avito.tech

XD(R) своими руками

Как мы детектируем атаки на Linux-инфраструктуру и не только

Тимур Котов — SOC Team Lead
Иван Клунный — Инженер

Москва — 2023

avito.tech

Москва — 2023

Agenda

- Prerequisites
- Targets
- Events
  - Gathering
  - Delivery
  - Processing
- Detections

avito.tech

Москва — 2023

Prerequisites

- 99% Linux-infra
- Front — Azure Stack
- Back — !#@!?$ & microservices
- 2k/s vs 1,1kk/s
- Linux audit

avito.tech

Москва — 2023

Targets

- Front — Azure Stack — сохранить.
- Back — !#@!?$ & microservices — убрать вовлечённость разработки.
- 2k/s vs 1,1kk/s — процессинг и фильтрация на земле.
- Linux audit — найти замену.
- >>> унифицировать flow обработки

avito.tech

Gathering

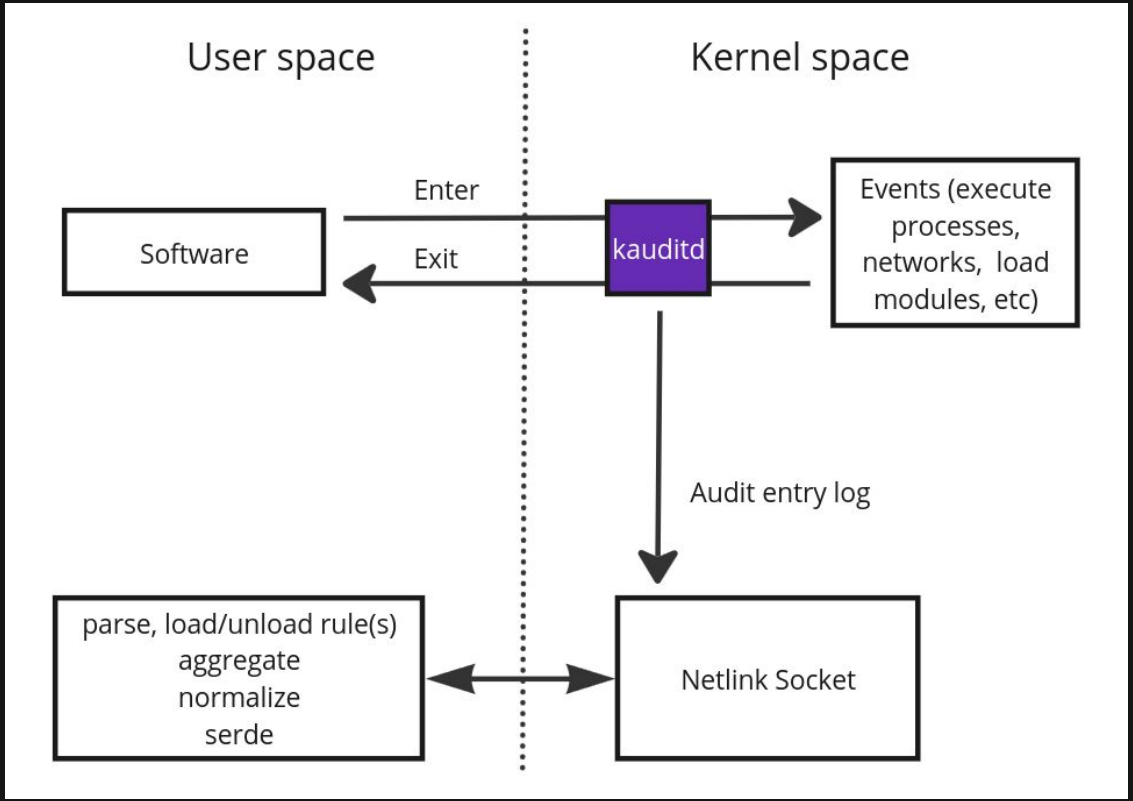
Москва — 2023

Linux audit

Подсистема ядра Linux по сбору security-related (и не только) ивентов.

Позволяет захватывать системные вызовы, сетевые или файловые события, аутентификацию.

Linux audit в Авито



Что не так с Linux audit?

01. Проблемы
с декодированием
аргументов.

Что не так с Linux audit?

01. Проблемы
с декодированием
аргументов.

```
syscall_args: ['55ba11795ca0','1ed','1','55ba11795010']
```

Что не так с Linux audit?

01. Проблемы с декодированием аргументов.

