APACHE KAFKA, CYBERSECURITY, DIGITAL FORENSICS, FORENSICS, MACHINE LEARNING, SECURITY, SIEM, SITUATIONAL AWARENESS, THREAT DETECTION, THREAT INTELLIGENCE, TIERED STORAGE

Kafka for Cybersecurity (Part 4 of 6) – Digital Forensics

9 minute read

This blog series explores use cases and architectures for Apache Kafka in the cybersecurity space, including situational awareness, threat intelligence, forensics, air-gapped and zero trust environments, and SIEM / SOAR modernization. This post is part four: Digital Forensics.

By KAI WAEHNER · 23. July 2021

Apache Kafka became the de facto standard for processing data in motion across enterprises and industries. Cybersecurity is a key success factor across all use cases. **Kafka** is not just used as a backbone and source of truth for data. It also **monitors, correlates, and proactively acts on events from various real-time and batch data sources to detect anomalies and respond to incidents**. This blog series explores use cases and architectures for Kafka in the cybersecurity space, including situational awareness, threat intelligence, forensics, air-gapped and zero trust environments, and SIEM / SOAR modernization. **This post is part four: Digital Forensics**.

Blog series: Apache Kafka for Cybersecurity

This blog series explores why security features such as RBAC, encryption, and audit logs are only the foundation of a secure event streaming infrastructure. Learn about use cases, architectures,

and reference deployments for Kafka in the cybersecurity space:

- Part 1: Data in Motion as cybersecurity backbone
- Part 2: Situational awareness
- Part 3: Threat intelligence
- **Part 4 (THIS POST): Forensics**
- Part 5: Air-gapped and zero trust environments
- Part 6: SIEM / SOAR modernization

Subscribe to my newsletter to get updates immediately after the publication. Besides, I will also update the above list with direct links to this blog series's posts as soon as published.

Digital Forensics

Let's start with the definition of the term "Digital Forensics". In the IT world, we can define it as **analytics of historical data sets to find insights**. More specifically, digital forensics means:

- Application of science to criminal and civil laws, mainly during a **criminal investigation**.
- **I**t is applied to **internal corporate investigations in the private sector** or, more generally, **to intrusion investigations in the public and private sector** (a specialist probe into the nature and extent of an unauthorized network intrusion).
- Forensic scientists collect, preserve, and analyze scientific evidence during the course of investigating **digital media** in a forensically sound manner.
- **Identify, preserve, recover, analyze and present facts** and opinions about digital information.

The technical aspect is divided into several sub-branches relating to the type of digital devices involved: Computer forensics, network forensics, forensic data analysis, and mobile device forensics.

A **digital forensic investigation commonly consists of three stages: acquisition, analysis, and reporting. The final goal is to reconstruct digital events.** Let's see what role Kafka and its ecosystem play here.

Digital Forensics with Kafka's Long Term Storage and Replayability

Kafka stores data in its distributed commit log. The log is durable and persists events on the disk with guaranteed order. The replication mechanism guarantees no data loss even if a node goes down. Exactly-once semantics (EOS) and other features enable transactional workloads. Hence, more and more deployments leverage Kafka as a database for long-term storage.

Forensics on Historical Events in the Kafka Log

The ordered historical events enable Kafka consumers to do digital forensics:

- Capture the complete attack vector
- Playback of an attack for the training of humans or machines
- Create threat surface simulations
- Compliance / regulatory processing
- Etc.

The **forensics consumption is typically a batch process to consume all events from a specific timeframe**. As all consumers are truly decoupled from each other, the "normal processing" can still happen in real-time. There is no performance impact due to the concepts of Kafka's decoupling to enable a domain-driven design (DDD). The forensics teams use different tools to connect to Kafka. For instance, data scientists usually use the Kafka Python client to consume historical data.

