SECURE AND RESILIENT ROLLOUT OF SOFTWARE SERVICES

in the Smart Grid

E. Piatkowska[1], D. Umsonst[2], M. Chong[2], C. Gavriluta[1] and P. Smith[1]

{firstname.lastname}@ait.ac.at; umsonst@kth.se; mchong@kth.se

[1] AIT Austrian Institute of Technology

[2] KTH Royal Institute of Technology

This research has received funding in the framework of the joint programming initiative ERA-Net Smart Grids Plus, with support from the European Union’s Horizon 2020 research and innovation programme.

PRESENTATION OUTLINE

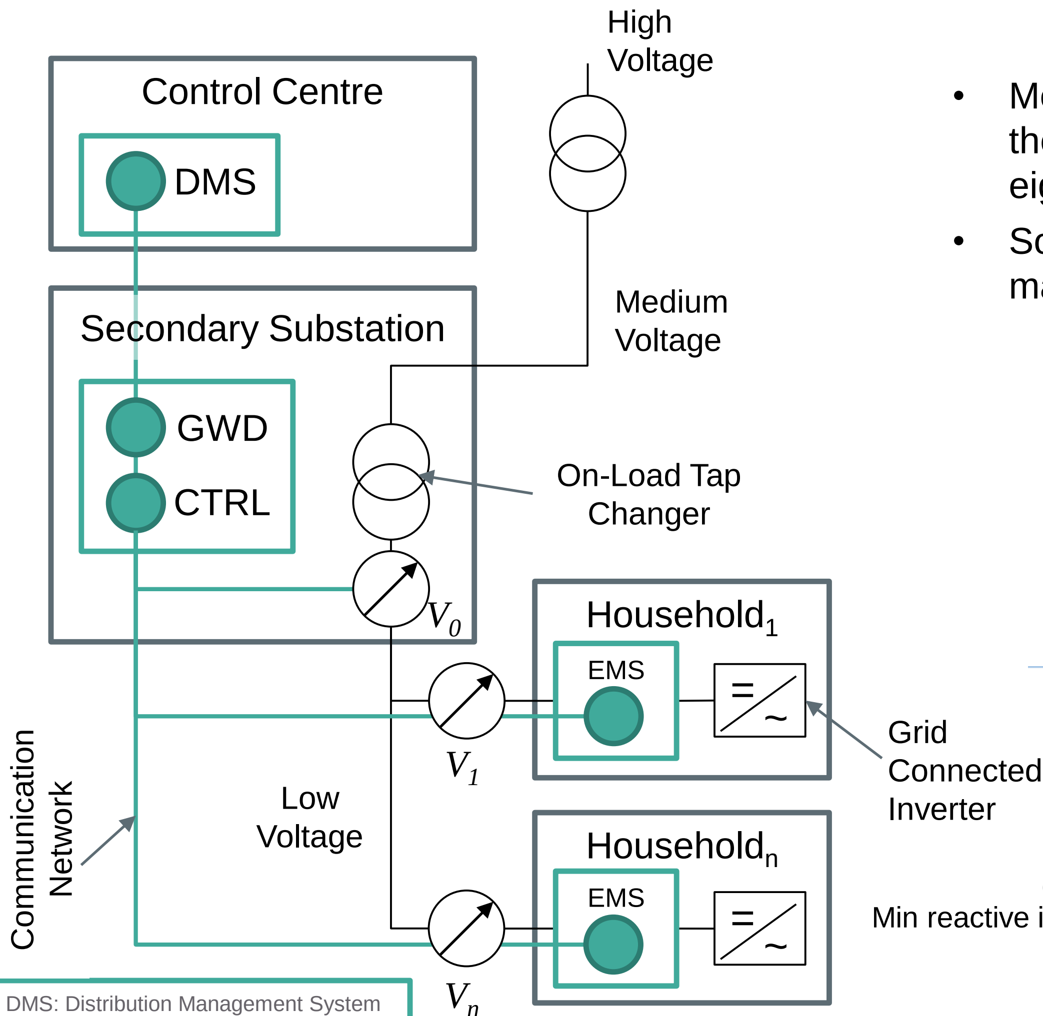
1. Large-scale rollout of software in the smart grid
2. Example scenario in a medium-to-low voltage distribution network
3. Motivate the need for adaptive approach to rolling out software in the Smart Grid
4. Adaptive rollout approaches
5. Detecting failures and reasoning about their root causes

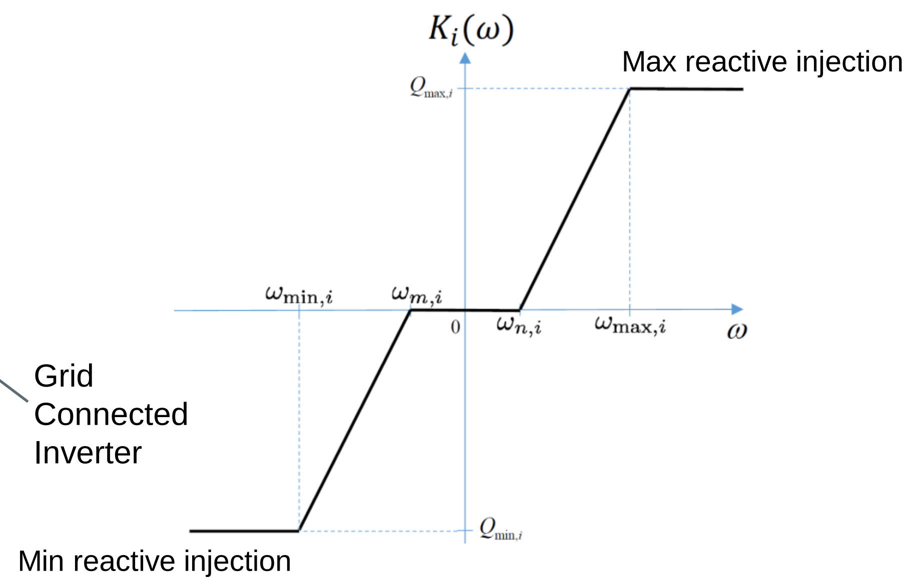
MOTIVATION: LARGE-SCALE SOFTWARE ROLLOUTS IN THE SMART GRID