02. «Перегруженная» схема данных.

Что не так с Linux audit?

01. Проблемы с декодированием аргументов.

02. «Перегруженная» схема данных.

03. Непредсказуемое поведение на некоторых ядрах / нагрузке.

Что не так с Linux audit?

01. Проблемы с декодированием аргументов.

02. «Перегруженная» схема данных.

03. Непредсказуемое поведение на некоторых ядрах / нагрузке.

Авито не работает в крупных городах. У сайта проблемы с авторизацией

Что не так с Linux audit?

01. Проблемы с декодированием аргументов.

02. «Перегруженная» схема данных.

03. Непредсказуемое поведение на некоторых ядрах / нагрузке.

04. Заметное performance penalty.

Что не так с Linux audit?

01. Проблемы с декодированием аргументов.

02. «Перегруженная» схема данных.

03. Непредсказуемое поведение на некоторых ядрах / нагрузке.

04. Заметное performance penalty.

Есть ощущение, что оно тормозит из-за аудита

Альтернативы?

Альтернативы?
eBPF?
Kmod?

- Sysdig
- Falco
- Tracee
- SysFlow

Linux eBPF

Runtime (среда исполнения) в контексте ядра для решения огромного спектра задач.

eBPF предоставляет интерфейс для инструментации ядра Linux, приложений.

Почему eBPF?

01. Максимальная гибкость: мы можем заложить произвольную логику в eBPF-пробы.

Почему eBPF?

01. Максимальная гибкость: мы можем заложить произвольную логику в eBPF-пробы.

02. Верификация загружаемых программ.

Почему eBPF?

01. Максимальная гибкость: мы можем заложить произвольную логику в eBPF-пробы.

02. Верификация загружаемых программ.

03. eBPF это не только про security.

Почему eBPF?

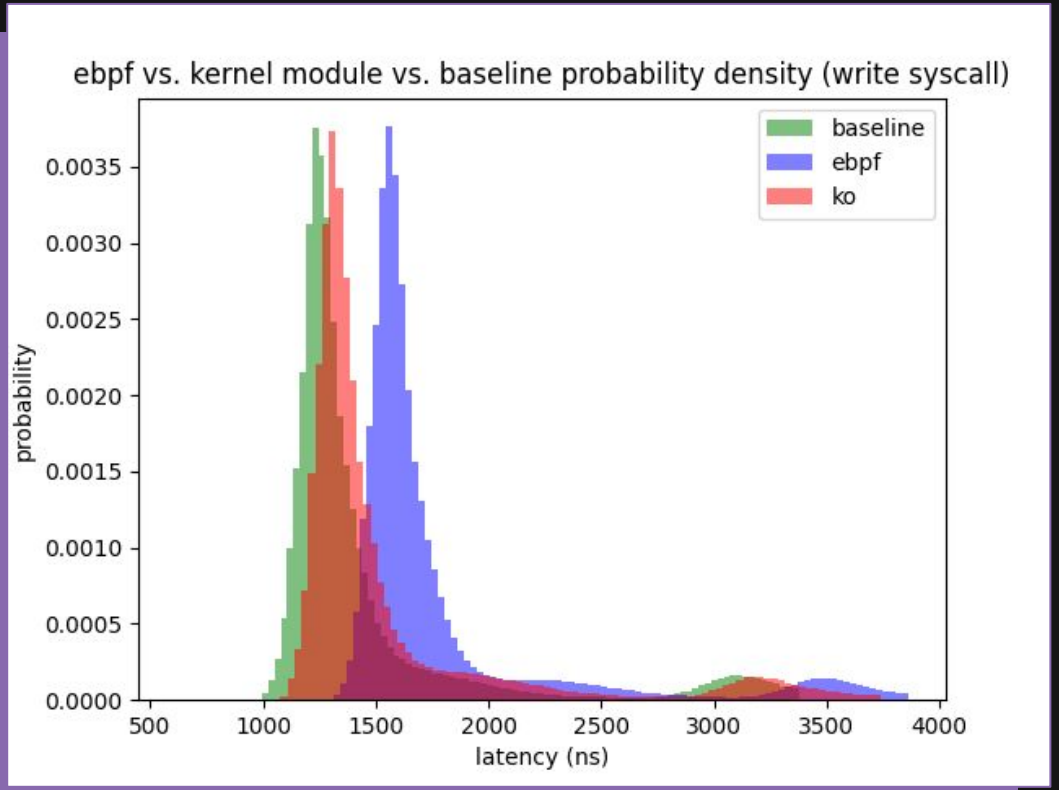
01. Максимальная гибкость: мы можем заложить произвольную логику в eBPF-пробы.

