AI-Driven Data Onboarding

17.03.2026 | Frankfurt

DB Systel

Inhaltsverzeichnis.

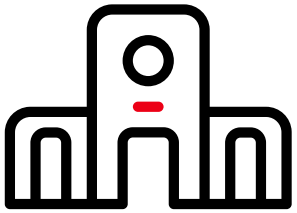
Wer sind wir?

- Wir sind ein agiles DevOps-Team in der DB Systel GmbH, die eine 100 %-ige Tochtergesellschaft der Deutsche Bahn AG ist.
- Wir betreiben eine zentrale Splunk-Umgebung für den gesamten Konzern und stellen diese als Plattform für Geschäftsfelder, Unternehmen und Teams mandantengetrennt zur Verfügung.
- Unsere stetig wachsende Umgebung umfasst aktuell 60 Indexer in einem MultiSite-Cluster und mehrere SearchHead Cluster.
- Wie bedienen Operation und Security UseCases.

Deutsche Bahn

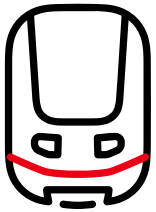
Schienennetz

ca. 33.000km Strecke
ca. 64.000 Weichen
ca. 3.800 Signalanlagen



Bahnhöfe

ca. 5.400 Bahnhöfe



Personenverkehr

ca. 410 ICEs
ca. 4.000 Triebzüge
ca. 6.000 Busse

Güterverkehr

ca. 2.500 Lokomotiven
ca. 65.000 Güterwagen

Quelle: DB Konzern Geschäftsbericht 2024

Herausforderungen

Herausforderungen ans Logging

Ein großer Konzern. **5 Geschäftsfelder** mit über **250 Unternehmen** in sehr vielen **Standorten** und mit sehr vielen **Fahrzeugen**.

Wir wollen **Security** und **Operations** abdecken, auf **System-** und **Anwendungsebene**.

Viele Systeme. Über **3.000 Anwendungen** im Konzern. **On-prem**, Private **Clouds, Software-as-a-Service**-Lösungen und **Anlagen**.

Kann uns Splunk hier helfen?

Datenspeicherung

- Index Cluster
- Multi Site
- Skaliert horizontal

Analyse und Zugriffe

- SearchHead Cluster
- Multiple parallele Installationen
- Skaliert horizontal

Datenerfassung

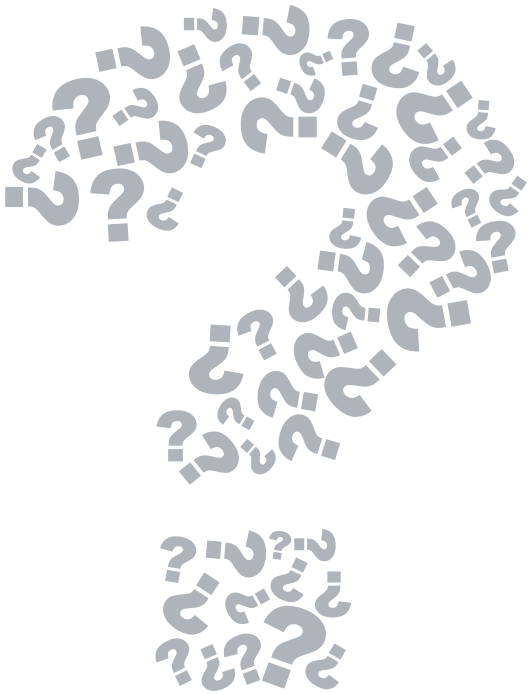
- Splunk Universal Forwarder für die gängigen OS-Systeme
- Zentrale Konfiguration
- Skaliert horizontal

Datenarten

- Auslesen von Dateien
- HTTP Event Collector
- openTelemetry OTEL
- ...

Also einschalten, aktivieren und fertig?

Diversität der Daten



OS-Log: Windows, Linux, Unix, Netzwerk. Relativ standardisiert

openTelemetry: standardisiert

SaaS: Wir sind nicht die Ersten, hier existieren bereits Lösungen

Eingekaufte Anwendungen: für die bekanntesten existieren Lösungen, aber für viele nicht

DB eigenentwickelte Anwendungen: durch die Größe des Konzerns, die Historie keine Logging-Standardisierung und fast alles individuell

Was bedeutet das für uns?