- Energy distribution systems are undergoing a transition into so-called Smart Grids, which involves the increased use of software systems
- In many cases software-based services are used to support grid control
  - Voltage control in substations
  - Active and reactive power management of inverters
  - Implementation of energy services (e.g., demand-response schemes)
  - Electric vehicle charging
- Consequently, there is a coupling between the state (**correctness**) of software-based systems and power system behaviour
- For several reasons, the software and its configuration in the smart grid will require updating
  - (Security) patches, adaptation to grid behaviour, new services, …
- The LarGo! project is concerned with the **secure and resilient** large-scale rollout of software services in the Smart Grid

EXAMPLE SOFTWARE ROLLOUT SCENARIO

- Medium-to-low voltage network based on the CIGRE benchmark network with eighteen loads
- Software components control voltage levels may be subject to updates



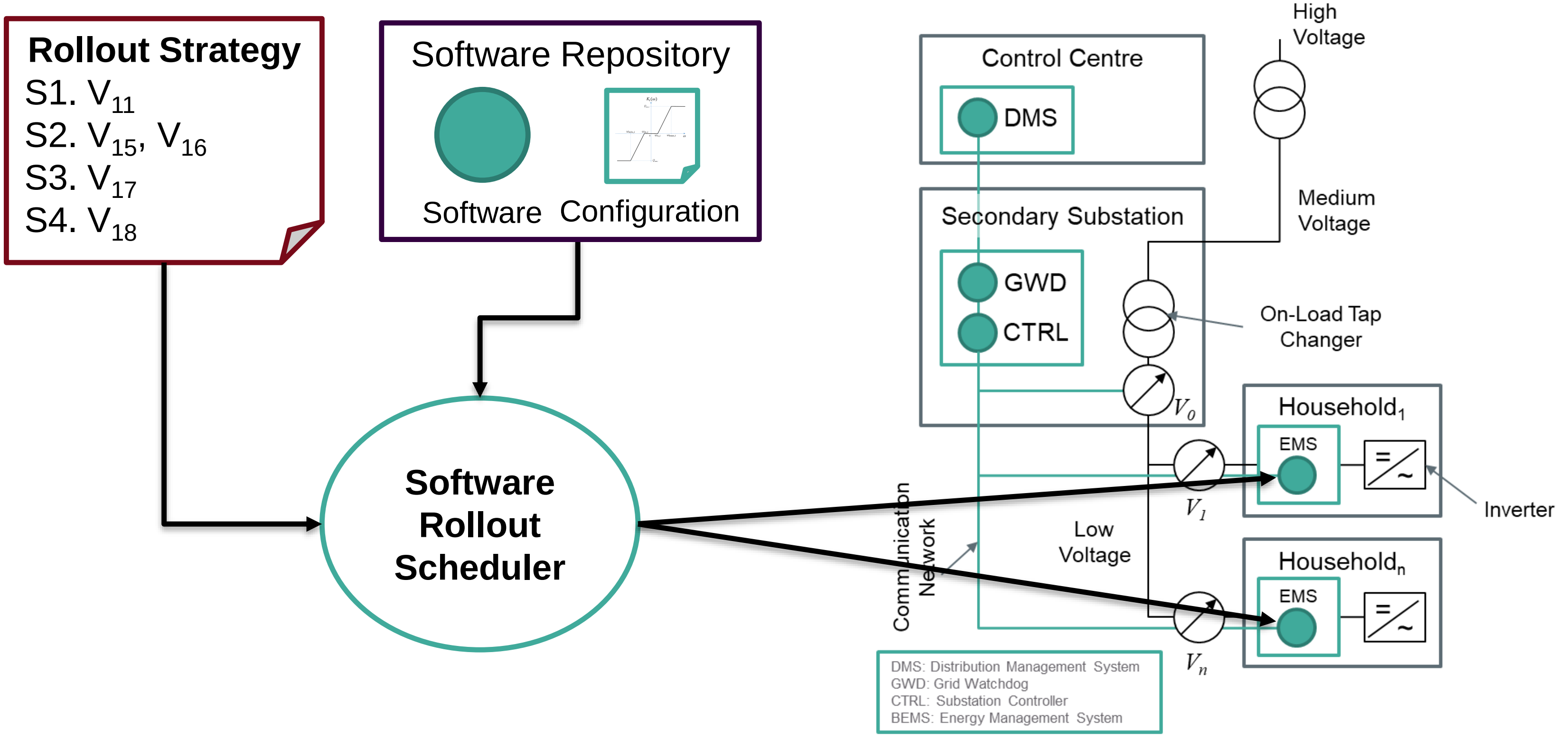


DMS: Distribution Management System
GWD: Grid Watchdog
CTRL: Substation Controller
BEMS: Energy Management System

A patch may involve new settings of $\omega_{\max|\min|m|n}$ and $Q_{\max|\min}$ parameters

