
29th ANNUAL FIRST CONFERENCE
SAN JUAN PUERTO RICO
JUNE 11-16, 2017
FIGHTING PIRATES AND PRIVATEERS
WWW.FIRST.ORG

Marvin – Automated Incident Handling at DFN-CERT

Jan Kohlrausch, Eugene Brin, Christian Keil

Introduction DFN-CERT

- Started 1993 as a research project at the University of Hamburg
- Computer Emergency Response Team for the German research network (DFN)
- Activities:
  - CERT services: Incident handling and coordination, vulnerability management, protection of the DFN network infrastructure (DDoS)
  - Consulting, training, and risk management
  - Participation in research projects
  - Provision of PKI services

Introduction DFN

- DFN: German Research network
  - Serves all German universities and research institutions
  - Operates global infrastructure: X-WiN
  - Comprises 27 ASN (largest is AS680)
  - Very large number of constituent sites and networks

Incident handling at DFN-CERT

- Before 2000: Compromised UNIX servers and workstations for fun; e.g. portscans, IRC-bots
  - **Manual reaction to incidents**
- After 2001: Appearance of Internet worms and MS Windows incidents
  - **Significant increase in incident numbers**
- Current situation: Large number of security events
  - **Manual incident handling is unfeasible**

Incident handling: (R)Evolution

1) Shell and PERL scripts to handle security events

   Register incidents, acquire contact data, compile report

2) Import of data into SQL database

   Start of security events categorization

3) Starting to send incident reports (monolithic `PERL` script)

