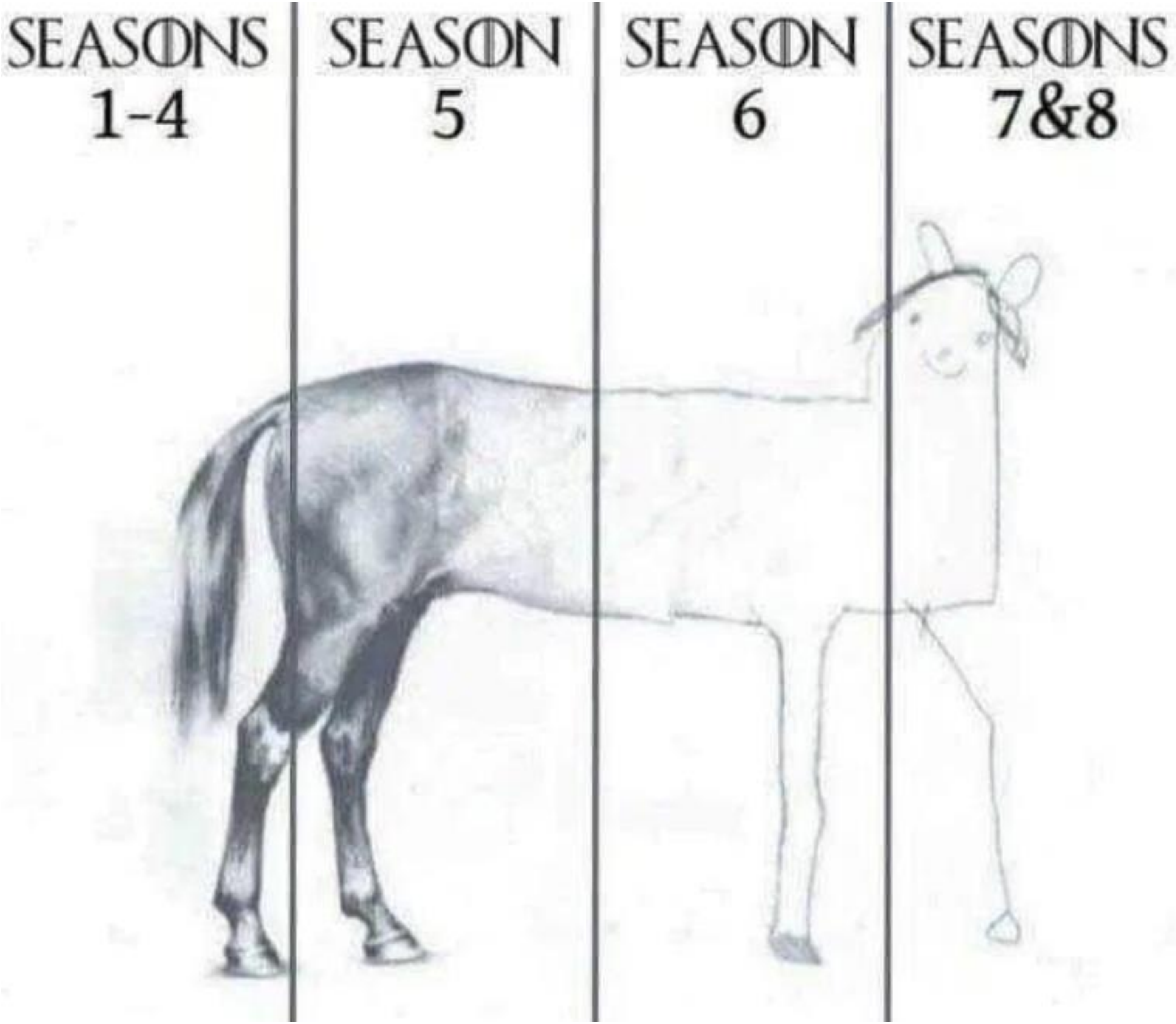
FILTERING IS HARD



SECURITY RELEVANT

POTENTIALLY SECURITY RELEVANT

NOT SECURITY RELEVANT

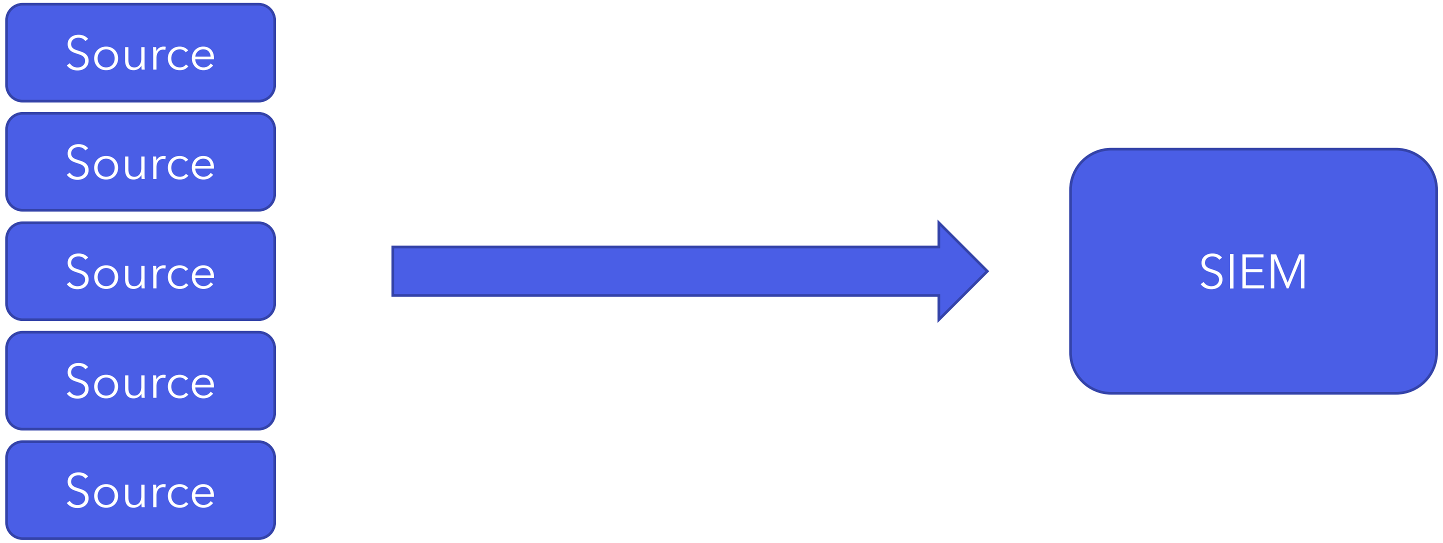

SEASONS 1-4
SEASON 5
SEASON 6
SEASONS 7&8