02. Верификация загружаемых программ.

03. eBPF это не только про security.

04. Хорошая производительность (не бесплатно).

eBPF vs Kmod performance

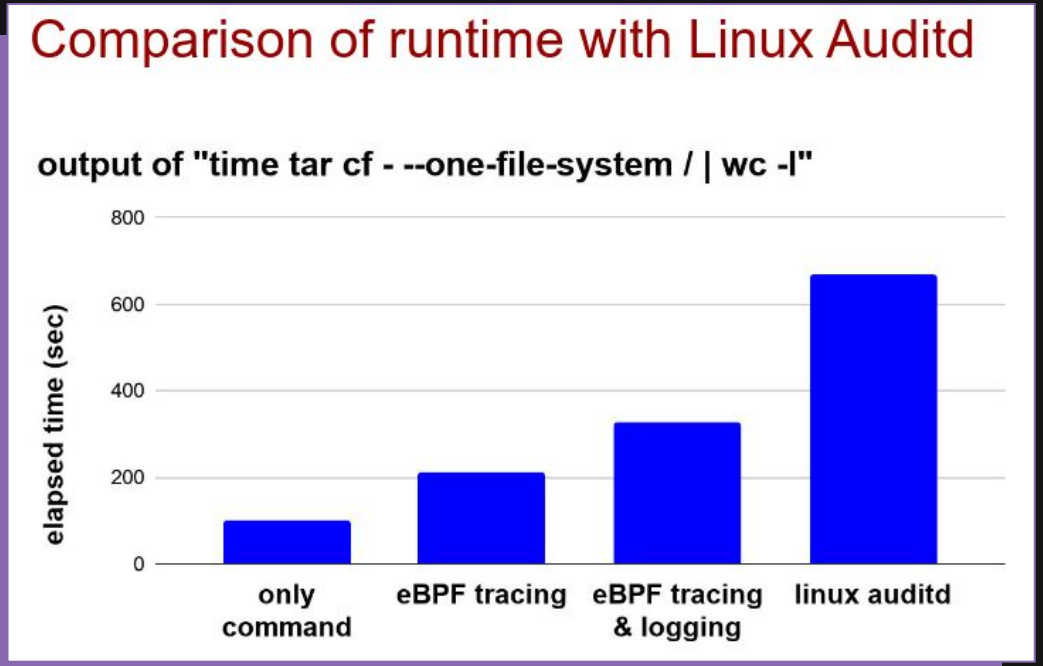


https://github.com/falcosecurity/libs/files/8334709/Falco.eBPF.Kernel.Module.Performance.pdf

eBPF vs Kmod performance

syscall	events #	Measurement	p95 (ns)
write	24024256	Baseline	3456
		eBPF	4193
		Kernel module	3687

eBPF vs Audit



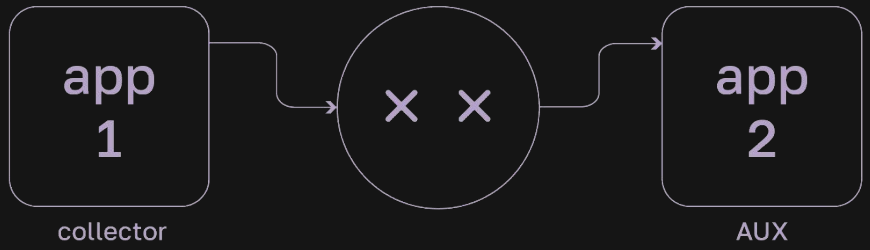
https://www3.cs.stonybrook.edu/~grd/GRD-2021/posters/GRD_2021_Poster-Rohit_Aich.pdf

Почему SysFlow

- Container awarness: есть интеграция с K8S (experimental).
- Объектно-реляционная модель с приятными абстракциями (sysflow / netflow / fileflow).
- Фильтрация событий в стиле falco rules.
- Возможность собирать агрегаты на хосте — sysflow-processor*.
- Эволюция и версионирование схемы данных.
- API для Go, Python, C++.