   Manual maintenance of contact information and administration

4) Development of Marvin

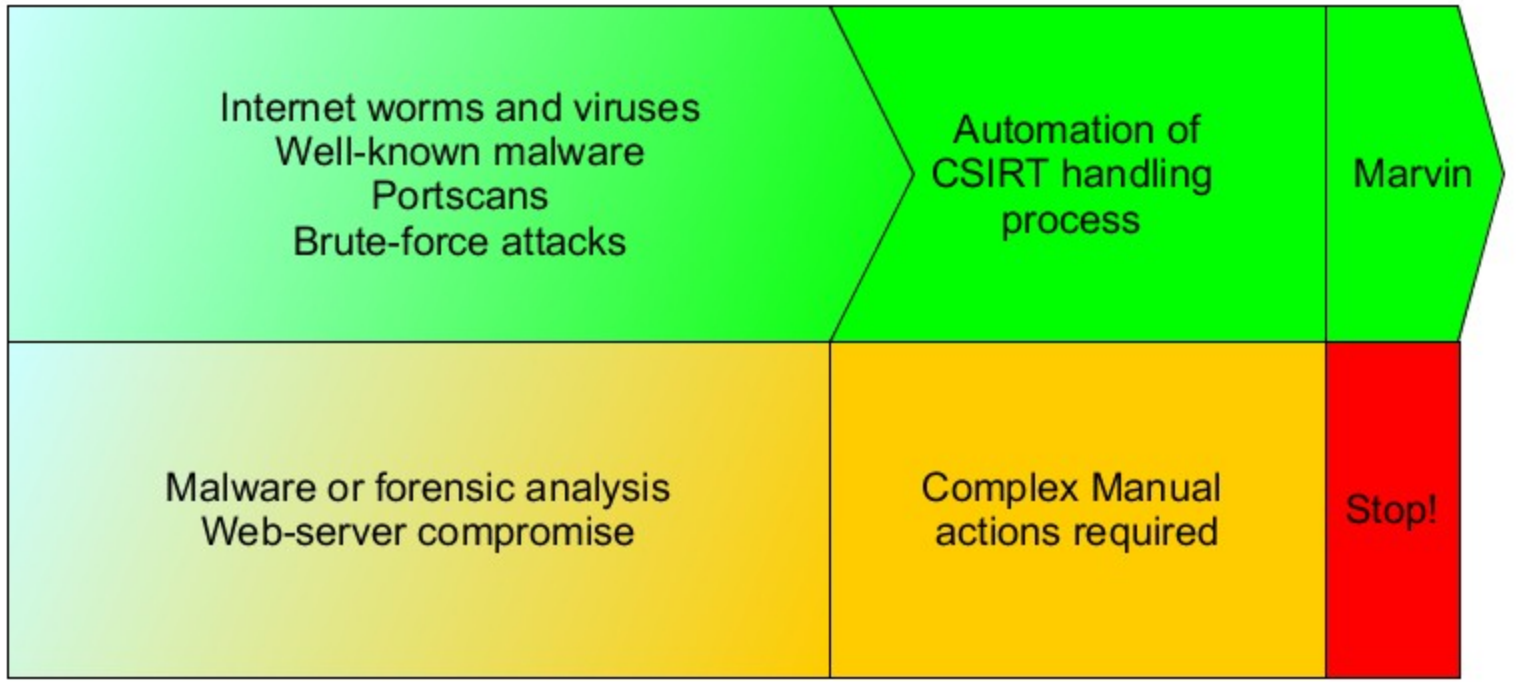
Early experiences with automated IH

- Security skills at universities vary from part-time administrator to security researcher (e.g. malware analysis)
- It is important to support administrators with limited experience
  - Provide as much as possible helpful information regarding to the threat: what happened, how to detect, and how to resolve
  - Avoid imprecise reports
  - Send report to appropriate contact; e.g. not an unspecific group accounts
  - **Actively involve constituency**

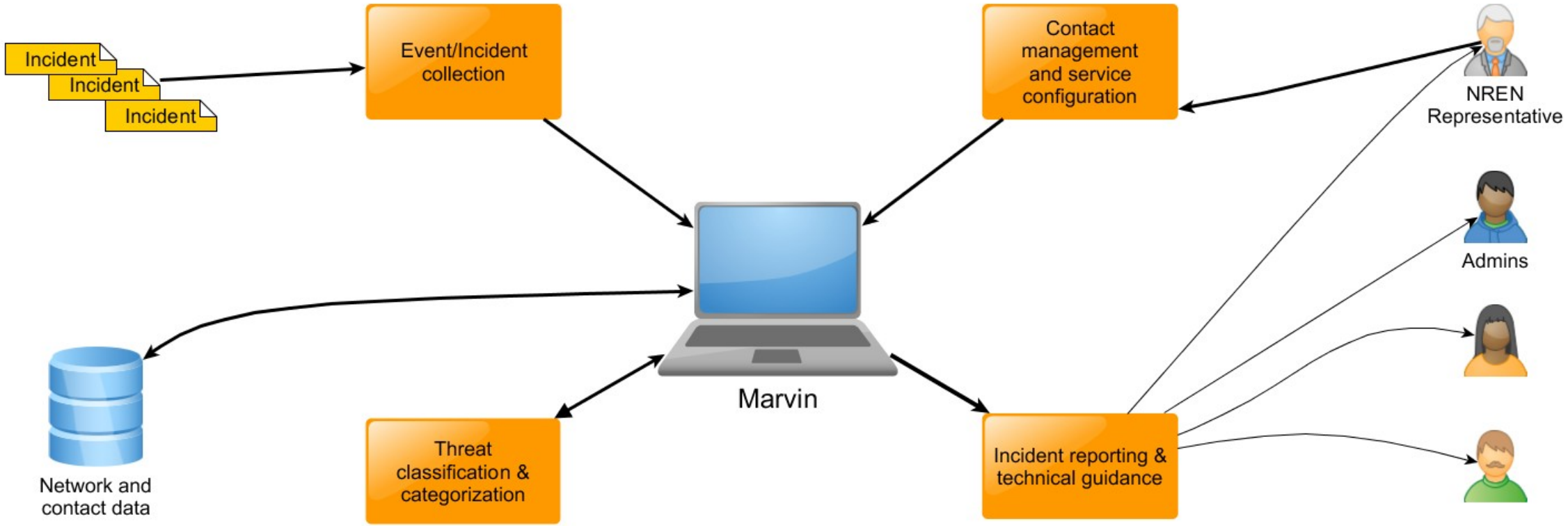
Lessons learned and take aways

- Share insights and experiences with an automated incident handling platform “Marvin”:
  - How to make actionable incident reports from security events
  - Demonstrate architectural requirements for such platform
  - How to ensure data quality
  - Applies to all teams coordinating incidents for their constituency
- **Importance of integrated application combining all aspects of IH**

Incident handling automation



Marvin: Integration of components



Ensure quality

- Select reliable sources of information
  - Security teams and CSIRTs with good reputation
- Specify and monitor processes:
  - **Ensure precise reports**: Avoid reports without further direction: e.g. *“your system might eventually be compromised, but we're not sure.”*
  - Prevent erroneous reports: e.g. false positives
  - Provide informational value: What information must be present to understand the issue? (e.g. exact timestamp, target ip/port, …)