Challenges with Long-Term Storage in Kafka

Storing data long-term in Kafka is possible since the beginning. **Each Kafka topic gets a retention time**. Many use cases use a retention time of a few hours or days as the data is only processed and stored in another system (like a database or data warehouse). However, more and more projects use a **retention time of a few years or even -1 (= forever)** for some Kafka topics (e.g., due to compliance reasons or to store transactional data).

The **drawback** of using Kafka for forensics is the huge volume of historical data and its related **high cost and scalability issues**. This gets pretty expensive as Kafka uses regular HDDs or SDDS as the disk storage. Additionally, data rebalancing between brokers (e.g., if a new broker is added to a cluster) takes a long time for huge volumes of data sets. Hence, rebalancing takes hours can impact scalability and reliability.

But there is a solution to these challenges: Tiered Storage.

Tiered Storage for Apache Kafka via KIP-405

Tiered Storage for Kafka separates compute and storage. This solves both problems described above:

- Significant **cost reduction** by using a much cheaper storage system.
- Much **better scalability and elasticity** as rebalancing is only needed for the brokers (that only store the small hot data sets)

KIP-405 **is the assigned open-source task that describes the plan and process for adding Tiered Storage to Apache Kafka**. Confluent is actively working on this with the open-source community. Uber is leading the initiative for this KIP and works on HDFS integration. Check out Uber's Kafka Summit APAC talk about Tiered Storage for more details.

Confluent Tiered Storage for Kafka

Confluent Tiered Storage is **generally available for quite some time in Confluent Platform and**

used under the hood in Confluent Cloud in thousands of Kafka clusters. Certified object stores include cloud object stores such as AWS S3 or Google Cloud Storage and on-premise object storage such as Pure Storage FlashBlade.

The **architecture of Confluent Tiered Storage** looks like this:

Benefits of Confluent Tiered Storage for Kafka include:

- **Store data forever in a cost-efficient way** using your favorite object storage (cloud and on-premise)
- **The separation between computing and storage** (hot storage attached to the brokers and cold storage via the cheap object store)
- **Easy scale up/down** as only the hot storage requires rebalancing – most deployments only store the last few hours in hot storage
- **No breaking code changes in Kafka clients** as it is the same regular Kafka API as before
- **Battle-tested** in Confluent Cloud in thousands of Kafka clusters
- **No impact on performance for real-time consumers** as these consume from page cache/memory anyway, not from the hot or cold storage

As you can see, Tiered Storage is a huge benefit to provide long-term storage for massive volumes of data. This allows rethinking your data lake strategy.

True Decoupling for SIEM, SOAR, and other Kafka Consumers

Kafka's Distributed Commit Log **captures the running history of signals**. This

- enables **true decoupling** and domain-driven design
- absorbs velocity and volume to **protect and stabilize slow consumers**
- allows **organic truncation** via the right retention time per Kafka topic

Various producers continuously ingest new events into Kafka without knowing or caring about slow consumers. **Kafka handles the backpressure**. Different consumer applications use their own capable speed and communication paradigm for data ingestion:

Affordability at Scale for Real-Time and Replay

Most **digital forensics and cybersecurity solutions use some batch style** for analytics. This is **expensive from cost but often even more important performance perspective**.

A Kafka-native streaming application can process data much faster for near real-time analytics. For instance, a **simple KSQL window query is used to sift through the data by playing back the topic**. This will be **much faster than stuffing into an index** of a SIEM and then running a query.

Having tiered and then creating a **filtered and aggregated "forensic" topic means the flight data recorder is always there**. Many forensics use cases don't have to needless stash data in a batch-based sink such as Spark or a SIEM.

In summary, a **Kafka cluster powered by Tiered Storage is an affordable solution for both real-time analytics and digital forensics at scale**.

Distributed Digital Forensics at Scale with Kafka and Spark

The paper "**Digital Forensics Compute Cluster** (DFORC2): A High Speed Distributed Computing Capability for Digital Forensics" is a great example. The infrastructure processes big data at scale for forensics use cases.