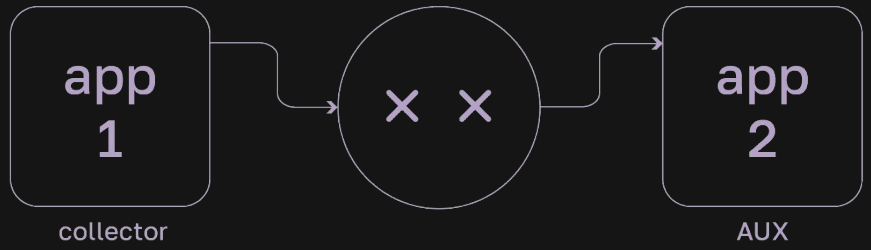
SysFlow в Авито

- Написали In-house демон: клей между коллектором и остальной частью пайплайна .
- Кастомизировали схему данных.
- Обогатили метаданными.
- Поменяли формат сериализации данных: AVRO → MessagePack.



Gathering

Визуализация



avito.tech

Delivery

Москва — 2023

Data delivery

Факты

01. Kafka в качестве брокера сообщений.

Data delivery

Факты

01. Kafka в качестве брокера сообщений.

02. Локальные и глобальные (across DC) инсталляции.

Data delivery

Факты

01. Kafka в качестве брокера сообщений.

02. Локальные и глобальные (across DC) инсталляции.

03. Kafka-connect для связи с внешними системами.

Data delivery

Факты

01. Kafka в качестве брокера сообщений.

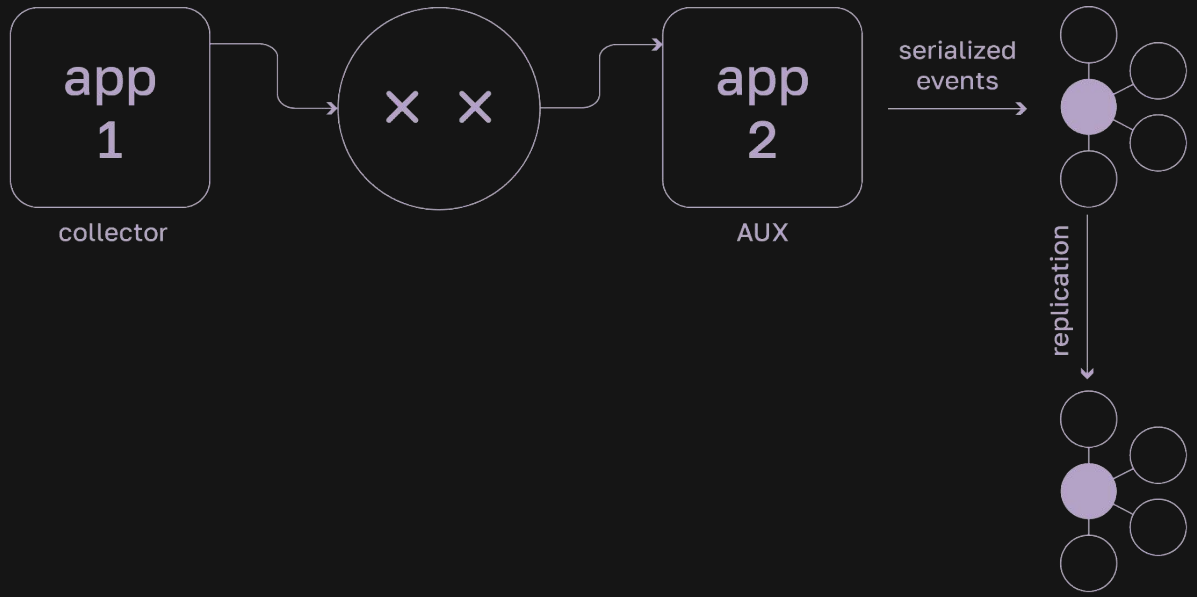
02. Локальные и глобальные (across DC) инсталляции.

03. Kafka-connect для связи с внешними системами.

04. «At least once» гарантии.

Data delivery

Визуализация



avito.tech

Processing

Москва — 2023

Processing

Realtime == stream processing?

- Samza
- Spark streaming
- Flink
- ksqlDB

Processing

ksqlDB pros & cons

01. Обработка данных в режиме реального времени.

Processing

ksqlDB pros & cons

01. Обработка данных в режиме реального времени.

02. SQL like синтаксис.

Processing

ksqlDB pros & cons

01. Обработка данных в режиме реального времени.

02. SQL like синтаксис.

03. Kafka-native.

Processing

ksqlDB pros & cons

01. Обработка данных в режиме реального времени.

02. SQL like синтаксис.

03. Kafka-native.

04. Много фич из коробки: материализованные представления, джойны, окна, etc.