Viele Stakeholder

Zuordnung CIM erfordert mehr Know-how als nur Splunkwissen

Anwendungsverantwortlicher ist in der Regel nicht der Betreiber

Wie viele Anwendungen waren das noch gleich?

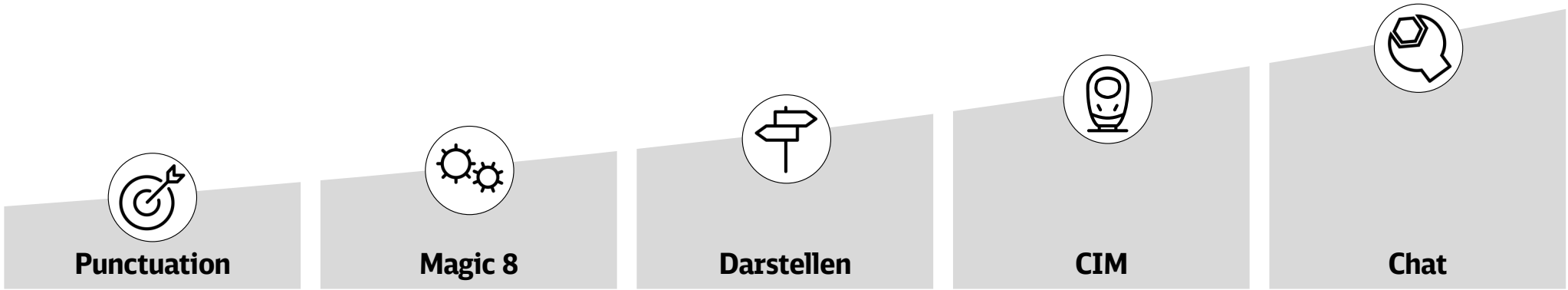
Hohe Aufwände – alles individuelle File Strukturen

Betreiber haben in der Regel kein Splunkwissen

Analyse und Onboarding der Logfiles

Idee

Unsere Reise in *5* Schritten



Punctuation

- Erster Ansatz war zu erkennen, ob das Format schon bekannt ist. (Punct/Sourcetype)
- Doppelte Configs sollten verhindert werden.

Magic 8

- Danach kam die Idee GPT bzgl. der Magic 8 / Great 8 für die props.conf zu fragen.
- Das lieferte erstaunlich gute Ergebnisse, trotz recht einfachem Prompt.

Darstellen

- Ergebnisse sollten nicht nur in einer Konfiguration erscheinen, wir wollten diese auch direkt anhand unseres Logsamples testen und darstellen.

CIM

- Wenn Magic 8 schon funktioniert, warum sollte dann nicht auch CIM-Zuordnung und Feldextraktion funktionieren.

Chat

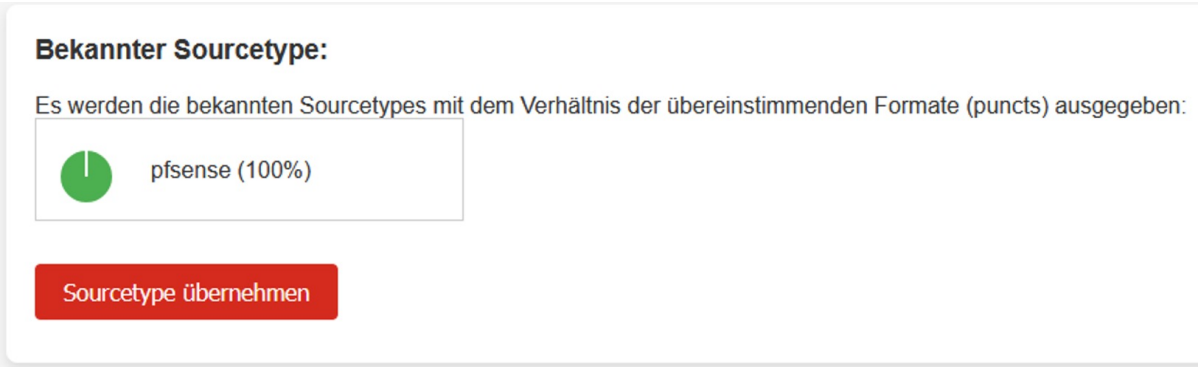
- Und letztendlich ist man, wenn man dann länger mit GPT arbeitet, Chats gewohnt.
- Also warum nicht mit der KI für besser Ergebnisse chatten.