Maintain contact information

- Assign administrative contact to each DFN member site
  - Provide credentials for web-portal account
- Assign network data (netblocks) to each member site
- Allow administrative contact to setup:
  - Sub-networks belonging to the site
  - Assign contact data to sub-networks

Make events actionable

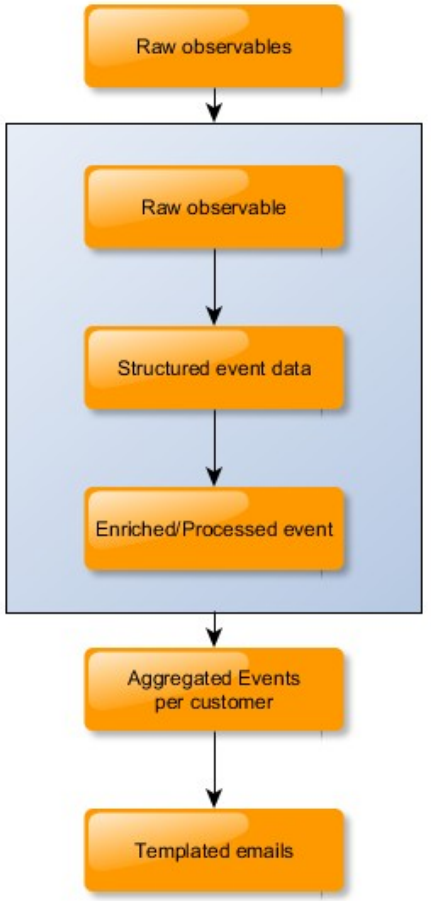
- Each data provider uses a private naming convention and data format
- Normalize events: assign “DFN-CERT” *category* and *diagnosis:*
  - Category (and subcategory): Type of attack
    - Attack/Login, Scan
    - Bot/HTTP, Bot/IRC
    - Configuration/Open Resolver, Amplifier
  - Diagnosis: Type of malware
    - w32.sality, WannaCry, ...

Make events more actionable

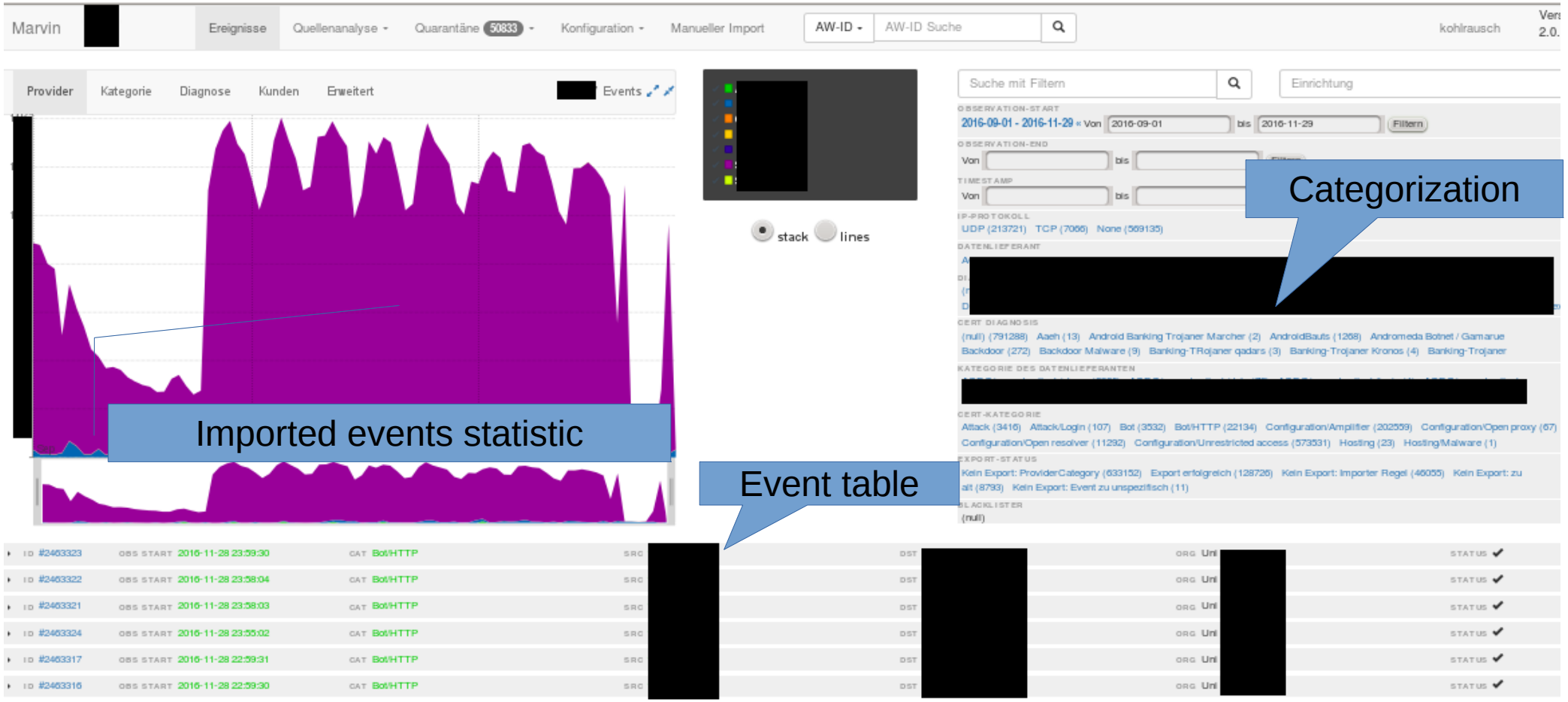
- Add background information and recipe to resolve incident:
  - How to detect
  - How to resolve
  - General information detailing incident handling for major OS
- Produce comprehensible report