The foundation of the project is the **digital forensics platform Autopsy.** Autopsy is computer software that makes it simpler to deploy many of the open-source programs and plugins used in The Sleuth Kit. The **graphical user interface displays the results from the forensic search of the underlying volume**. Using the GUI makes it easier for investigators to flag pertinent sections of data.

The team **extended Autopsy with a Kafka and Spark integration to add distributed compute power for data processing:**

The paper was published in 2017. **Today, several distributed workloads could also leverage only the Kafka ecosystem including Kafka Streams and ksqlDB**. The advantages would be the single infrastructure instead of two separate distributed clusters and providing better real-time capabilities. Having said that, some big data batch workloads like map-reduce or shuffling are still a great fit for Spark jobs.

In the case of Autopsy, a tool to analyze and process large volumes of files in batch, Spark is a good choice. But similarly, a lot of file processing can be done in a stream processing application; as long as the files are processed independently and not shuffled together.

The Role of AI and Machine Learning in Digital Forensics

Digital Forensics is all about collecting, analyzing, and acting on historical events. **SIEM / SOAR and other cybersecurity applications** are great for many use cases. However, they **are often not real-time and do not cover all scenarios**. In an ideal world, you can act in real-time or even in a predictive way to prevent threats.

In the meantime, Kafka plays a huge role in AI / Machine Learning / Deep Learning infrastructures. A good primer to this topic is the post "Machine Learning and Real-Time Analytics in Apache Kafka Applications". To be clear: **Kafka and Machine Learning** are different concepts and technologies. However, they **are complementary and a great combination to build scalable real-time infrastructures for predicting attacks and other cyber-related activities**.

The following sections show how machine learning and Kafka can be combined for model scoring and/or model training in forensics use cases.

Model Deployment with ksqlDB and TensorFlow

Analytics models enable predictions in real-time if they are deployed to a real-time scoring application. **Kafka natively supports embedding models for real-time predictions at scale**:

This example uses a trained TensorFlow model. A ksqlDB UDF embeds the model. Of course, **Kafka can be combined with any AI technology**. An analytic model is just a binary. No matter if you train it with an open-source framework, a cloud service, or a proprietary analytics suite.

Another option is to leverage a streaming model server to connect a deployed model to another streaming application via the Kafka protocol. Various model servers already provide a Kafka-native interface in addition to RPC interfaces such as HTTP or gRPC.

Kafka-native Model Training with TensorFlow I/O

Embedding a model into a Kafka application for low latency scoring and decoupling is an obvious approach. However, in the meantime, more and more companies also **train models via direct consumption from the Kafka log**:

Many AI products provide a native Kafka interface. For instance, TensorFlow I/O offers a Kafka plugin. There is **no need for another data lake just for model training!** The model training itself is still a batch job in most cases. That's the beauty of Kafka: **The heart is real-time, durable, and scalable. But the consumer can be anything: Real-time, near real-time, batch, request-response.** Kafka truly decouples all consumers and all producers from each other.

We have built a demo project on Github that shows the native integration between Kafka and TensorFlow for model training and model scoring.

Kafka and Tiered Storage as Backbone for Forensics

Digital Forensics **collects and analyzes historical digital information to find and present facts about criminal actions**. The insights help to reconstruct digital events, find the threat actors, and build better situational awareness and threat detection in the future. This post showed what role Apache Kafka and its ecosystem play in digital forensics.

Often, Kafka is the **integration pipeline that handles the backpressure for slow consumers such as SIEM / SOAR products**. Additionally, the **concept of Tiered Storage for Kafka enables long-term storage and digital forensics use cases**. This can include Kafka-native model training. All these use cases are possible parallel to any unrelated real-time analytics workloads as Kafka truly decouples all producers and consumers from each other.

Do you use Kafka for forensics or any other long-term storage use cases? Does the architecture leverage Tiered Storage for Kafka? Let's connect on LinkedIn and discuss it! Stay informed about new blog posts by subscribing to my newsletter.