Demo

Erkennung: Existiert das Format schon?

- **Doppelte Arbeit vermeiden & Wissen heben.**

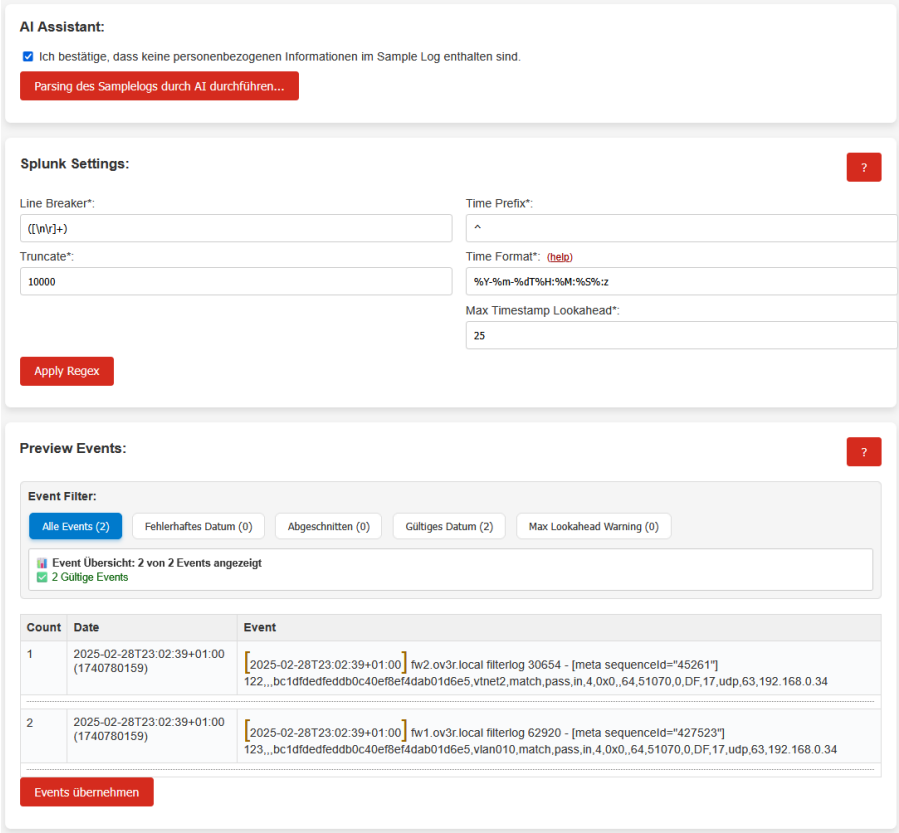
- Vergleich der hochgeladenen Logs über Punctuation-/Struktur-Signaturen mit bereits indexierten Mustern.
- Output-Lookup (KV) hält: PUNCT, Sourcetype, zugehörige props-Konfiguration.
- Aktuelle Limitation: Multiline-Events nur teilweise erkannt (zeilenorientierter Ansatz).
- Ausblick: Hybrid Matching (Embedding + Struktur + PUNCT) für höhere Trefferquote & Multiline-Stabilität.
- Nutzen: Sofortige Transparenz – Anwender (auch ohne weitreichende Splunk-Rechte) sehen: Quelle schon vorhanden? Wiederverwendbar?



Event Breaking & Zeitstempel-Validierung

- **Saubere Basis vor jeder weiteren Investition.**

- Live-Vorschau nach Upload: korrekt gebrochene Events, Timestamp-Validität, Lücken/Anomalien.
- KI-gestützte Vorschläge der Great8 reduziert auf: LINE_BREAKER, TIME_PREFIX, TIME_FORMAT, MAX_TIMESTAMP_LOOKAHEAD.
- Iterative Verfeinerung via Chat statt Try-&-Error in Config-Files.
- Nutzen: Frühe Qualitätssicherung → Fehlersuche später entfällt, Onboarding-Zeit sinkt deutlich.