Marvin architecture: Data flow

- Marvin Processing I/O
  - From raw observables to templated emails
  - Email based auto warn infrastructure
- Marvin Webservice
  - Webbased analysis toolkit
  - All seen events are saved for analysis purposes
    - Privacy: Limited timeframe for sensitive data

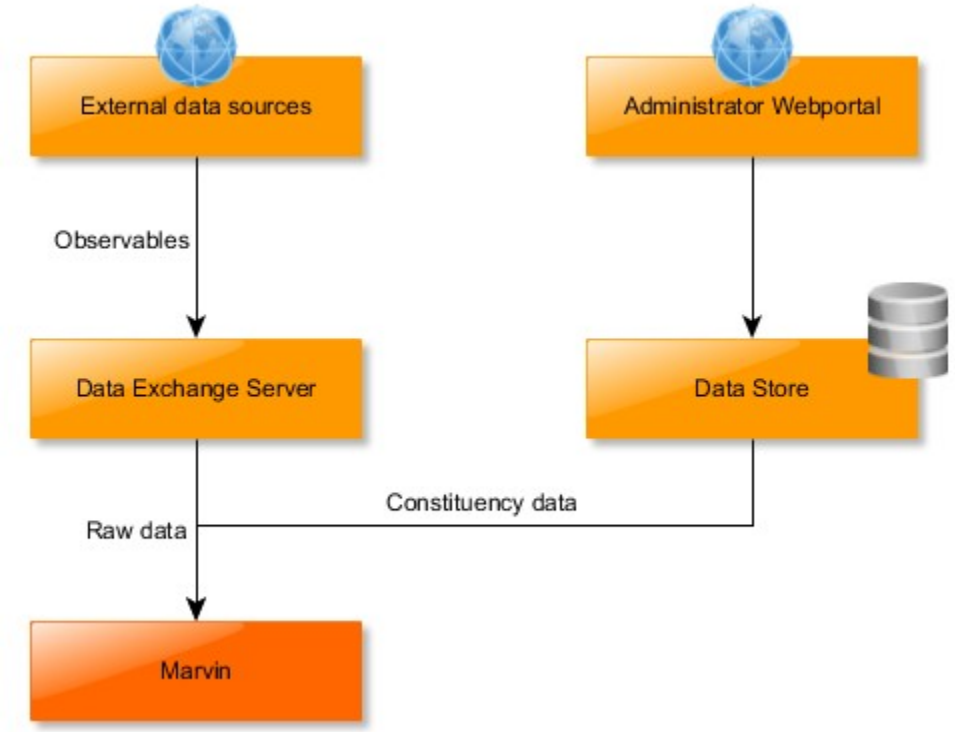


Marvin: Application



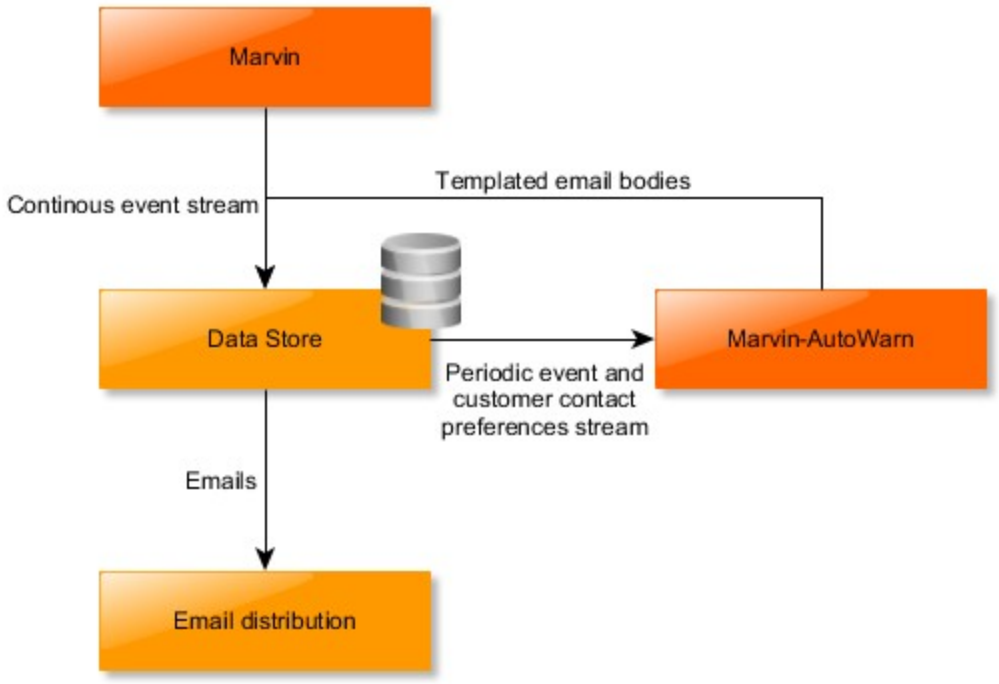
Marvin architecture: Security

- Automation ensuring privacy and security
- Privacy of information providers
  - Traffic Light Protocol
- Security
  - Isolation
- Privacy of constituency
  - All data sources internalized, e.g. no WHOIS, DNS queries



Marvin architecture: Reporting

- Marvin Output
  - Large constituency spanning a variety of requirements, e.g. no reports on weekends, empty notification mails, human-readable, machine-interpretable, etc.
  - Distribution according to admin preferences



QA through statistical models

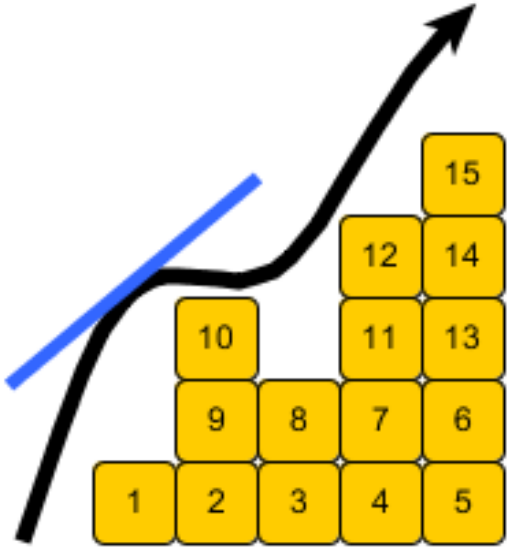
- Data providers may fail to send data
- Servers may fail
- Importers may fail
- Critical situations occur by new attacks, worms, or malware
- University networks may suffer from large scale attack

Data properties

- Number of events vary, but if the number of events is sufficiently high:
  - Fluctuations are randomly distributed
  - Average number of events is quite stable
  - Large-scale disturbances are only caused by significant anomalies (e.g. software failures or significant attacks)