Processing

ksqlDB pros & cons

01. Высокое потребление ресурсов и удовлетворительная производительность.

Processing

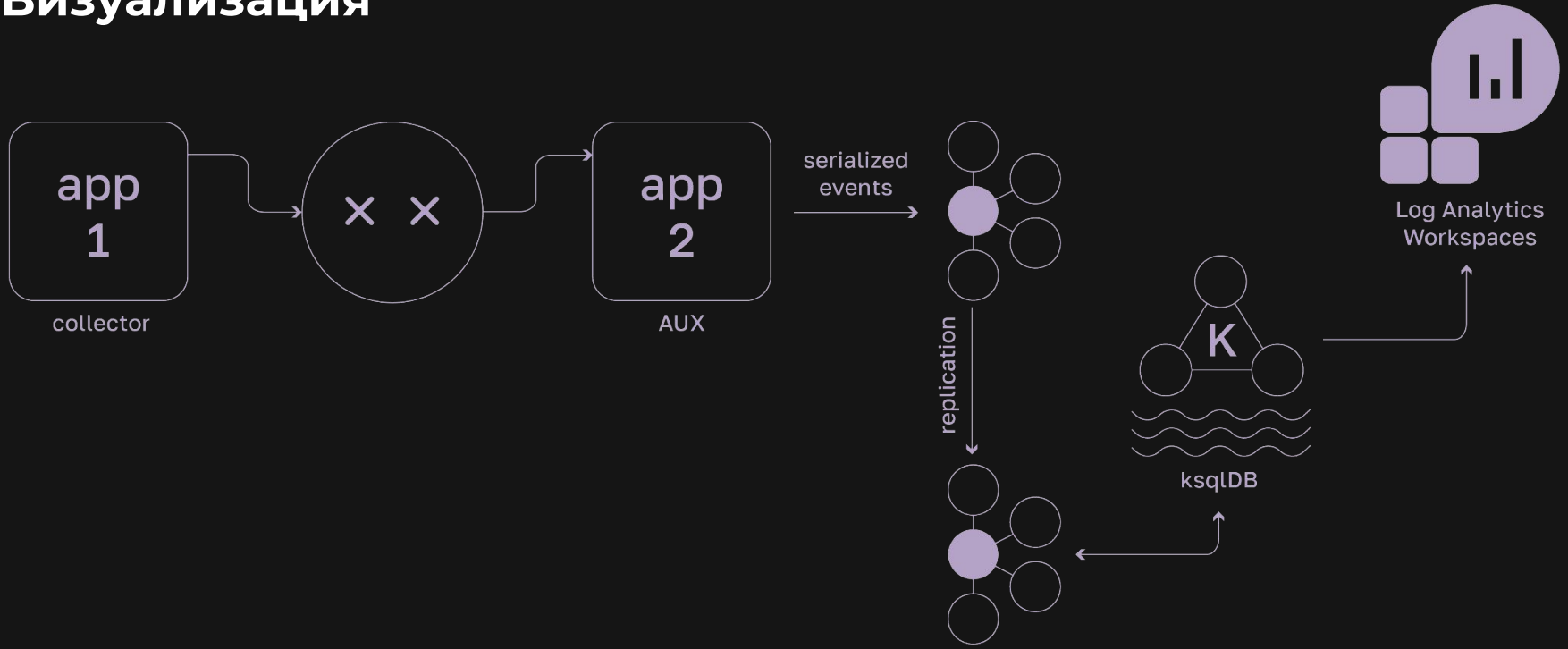
ksqlDB pros & cons

01. Высокое потребление ресурсов и удовлетворительная производительность.

02. Огромное количество ограничений со стороны ksqlDB и Kafka Streams.

Processing

Визуализация



avito.tech

Немного цифр...