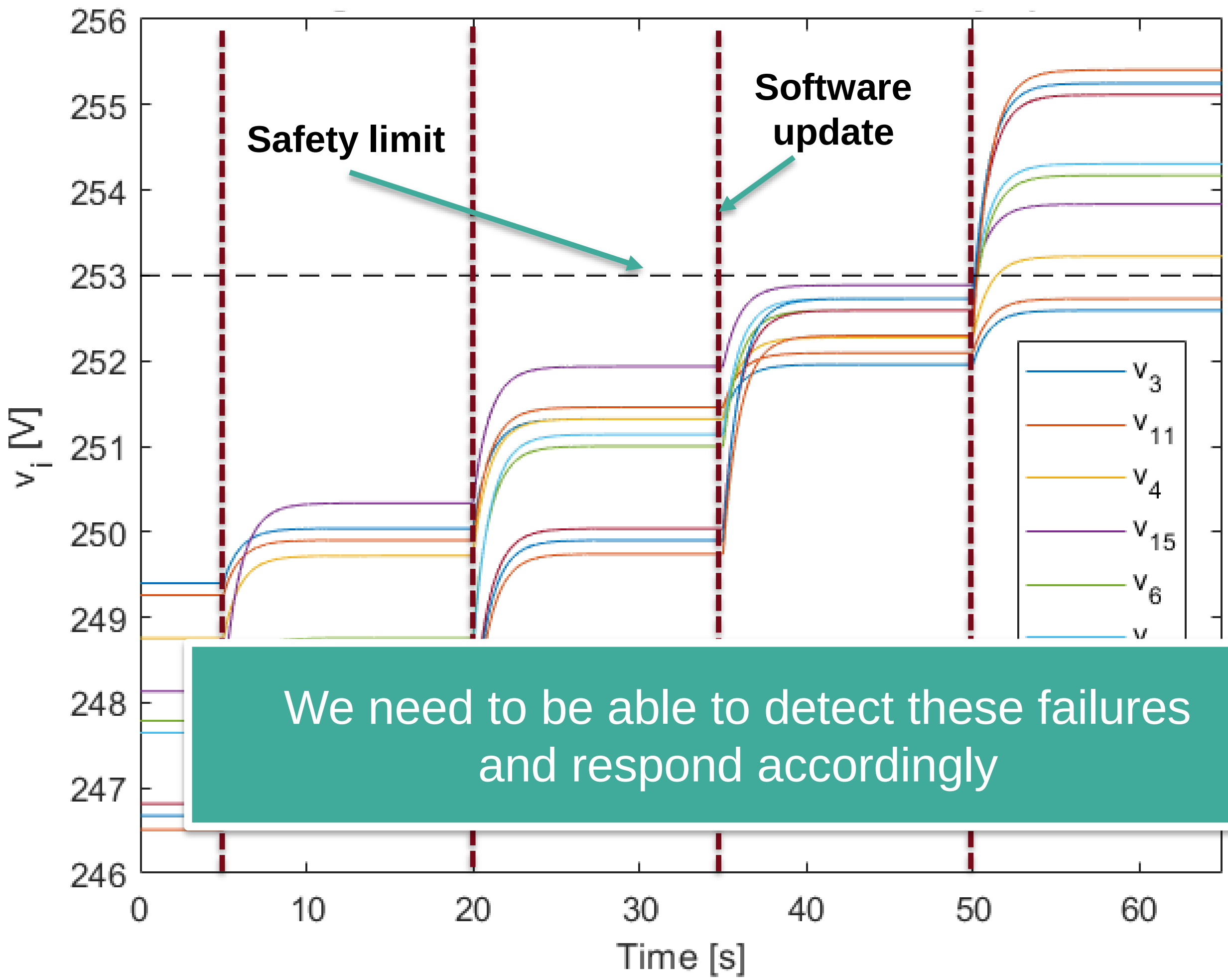
SOFTWARE ROLLOUT SCHEDULER

- Offline process identifies a safe rollout strategy to update EMSs, including the droop law settings; this strategy is executed using a rollout scheduler



AUSTRIAN INSTITUTE OF TECHNOLOGY

AN EXAMPLE SOFTWARE ROLLOUT FAILURE



We need to be able to detect these failures and respond accordingly

Voltage at substation: 251V; Max reactive power for an inverter: 2500V

Scenario

- Patch software in EMSs, including update of Droop law
- Failed update – flipped Droop law configuration
- Inverters inject rather than draw power as voltages increase; problem not corrected during rollout

- Updates at 5s, 20s, 35s, and 50s

  Update order is V_{11}, (V_{15}, V_{16}), V_{17}, V_{18}

- Eventually voltage exceed safety threshold at several locations

ADAPTIVE ROLLOUT STRATEGIES

- Based on the **root cause of a failure**, different responses to failures may be desirable, e.g., to expedite a large-scale rollout

Strategy	Example Root Causes of Failure
Skip and Continue	**Local and Persistent** Device misconfigurations; mismatch between expected and actual target system state for ope…
Retry and Continue	**Local and Tra…** …
Hal…	…misconfigurations (cf. droop law); …ystem state mismatches (asset mgmt.)
Rollb…	Optionally, it could be desirable to rollback to a previous known-good state, although this may not be desirable or possible