USER MANAGEMENT

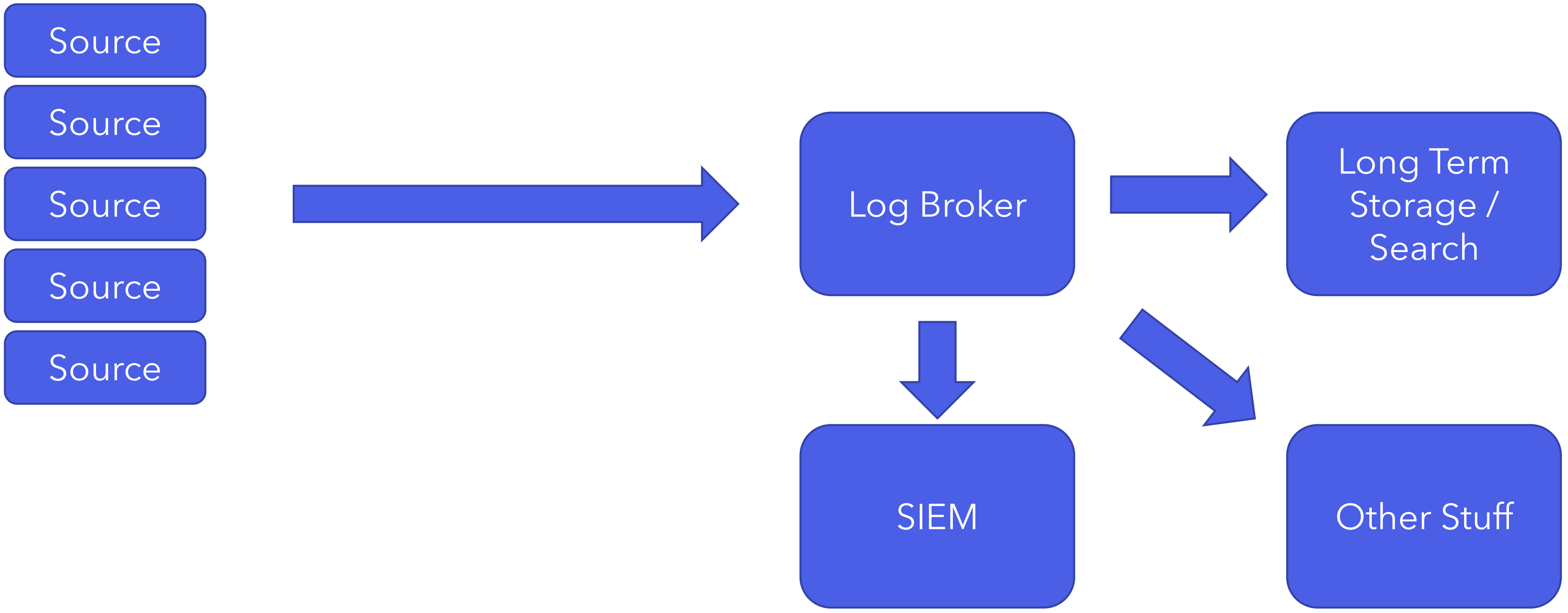
- Access is automatically provisioned from SailPoint using the Arkime API
- Authentication handled by Apache for Azure AD integration and MFA

WISE

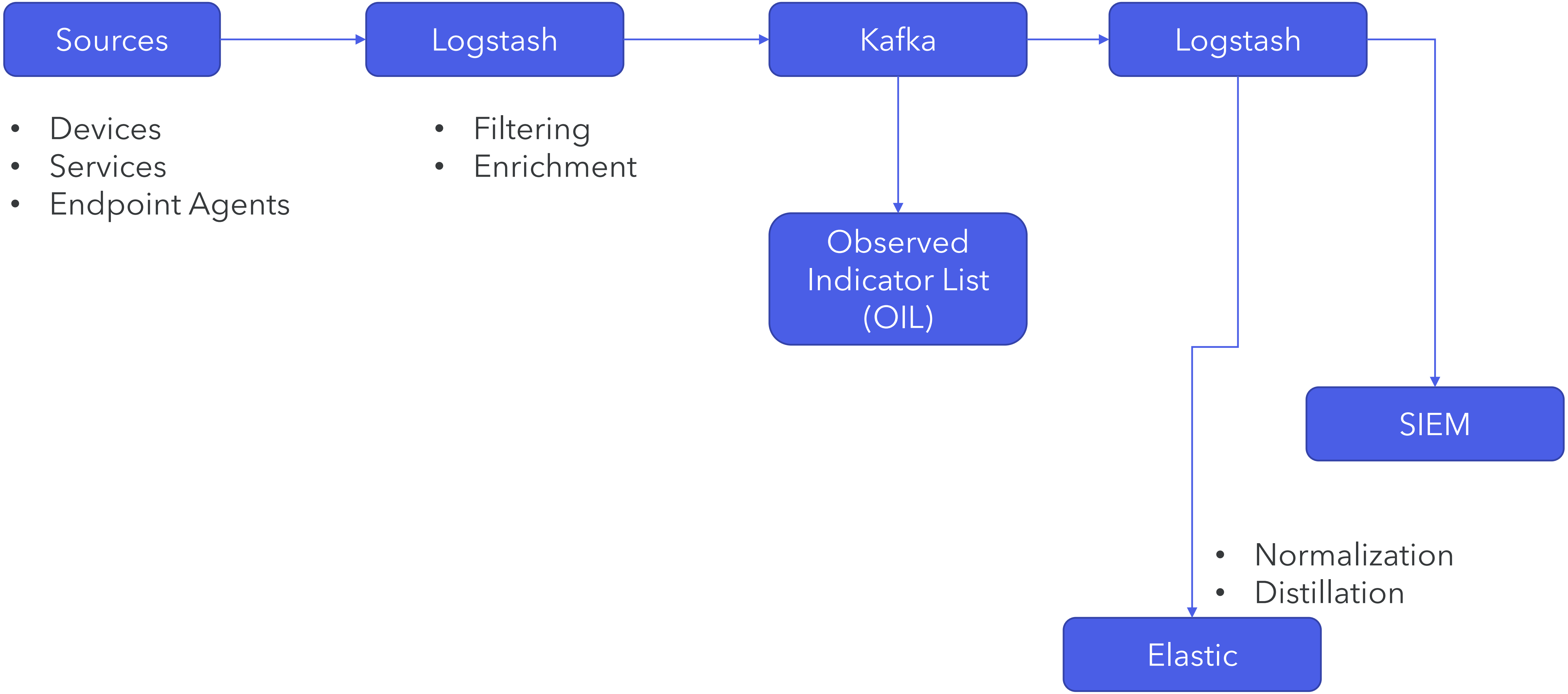
- Run Redis on each sensor
- Use dynomite (https://github.com/Netflix/dynomite) to write to all of them at once
- Load up Redis with asset data, user awareness, Newly Observed Domains, threat feeds, etc