Москва — 2023

1.6kk	19 Gb/s	20 ms
events per second	rx traffic	e2e latency

Detections

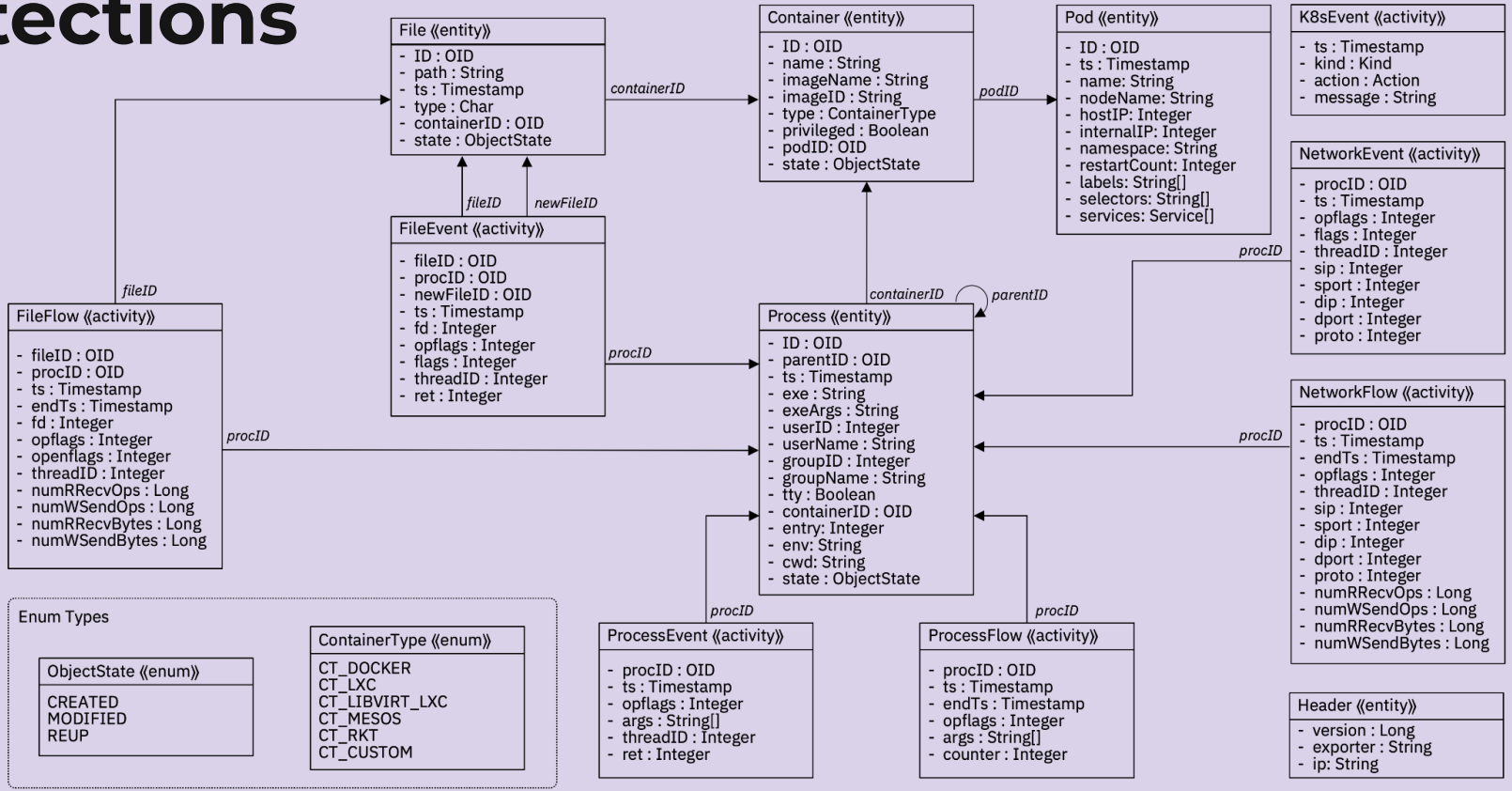
- MITRE — приоритезация / классификация

1. T1059 - Command and Scripting Interpreter
2. T1053 - Scheduled Task/Job
3. T1574 - Hijack Execution Flow
4. T1562 - Impair Defenses
5. T1543 - Create or Modify System Process
6. T1021 - Remote Services
7. T1003 - OS Credential Dumping
8. T1055 - Process Injection
9. T1548 - Abuse Elevation Control Mechanism

- Data set — host filter (sysflow collector)

```
FILTER="fd.filename = sudoers or evt.type in (execve, clone, exit, setuid)
 or fd.directory in (/etc/systemd/system, /usr/lib/systemd/system, /.config/systemd/user, /etc/systemd/user, /.local/share/systemd/user, /usr/lib/systemd/user)
 or (evt.type in (connect, accept) and fd.ip ≠ '127.0.0.1')"
```