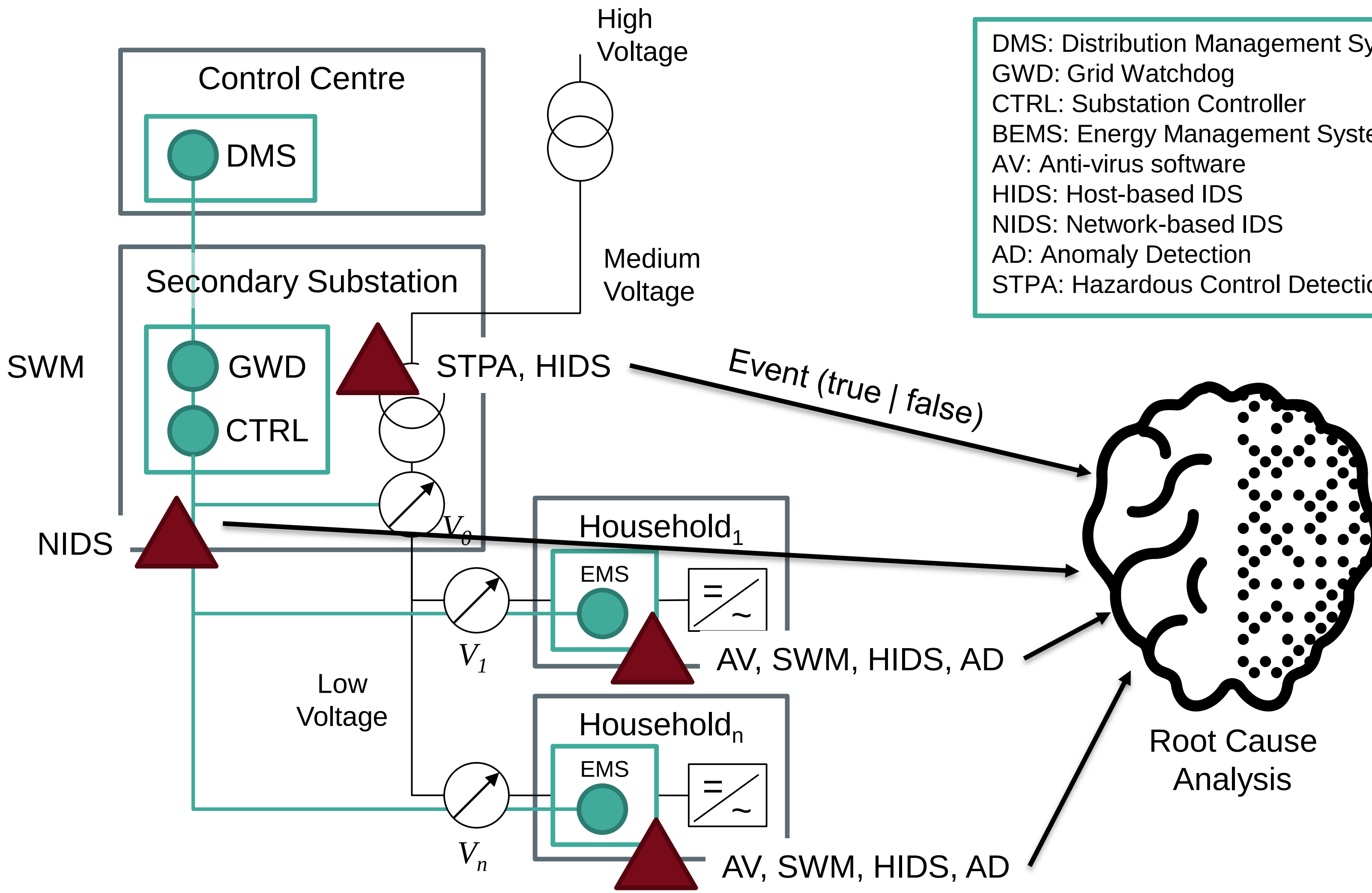
Q: How do we determine the root cause of a failure programmatically?

DETECTING SOFTWARE ROLLOUT FAILURES

- To determine the root cause of a software rollout failure, distributed “sensors” are required, located in the substation and EMSs

Sensor	Description
AV: Anti-virus software	A host-based antivirus system running on the EMSs
HIDS: Host-based IDS	A host-based IDS running on the EMSs (e.g., OSSEC)
NIDS: Network-based IDS	A network-based IDS running on the EMSs and in the substation (SSN) (e.g., Snort)
AD: Anomaly Detection	An anomaly detection system that identifies unusual voltage measurements at the EMSs, e.g., based on residuals
SWM: Software Manager	A software that is located at the EMS that checks whether a software update has completed successfully
STPA: Hazardous Control Detection	A system that checks whether control actions that are carried out in the substation could cause hazards, based on results from an STPA analysis