- Slow to search
- Can’t send logs to other applications
- Retention sucks
- Giant pain in the ass to change vendors



Source
Source
Source
Source
Source
Log Broker
Long Term Storage / Search
SIEM
Other Stuff



Sources

- Devices
- Services
- Endpoint Agents

Logstash

- Filtering
- Enrichment

Kafka

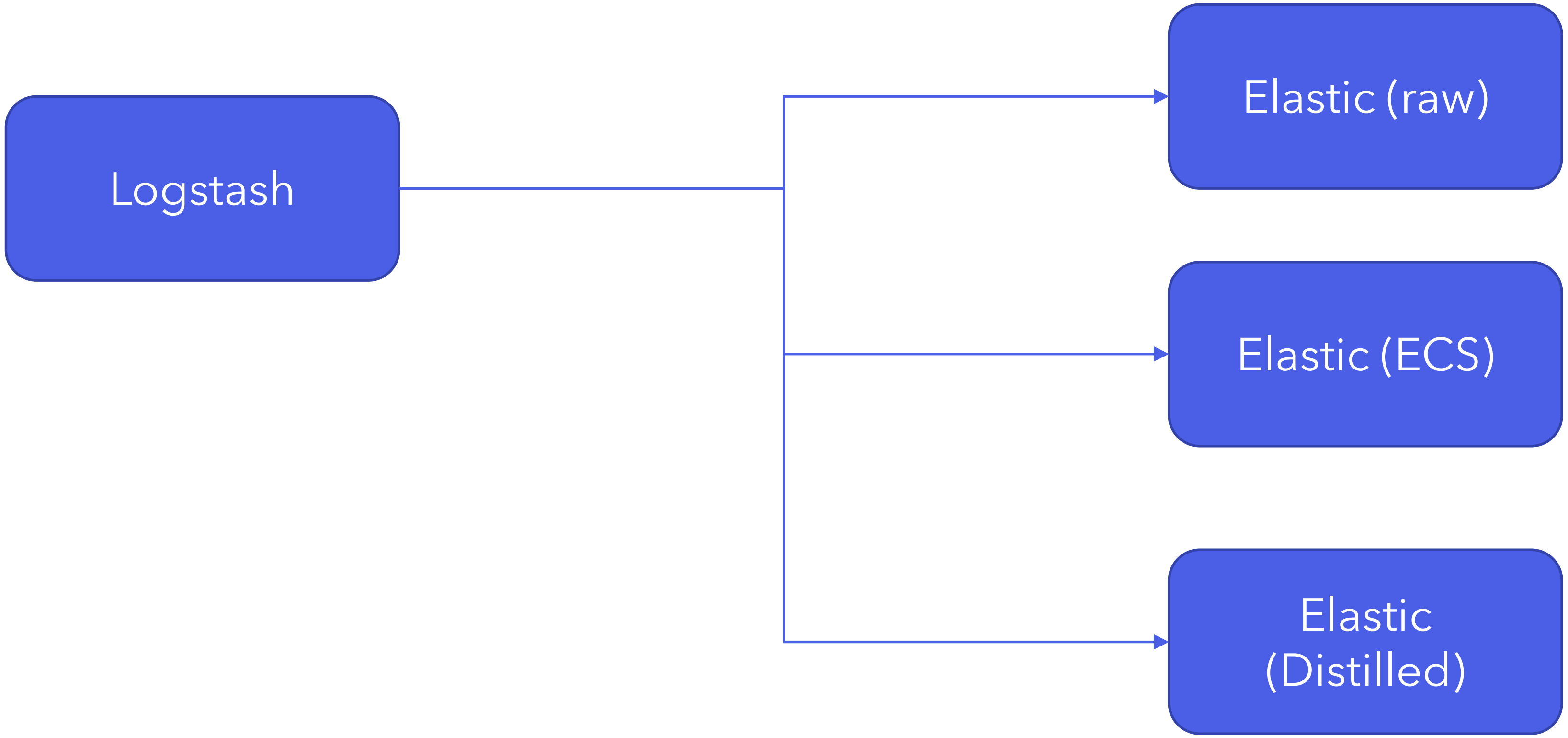
Observed Indicator List (OIL)