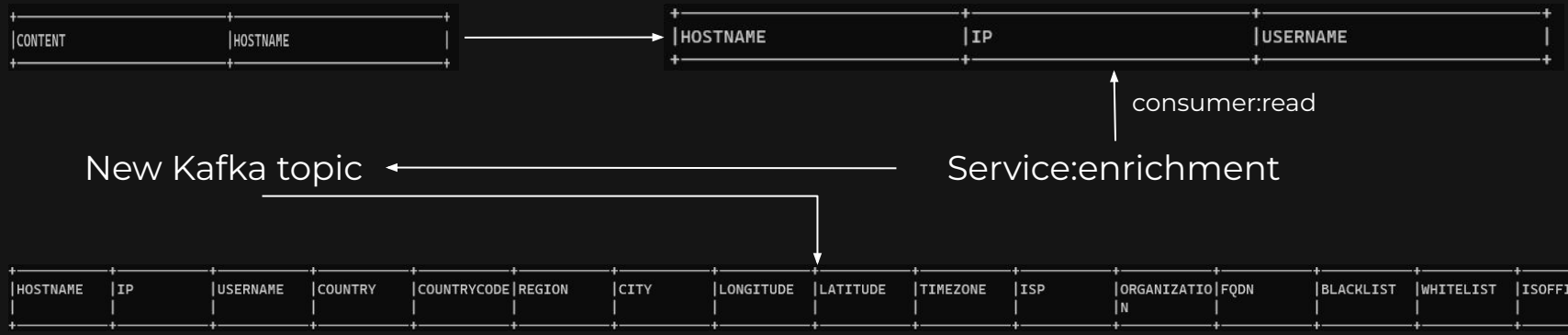
Detections


File ⟪entity⟫
- ID : OID
- path : String
- ts : Timestamp
- type : Char
- containerID : OID
- state : ObjectState
containerID
Container ⟪entity⟫
- ID : OID
- name : String
- imageName : String
- imageID : String
- type : ContainerType
- privileged : Boolean
- podID: OID
- state : ObjectState
podID
Pod ⟪entity⟫
- ID : OID
- ts : Timestamp
- name: String
- nodeName: String
- hostIP: Integer
- internalIP: Integer
- namespace: String
- restartCount: Integer
- labels: String[]
- selectors: String[]
- services: Service[]
K8sEvent ⟪activity⟫
- ts : Timestamp
- kind : Kind
- action : Action
- message : String
fileID
newFileID
FileEvent ⟪activity⟫
- fileID : OID
- procID : OID
- newFileID : OID
- ts : Timestamp
- fd : Integer
- opflags : Integer
- flags : Integer
- threadID : Integer
- ret : Integer
procID
NetworkEvent ⟪activity⟫
- procID : OID
- ts : Timestamp
- opflags : Integer
- flags : Integer
- threadID : Integer
- sip : Integer
- sport : Integer
- dip : Integer
- dport : Integer
- proto : Integer
fileID
FileFlow ⟪activity⟫
- fileID : OID
- procID : OID
- ts : Timestamp
- endTs : Timestamp
- fd : Integer
- opflags : Integer
- openflags : Integer
- threadID : Integer
- numRRecvOps : Long
- numWSendOps : Long
- numRRecvBytes : Long
- numWSendBytes : Long
containerID
parentID
Process ⟪entity⟫
- ID : OID
- parentID : OID
- ts : Timestamp
- exe : String
- exeArgs : String
- userID : Integer
- userName : String
- groupID : Integer
- groupName : String
- tty : Boolean
- containerID : OID
- entry: Integer
- env: String
- cwd: String
- state : ObjectState
procID
procID
procID
NetworkFlow ⟪activity⟫
- procID : OID
- ts : Timestamp
- endTs : Timestamp
- opflags : Integer
- threadID : Integer
- sip : Integer
- sport : Integer
- dip : Integer
- dport : Integer
- proto : Integer
- numRRecvOps : Long
- numWSendOps : Long
- numRRecvBytes : Long
- numWSendBytes : Long
Enum Types
ObjectState ⟪enum⟫
CREATED
MODIFIED
REUP
ContainerType ⟪enum⟫
CT_DOCKER
CT_LXC
CT_LIBVIRT_LXC
CT_MESOS
CT_RKT
CT_CUSTOM
ProcessEvent ⟪activity⟫
- procID : OID
- ts : Timestamp
- opflags : Integer
- args : String[]
- threadID : Integer
- ret : Integer
ProcessFlow ⟪activity⟫
- procID : OID
- ts : Timestamp
- endTs : Timestamp
- opflags : Integer
- args : String[]
- counter : Integer
Header ⟪entity⟫
- version : Long
- exporter : String
- ip: String