DEPLOYMENT OF DISTRIBUTED SENSORS



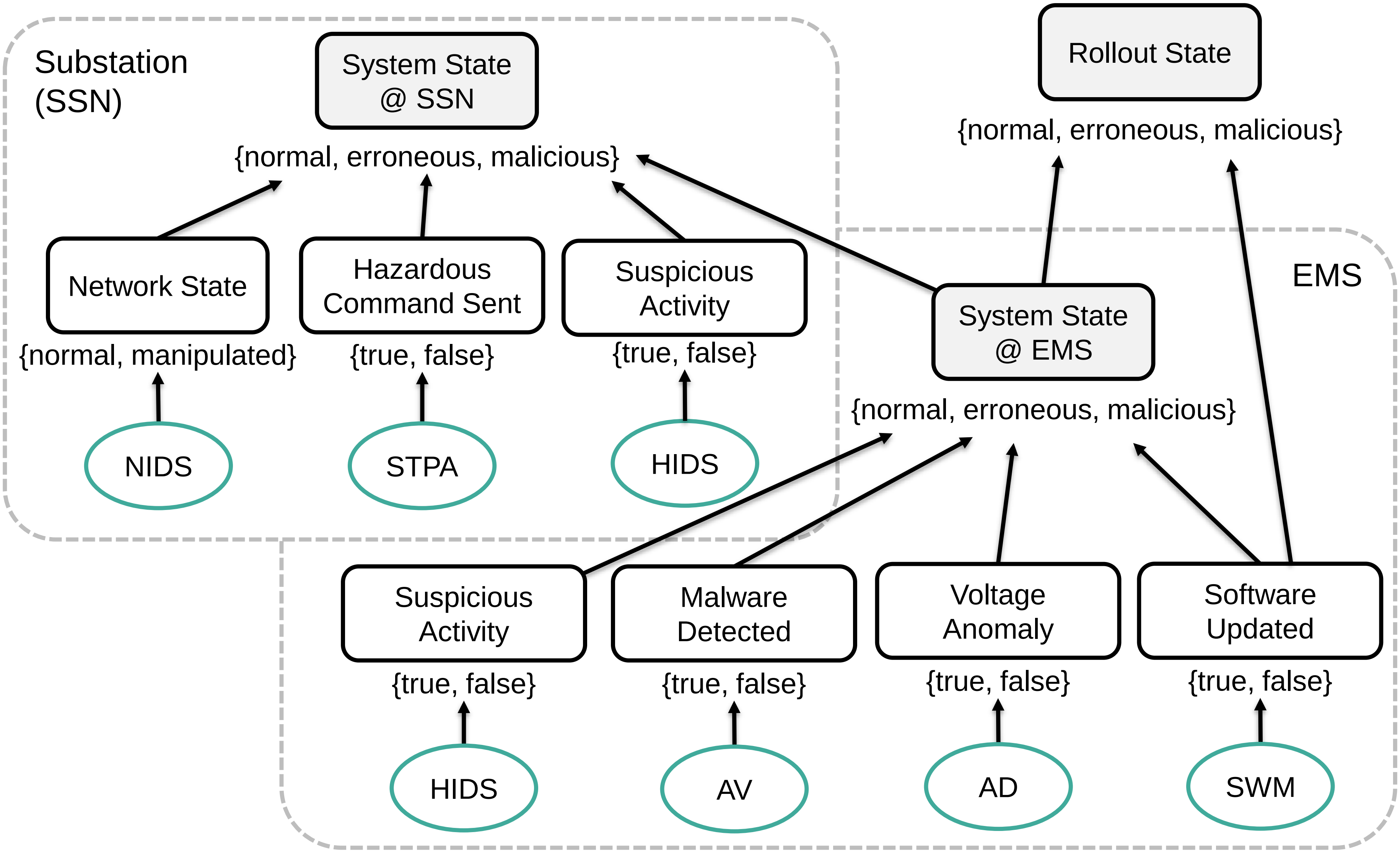
DMS: Distribution Management System
GWD: Grid Watchdog
CTRL: Substation Controller
BEMS: Energy Management System
AV: Anti-virus software
HIDS: Host-based IDS
NIDS: Network-based IDS
AD: Anomaly Detection
STPA: Hazardous Control Detection

ROOT CAUSE ANALYSIS WITH EVIDENTIAL NETWORKS

- An **evidential network** is a graph structure for knowledge representation and inference
- Nodes in the graph represent **variables**, e.g.:
  - Control system state
  - HIDS and NIDS alarms
- Variables have a **frame** that defines their mutually exclusive values
- Relations between variables are given as **mass functions** that describe beliefs
- **Dempster Shafer (DS) theory** allows relation implication rules with uncertainty measures
- Inference within the evidential network is achieved by two operators, called **combination** and **marginalisation**

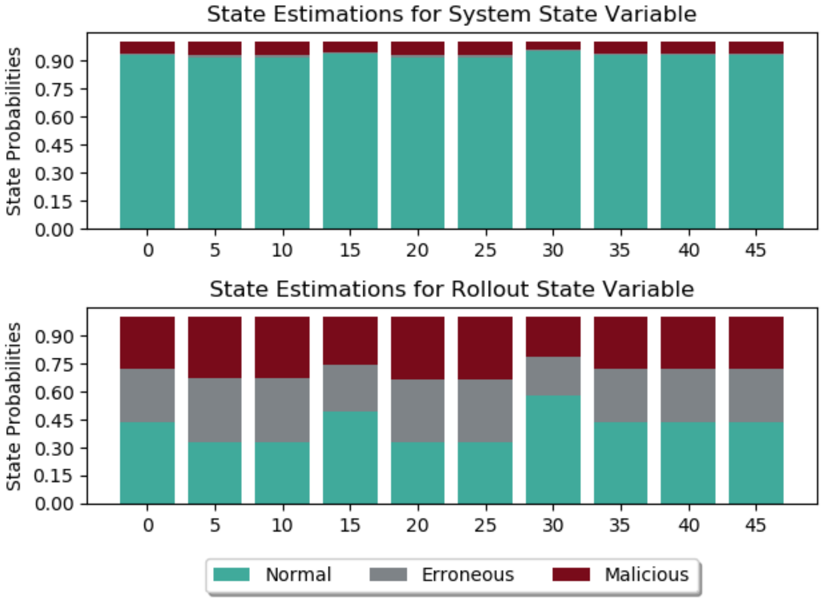
P. P. Shenoy, A valuation-based language for expert systems,
International Journal of Approximate Reasoning, Vol. 3, pp. 383–411, 1989.