Feldextraktion & Normalisierung

- **Schnell zu korrelationsfähigen Daten.**

- KI erkennt Kandidaten-Felder, zeigt Coverage / Lücken transparent.
- Vorläufiges CIM-Mapping (Standardisierung & spätere Korrelation beschleunigt).
- Erklärbare Regex-Generierung statt individueller Skriptfragmente.
- Nutzen: Weniger manuelle Schleifen, konsistente Ergebnisse, Wissensaufbau strukturiert.

Transforms.conf Definition

```
[extract_firewall_logs]
REGEX = (?<timestamp>\d{4}-\d{2}-\d{2}T\d{2}:\d{2}:\d{2}[+-]\d{2}:\d{2})\s+(?<hostname>[\w\.]+)\s+(?<log_type>\w+)\s+(?<log_id>\d+)\s+-\s+\[meta\s+sequenceId="(?<sequence_id>\d+)"\]\s+(?<rule>\d+),,,(?<tracker_id>[a-f0-9]+),(?<interface>[\w\d]+),(?<reason>\w+),(?<action>\w+),(?<direction>\w+),(?<ip_version>\d),(?<tos>[^,]+),(?<ecn>[^,]*),\s+(?<ttl>\d+),(?<id>\d+),(?<offset>\d+),(?<flags>\w+),(?<protocol_id>\d+),(?<protocol>\w+),(?<length>\d+),(?<src_ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}),(?<dst_ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}),(?<src_port>\d+),(?<dst_port>\d+),(?<payload_length>\d+)
```

Erkannte Felder

Feldname	Gefunden in (%)	Top 5 Werte	CIM Mapping
timestamp	100%	• 2025-02-20T17:54:03+01:00 (18) • 2026-02-20T17:54:03+01:00 (2)	timestamp
hostname	100%	• fw1.ov3r.local (20)	host
log_type	100%	• filterlog (20)	log_type
log_id	100%	• 34049 (20)	log_id
sequence_id	100%	• 12711029 (18) • 12711028 (2)	sequence_id
rule	100%	• 83 (18) • 143 (2)	rule
tracker_id	100%	• fae559338f65e11c53669fc3642c93c2 (18) • 1b5a7e5825cd7208895e6ab9a1260a71 (2)	tracker_id
interface	100%	• vlan010 (18) • vlan030 (2)	interface
reason	100%	• match (20)	reason
action	100%	• pass (20)	action
direction	100%	• out (18) • in (2)	direction
ip_version	100%	• 4 (20)	ip_version
tos	100%	• 0x0 (20)	tos
ecn	0%	Keine Werte gefunden	-
ttl	100%	• 127 (18) • 128 (2)	ttl
id	100%	• 51561 (20)	id
offset	100%	• 0 (20)	offset
flags	100%	• none (20)	flags
protocol_id	100%	• 17 (20)	protocol_id
protocol	100%	• udp (20)	protocol
length	100%	• 60 (20)	length
src_ip	100%	• 192.168.3.201 (20)	src_ip
dst_ip	100%	• 192.168.0.200 (20)	dst_ip
src_port	100%	• 65089 (20)	src_port
dst_port	100%	• 53 (20)	dst_port
payload_length	100%	• 40 (20)	payload_length

Events mit markierten Feldern

Zeige Events 1 bis 10 von 194

Event 1

2466," title="offset">025-02-20T17:54:03+01:00 fw1.ov3r.local filterlog 34049 - [meta sequenceId="12711029"]
83,,,fae559338f65e11c53669fc3642c93c2,vlan010,match,pass,out,4,0x0,, 127,51561,0,none,17,udp,60,192.168.3.201,192.168.0.200,65089,53,40