Last name

Email Address *

SIGN UP

We don't spam! Read more in our privacy policy

Kai Waehner

builds cloud-native event streaming infrastructures for real-time data processing and analytics

VIEW COMMENTS (1)

YOU MAY ALSO LIKE

Apache Kafka + MQTT = End-to-End IoT Integration (Code, Slides, Video)

APACHE KAFKA, BIG DATA, CONFLUENT, EAI, INTERNET OF THINGS, KAFKA CONNECT, MESSAGING, MICROSERVICES, MQTT, OPEN SOURCE, STREAM PROCESSING

By KAI WAEHNER · 10. September 2018

MQTT and Apache Kafka are a perfect combination for end-to-end IoT integration from edge to data center. This post discusses two different approaches and refers to implementations on Github using Apache Kafka, Kafka Connect, Confluent MQTT Proxy and Mosquitto.

READ MORE >

Can Apache Kafka Replace a Database? – The 2020 Update

ANALYTICS, APACHE KAFKA, ARCHITECTURE, BIG DATA, CONFLUENT, DATABASE, INTEGRATION, KAFKA CONNECT

By KAI WAEHNER · 12. March 2020

Can and should Apache Kafka replace a database? How long can and should I store data in Kafka?...

READ MORE >

1

TECHNOLOGY EVANGELIST

SUBSCRIBE TO MY NEWSLETTER

Kai Waehner

builds cloud-native event streaming infrastructures for real-time data processing and analytics

END-TO-END INTEGRATION

FEATURED POSTS

1. Apache Kafka, KSQL and Apache PLC4X for IIoT Data Integration and Processing
2. Apache Kafka vs. Middleware (MQ, ETL, ESB) – Slides + Video
3. Deep Learning Example: Apache Kafka + Python + Keras + TensorFlow + Deeplearning4j

Stay informed about new blog posts!

Email Address *

First name

Last name

SIGN UP

We don't spam! Read our privacy policy for more info.

CATEGORIES

Select Category

TAG – CLOUD

ANALYTICS APACHE APACHE CAMEL APACHE KAFKA AWS BIG DATA BUSINESSWORKS CLOUD CLOUD-NATIVE CONFLUENT DEEP LEARNING DOCKER EAI EDGE ENTERPRISE APPLICATION INTEGRATION ENTERPRISE SERVICE BUS ESB EVENT STREAMING HADOOP HYBRID IBM IIOT INTEGRATION IOT J2EE JAVA JAVA ENTERPRISE EDITION JEE KAFKA KAFKA CONNECT KAFKA STREAMS KSQL KUBERNETES MACHINE LEARNING

MICROSERVICES MIDDLEWARE

OPEN SOURCE ORACLE

REAL TIME SOA STREAMBASE

STREAMING ANALYTICS

STREAM PROCESSING TALEND

TIBCO

5G, APACHE KAFKA, BSS, CLOUD-NATIVE, EDGE, OSS, OTT (OVER THE TOP), TELCO, TELECOM, USE CASES, V2X

Cloud-Native 5G, MEC and OSS/BSS/OTT Telco with Apache Kafka and Kubernetes

KAI WAEHNER · 6. September 2021

5G, AIR GAPPED, APACHE KAFKA, EDGE, ENERGY, HYBRID CLOUD, MINING, OIL AND GAS

Panel Discussion about Kafka, Edge, Networking and 5G in Oil and Gas and Mining Industry

KAI WAEHNER · 20. August 2021

APACHE KAFKA, CONFLUENT, ELASTICSEARCH, IBM, KAFKA CONNECT, SIEM, SOAR, SPLUNK

Kafka for Cybersecurity (Part 6 of 6) – SIEM / SOAR Modernization

KAI WAEHNER · 9. August 2021

AIR GAPPED, APACHE KAFKA, CYBERSECURITY, EDGE, RANSOMWARE, SECURITY, SITUATIONAL AWARENESS, THREAT DETECTION, ZERO TRUST

Kafka for Cybersecurity (Part 5 of 6) – Zero Trust and Air-Gapped Environments

KAI WAEHNER · 2. August 2021