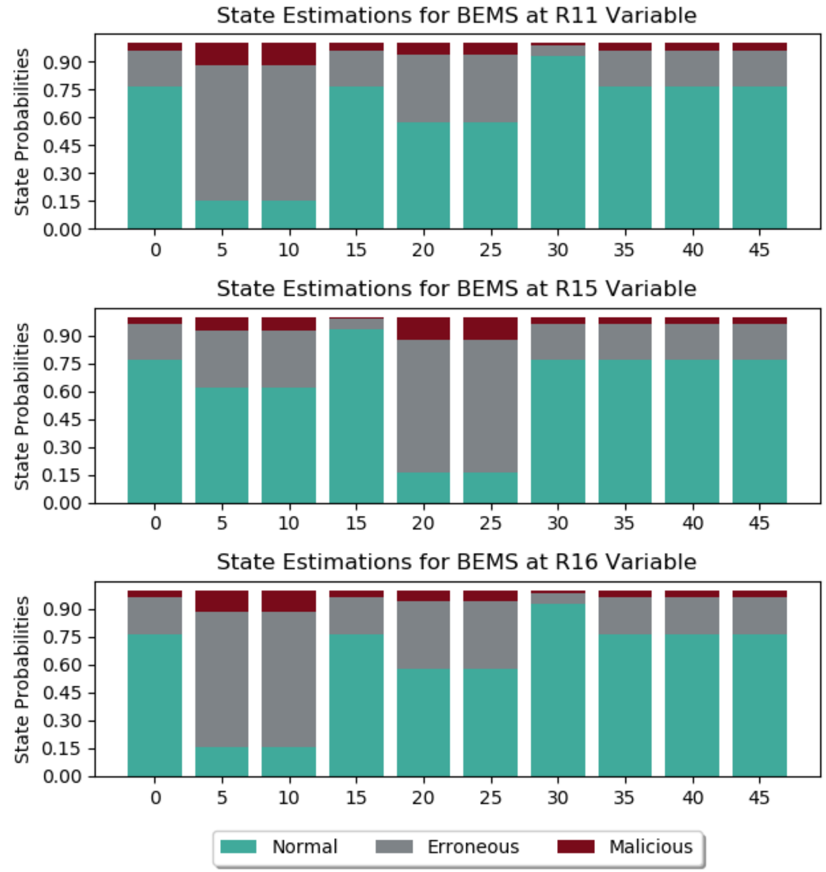
ROLLOUT SCENARIO
EVIDENTIAL NETWORK



SCENARIO 1:
NORMAL BEHAVIOUR DURING A ROLLOUT

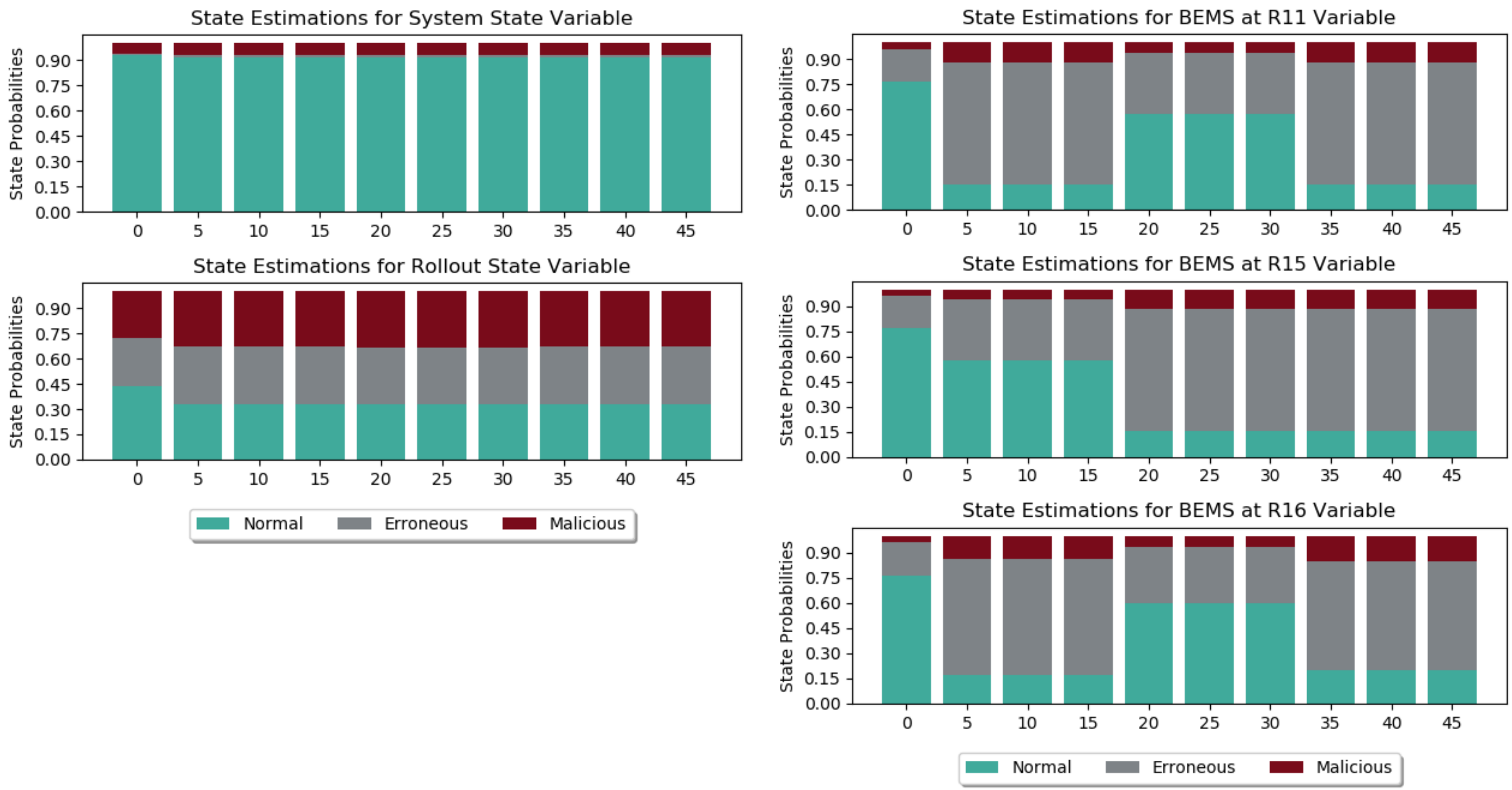
- BEMS report voltage anomalies – small perturbations within a limited time frame are considered normal