Mathematical model: ARIMA

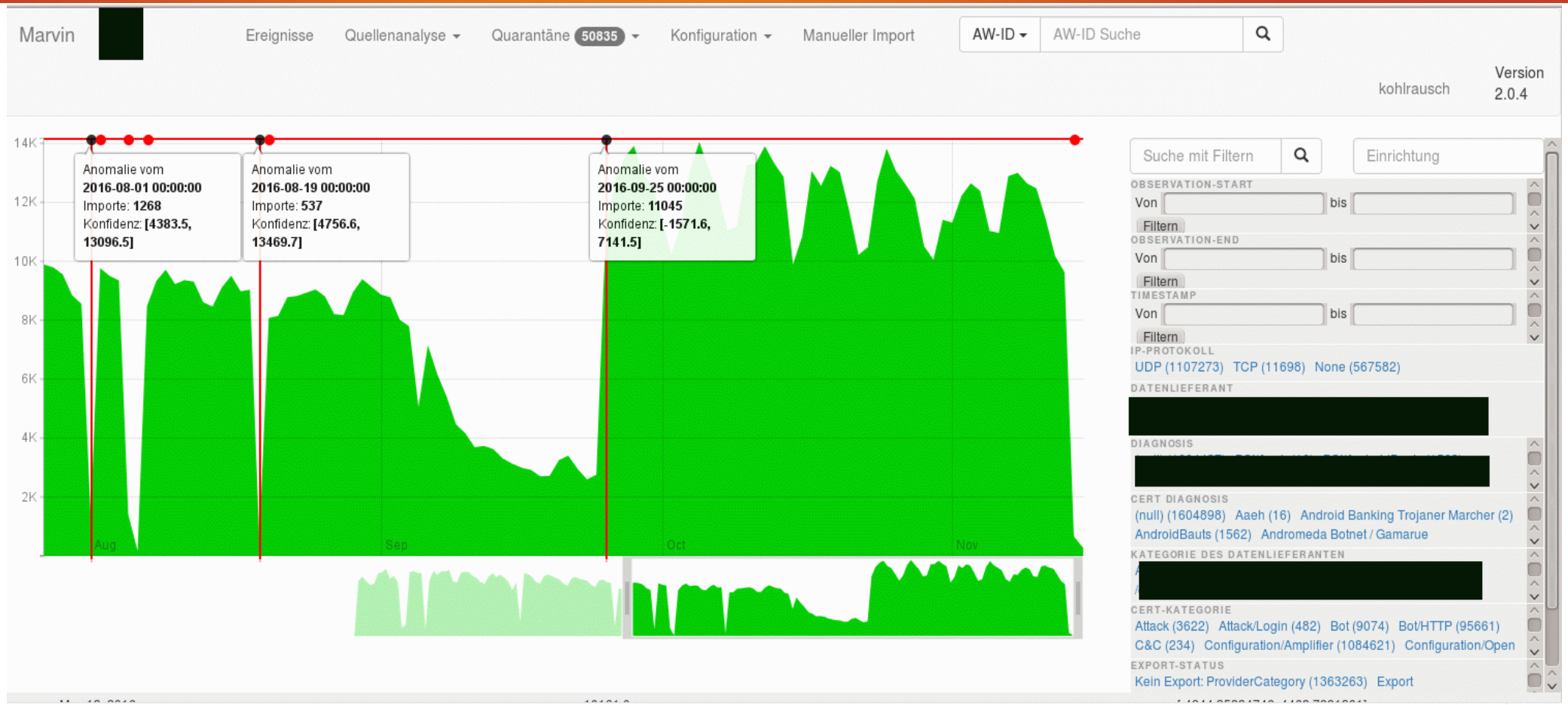
- ARIMA (autoregressive integrated moving average) is used to model time series of events in Marvin
- Approach assumes that time series is stationary:
  - Average is constant
  - Fluctuations are time-independent



ARIMA: Results

- Currently deployed to monitor overall number of events
- Model is applied to moving time window
- Model data is used to predict future data
- Anomaly is detected if the measured value is outside of confidence interval (95%)
- Implementation is based on Python `statsmodels` and `pandas`

ARIMA: Results



Summary and lessons learned

- Marvin became fundamental tool for automated Incident Handling
  - Important to integrate all previously mentioned aspects
- Excellent acceptance in constituency
- So far, only reliable and precise data sources reach the end-user
- Quality assurance is important

Future work

- Deploy incident handling automation beyond DFN
- Integration of new sources
  - For instance: MISP and IntelMQ
- New processes for unreliable data:
  - Finer-granular filtering and data checks enables the adoption of less reliable data sources
  - Monitoring and classification of anomalies
  - Make use of threat criticality

Questions?

Thank you!