Logstash

SIEM

- Normalization
- Distillation

Elastic

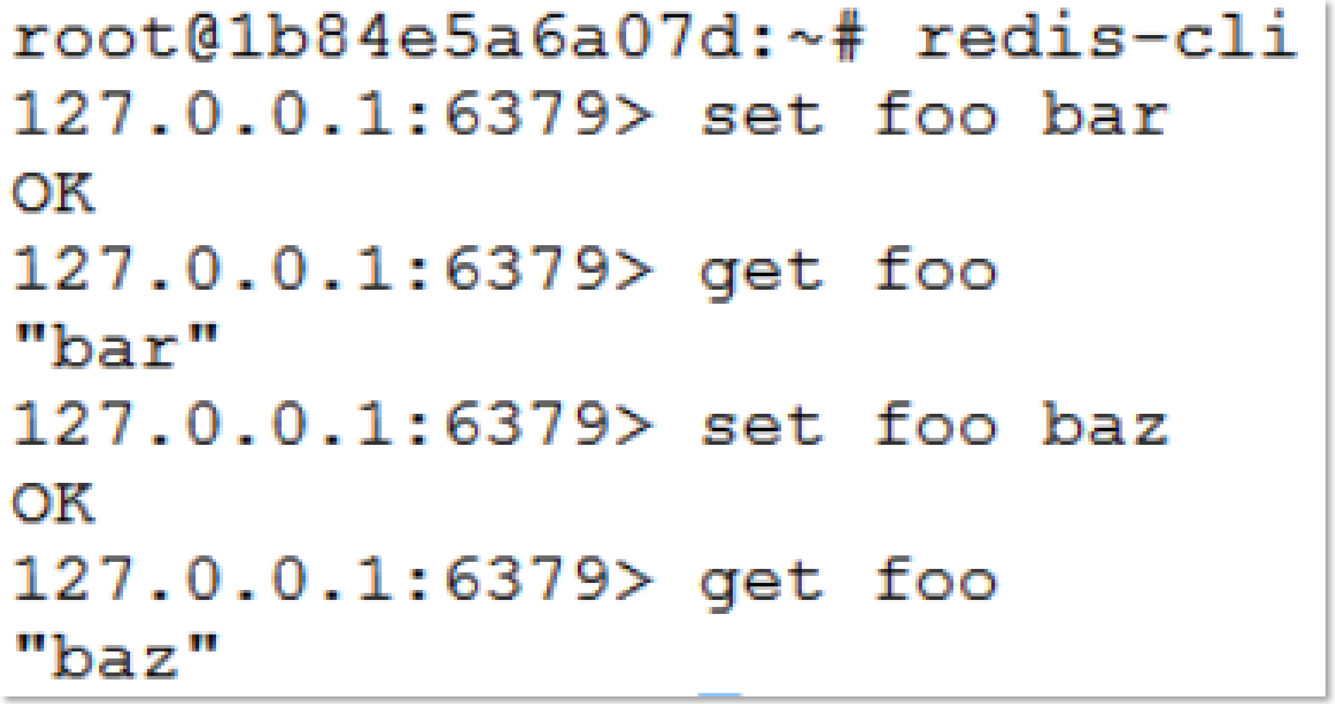

Logstash
Elastic (raw)
Elastic (ECS)
Elastic (Distilled)

DISTILLED EVENTS

- Input is a raw log (e.g. an apache web log)
- Output is an event (e.g. “customer X password reset from IP Y”)
- Events are much smaller, so we can store them for way longer and search them more quickly
- Events are normalized so we can create a unified timeline of events from a diverse set of sources

REDIS

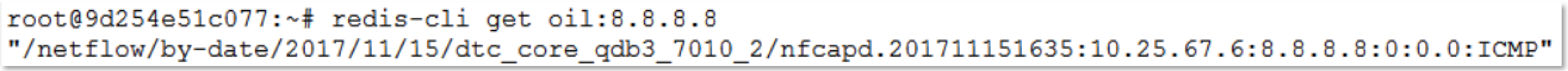
- Redis is an O(1) key / value store
- Data resides in memory, but also writes to disk at configured time intervals



```
root@1b84e5a6a07d:~# redis-cli
127.0.0.1:6379> set foo bar
OK
127.0.0.1:6379> get foo
"bar"
127.0.0.1:6379> set foo baz
OK
127.0.0.1:6379> get foo
"baz"
```

OBSERVED INDICATOR LIST (OIL)

- Every time we see a flow, shove it in a Redis database, using the IPs as keys
- Redis then contains the most recent time we’ve seen any given flow
- When we want to know if we’ve ever seen an indicator in our environment, we just check the OIL



```
root@9d254e51c077:~# redis-cli get oil:8.8.8.8
"/netflow/by-date/2017/11/15/dtc_core_qdb3_7010_2/nfcapd.201711151635:10.25.67.6:8.8.8.8:0:0.0:ICMP"
```

MEGA OBSERVED INDICATOR LIST (MEGAOIL)

- OIL all the things!
- Key is any atomic indicator we want to find
- Value is a log entry, flow, asset record, email address, domain, or anything else
- Web interface allows us to paste in hundreds of IOCs and get results back in seconds

MEGAOIL SOURCES

- Netflow
- EDR events (TODO)
- Azure logs
- Firewall logs
- Zscaler

- Asset database
- Citrix
- Okta
- DHCP
- CoxSight

COXSIGHT

- Passively create an asset database by monitoring security logs
  - Device type
  - Hostname / IP mappings
  - Device owner(s)
- Linux and Windows authentication logs
- Port 22 / 3389 traffic from endpoint agents and firewall logs

COUNT SOCULA

- API gateway to query all of the things and return Elastic Common Schema

COUNT SOCULA ENDPOINTS

- Passive DNS
- Indicator parser
- Carbon Black Response
- GeoIP
- VPN check
- LDAP lookups
- Asset lookups