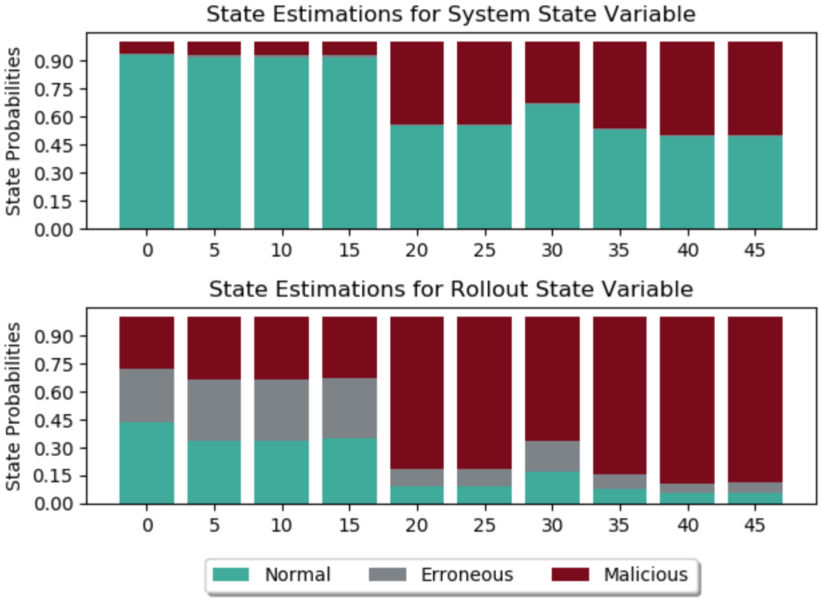
SCENARIO 2: MISCONFIGURATION OF DROOP LAW

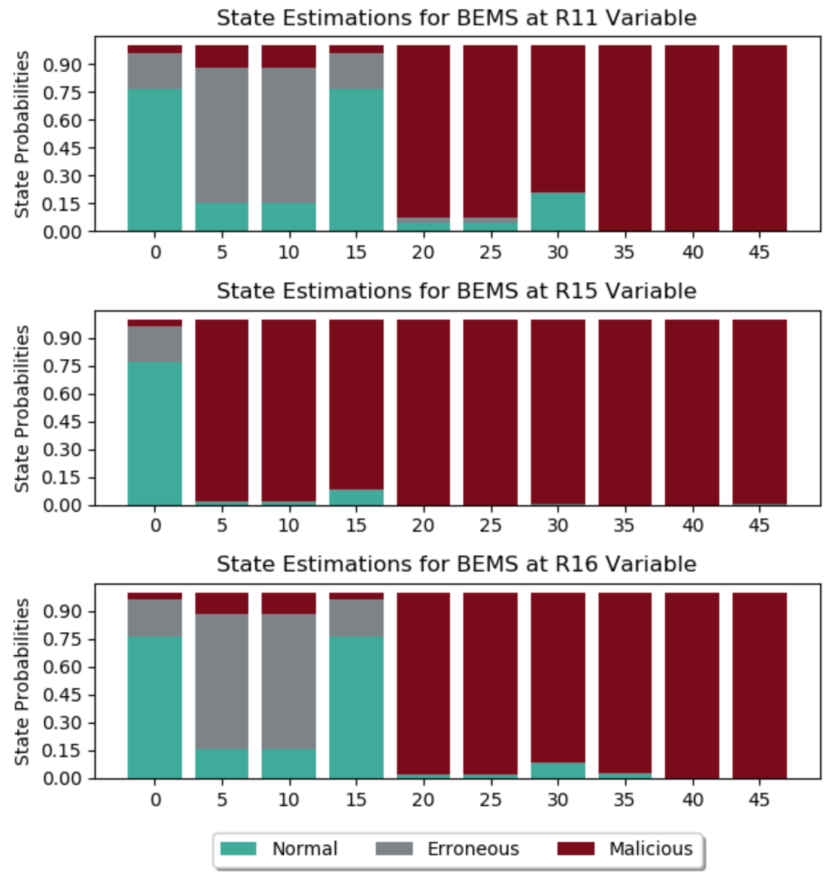
- Failed Rollout results in persistent disruptions in grid operation
- Sign error introduced to PV inverter controllers at node R11, R15, and R16



SCENARIO 3:
MALWARE ON THE EMSs

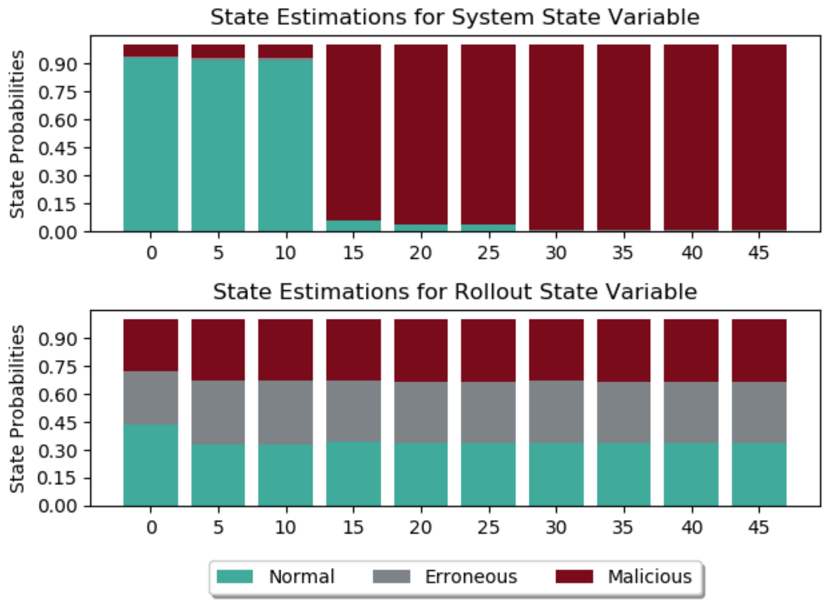
- Compromised rollout results in malware being installed on nodes (BEMS)