Detections

- Data set — ksqldb filtering / enrichment

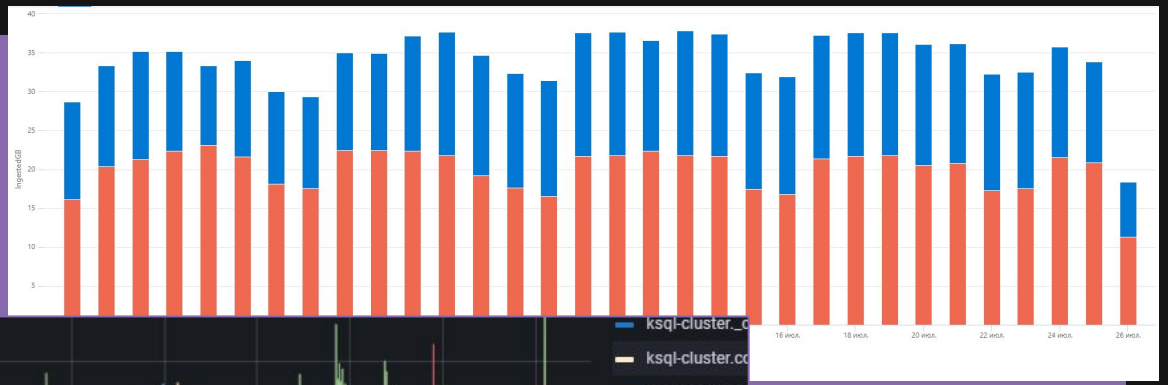
```
CREATE OR REPLACE STREAM OVPN_AUTH_S WITH (KAFKA_TOPIC='ksqldb-OVPN_AUTH_S', VALUE_FORMAT='JSON') AS SELECT
OVPN_S.HOSTNAME HOSTNAME,
REGEXP_EXTRACT('(?:MULTI: primary virtual IP for )((.*)\/)(([0-9]{1,3}\.){3}[0-9]{1,3})', OVPN_S.CONTENT, 3) IP,
REGEXP_EXTRACT('(?:MULTI: primary virtual IP for )((.*)\/)(([0-9]{1,3}\.){3}[0-9]{1,3})', OVPN_S.CONTENT, 2) USERNAME
FROM OVPN_S OVPN_S
WHERE (OVPN_S.HOSTNAME not like '***-%' AND OVPN_S.HOSTNAME not like '***-s%' AND OVPN_S.CONTENT LIKE '%MULTI: primary virtual IP for%')
EMIT CHANGES;
```

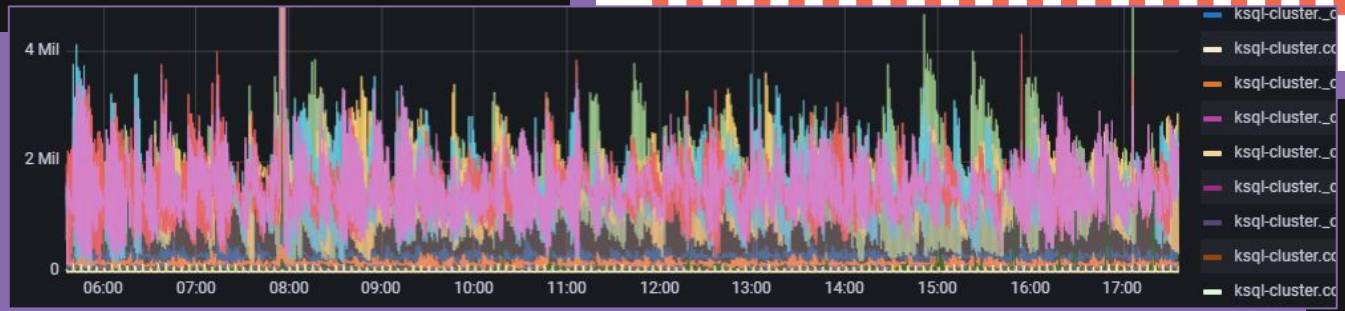


Detections

- Operational log / Cold storage
- Metrics
  1,6кк/s > 514k/s Cloud
  4k+ infra vs 100+hosts

Azure Log Analytics - Usage





Links

- Understanding Linux Audit
- SysFlow: Scalable System Telemetry for Improved Security Analytics
- Kafka streams vs ksqlDB

Спасибо за внимание

Иван - https://t.me/deadbeaf
Тимур - https://t.me/Kelcelenelelvial

feedback slide