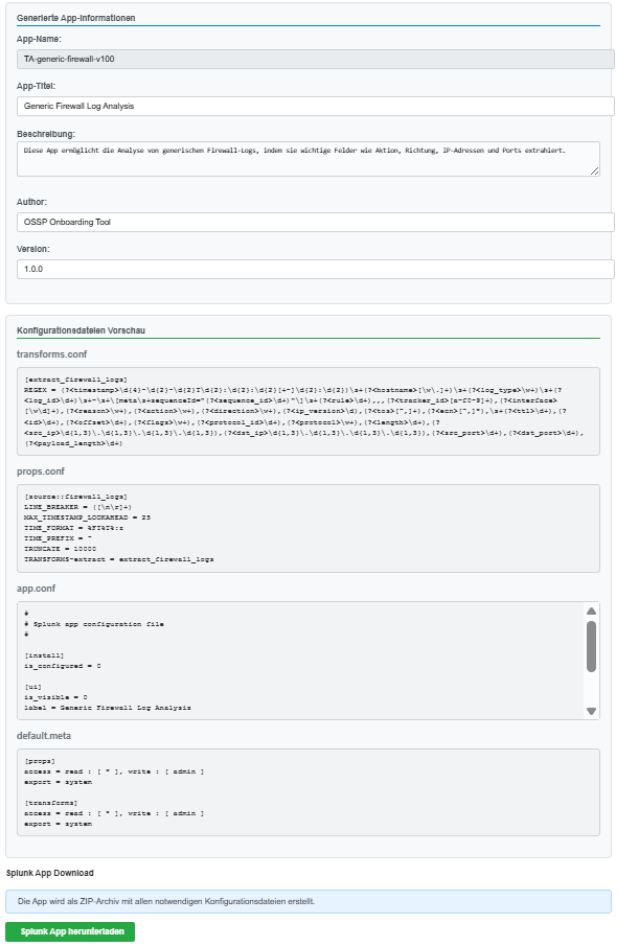
Event 2

2466," title="offset">026-02-20T17:54:03+01:00 fw1.ov3r.local filterlog 34049 - [meta sequenceId="12711028"]
143,,,1b5a7e5825cd7208895e6ab9a1260a71,vlan030,match,pass,in,4,0x0,, 128,51561,0,none,17,udp,60,192.168.3.201,192.168.0.200,65089,53,40

App Enablement & Release-Fähigkeit

- **Vom Roh-Log zur produktionsnahen TA in Sekunden.**

- Automatische Erstellung einer Splunk TA (props.conf, transforms.conf, app.conf, metadata, README).
- Namens- & Versionierungslogik gemäß Konvention → direkt CI/CD-fähig.
- Sofort einsetzbar für Test, Security & Compliance-Gates.
- Nutzen: Time-to-Value stark reduziert, standardisierte Basis für Deployment & Governance.



Vision

Automatisiertes Datenonboarding ohne Splunk Know-how. Volle Transparenz inklusive Selbstheilung.

Copyright: Deutsche Bahn AG / v.l. Dominic Dupont / Skydeck / Gorodenkoff Productions

Anwendungsverantwortliche bestellen für ihre Anwendungen ein Data-Onboarding.

Die **zuständigen Betriebsteams** werden automatisch ermittelt und informiert.

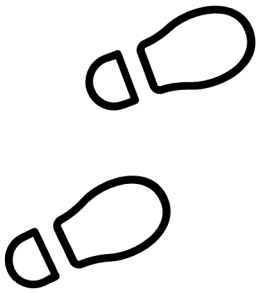
Nach Angabe der notwendigen Logfiles, werden diese ohne Zeitverzögerung und ohne Splunk-Know-how **analysiert** und **konfiguriert**.

Neue Sourcetypes fließen in ein **eigenes ML-Model** ein, um **Veränderungen** zukünftig zu **bemerken** und dem zuständigen Betriebsteam **avisieren** zu können.

Eine erneute **Analyse** mit anschließender **Rekonfiguration** kann so ganz leicht wieder angestoßen werden.

Jederzeit können **alle** Beteiligten sehen, welche **Datenquellen** zu welcher **ihrer Anwendungen** gehören, wie der **Status** des Onboardings ist und in welcher **Qualität** die Events ins Splunk-System laufen.

Next steps



- **GIT-Anbindung:** Automatisches Deployment über Pipelines mit Vier-Augen-Checks.
- **Umgang mit Strukturierte Daten verbessern:** Strukturierte Daten (XML/CSV) sind aktuell noch schwierig.
- **Nutzung einer lokalen KI/LLM:** Datenschutz ist immer ein großes Thema,
- **Nutzung von RAG/MCP:** Testen, ob wir bessere Ergebnisse zu bereits existierenden Sourcetypes bekommen. Punctuation hat seine Grenzen.
- **Bahninterne Informationen:** Event-Tagging und Konfigerweiterungen mit Anwendungs-IDs, IP-Netze, Zugnummern, etc.

Vielen Dank


DB