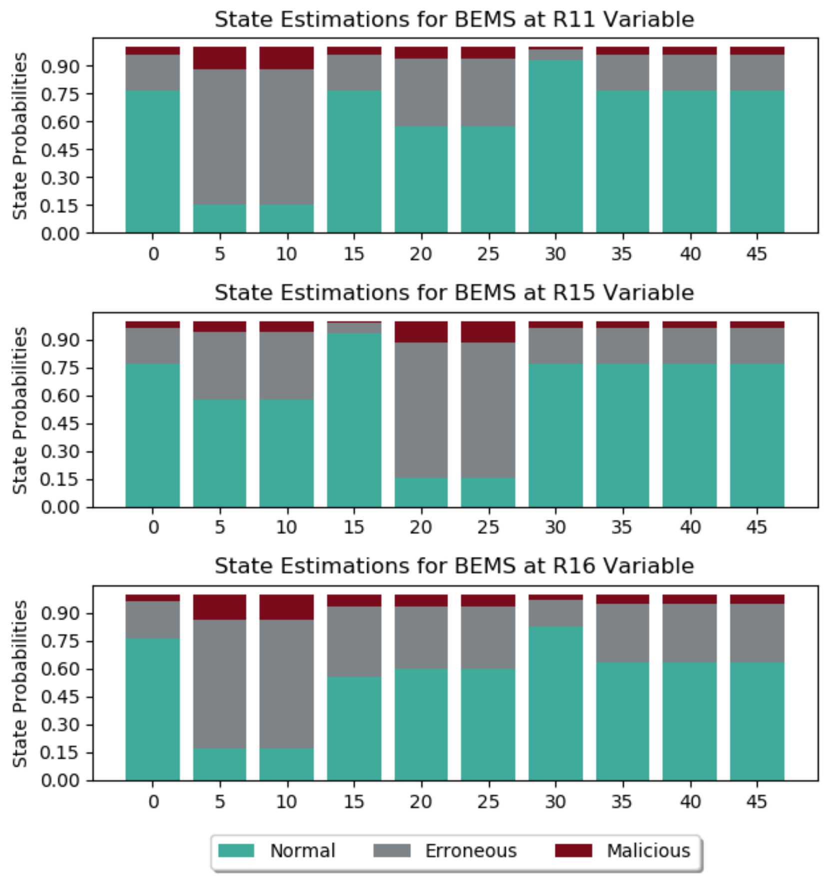




SCENARIO 4: MAN-IN-THE-MIDDLE ATTACK

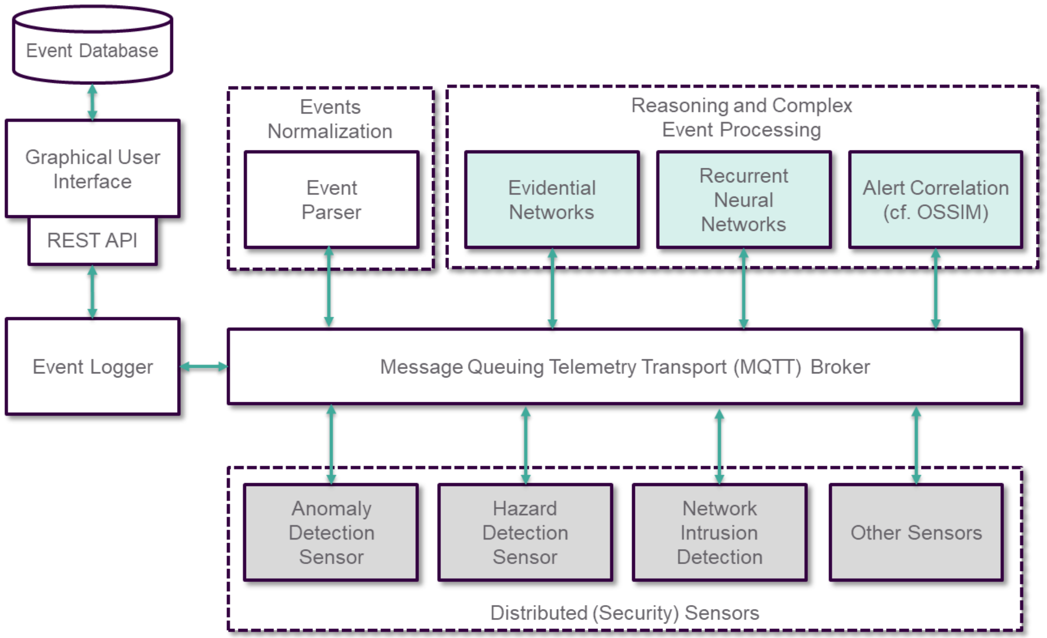
- Man-in-the-middle attack performed during the rollout to compromise communication between voltage sensors and voltage control at the substation
- Integrity attack performed to trigger unnecessary or unsafe control actions





CAUSAL ANALYSIS DEPLOYMENT ARCHITECTURE

- Event-driven architecture using microservices
- Communication between components with MQTT – an MQTT broker serves as an event bus
- Independent from testbed and implementation of components; intended to be scalable and easy to extend
- Main components:
  - Distributed sensors
  - Algorithms for complex event processing
  - Web-based graphical user interface



CONCLUSION AND OUTLOOK

- The Smart Grid contains large amounts of software that is used to support critical control applications
- Software in the Smart Grid will need to be updated
  - (Security) patches, adaptation to grid behaviour, new services, …
- Failures in the software rollout process can result in power systems consequences
- For large-scale software rollouts, it is desirable to automate the process and adapt the behaviour of the process based on the cause of failures
- Proposed an approach to analysing the root cause of deployment failures based on events generated by distributed sensors
- Future work will involve evaluating the approach in a lab-based environment and large-scale simulations

THANK YOU!

Paul Smith
{firstname.lastname}@ait.ac.at