Westfälische
Wilhelms-Universität
Münster

IT-Security in smart grids

Defence strategies for Remote Terminal Units in SCADA networks with limited communication

living.knowledge
WWUMünster

Christof, Marius, Thomas

27. Februar 2017

Table of contents

- The Scenario

Table of contents

- The Scenario
- Implementation in mosaik
  - Topology Loader
  - RTU Simulation
  - Intrusion Detection System
  - WebVis
  - Hacker Tools
  - Operator Tools

Table of contents

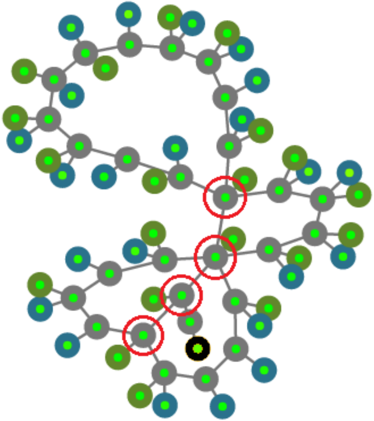
- The Scenario
- Implementation in mosaik
  - Topology Loader
  - RTU Simulation
  - Intrusion Detection System
  - WebVis
  - Hacker Tools
  - Operator Tools
- Attack Scenarios
  - Deterministic attacks
  - Random attacks
  - Defence mechanism specialized attacks
  - Attack to kill the IDS

Table of contents

- Discussion
  - Evaluation
  - Conclusion
  - Future work

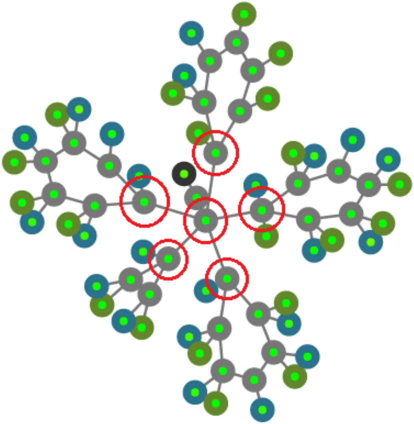
The Scenario

Topologie 1 and 1a



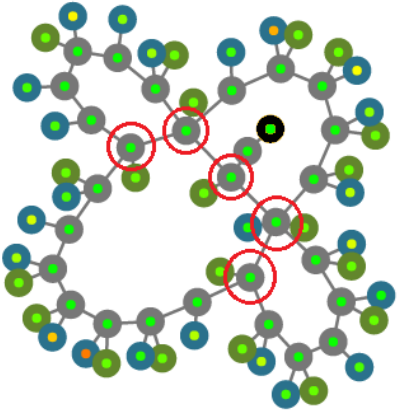
The Scenario

Topologie 2



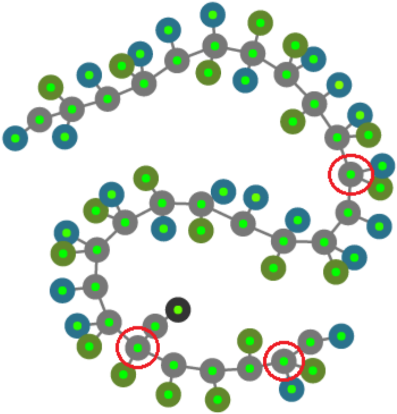
The Scenario

Topologie 3 and 3a



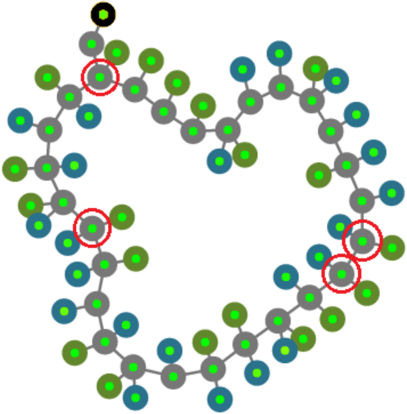
The Scenario

Topologie 4

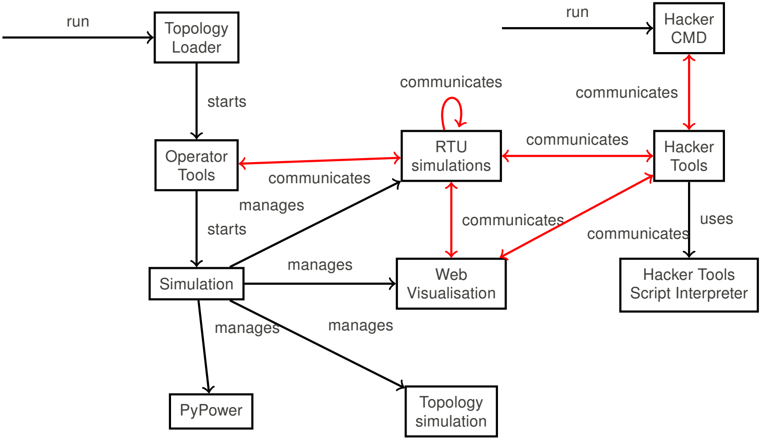


The Scenario

Topologie 5



Implementation in mosaik



Implementation in mosaik

- Topology Loader
  - provides a GUI
  - image of topology selected
  - some simulation configuration

Implementation in mosaik

- Topology Loader
  - provides a GUI
  - image of topology selected
  - some simulation configuration
- RTU Simulation
  - one main MonitoringRTU
  - handles the individual RTU simulations running in separate threads
  - passes data between mosaik and the RTUs
  - RTUs can communicate via server object

Implementation in mosaik

- Intrusion Detection System
  - behaviour specification based

Implementation in mosaik

- Intrusion Detection System
  - behaviour specification based
  - regulation
    - turn of branch when max current is exceeded
    - cut-off values to turn secondary branches on/off

Implementation in mosaik

- Intrusion Detection System
  - behaviour specification based
  - regulation
    - turn of branch when max current is exceeded
    - cut-off values to turn secondary branches on/off
  - validation
    - general system
      - trusted and untrusted sensors
      - warning value
      - warnings and great warnings
    - specific checks
      - Kirchhoff's Law
      - voltage within 10% of expected voltage
      - realistic physical value change

Implementation in mosaik

- Intrusion Detection System
  - validation
    - specific checks
      - voltage angle difference between two nodes not too big
      - check if all sensor values at a node are the same for voltage angle and voltage magnitude
        - + majority rule
        - + mistrust every sensor

Implementation in mosaik

- WebVis
  - switched from executable to Python script
  - added visualisation of attacks and RTU interventions

Implementation in mosaik

- WebVis
  - switched from executable to Python script
  - added visualisation of attacks and RTU interventions
- Hacker Tools
  - Hacker Tools CMD
    - simple command line shell
    - manipulate sensor data or change switch states
    - TCP communication with RTUs' servers and WebVis

Implementation in mosaik

- WebVis
  - switched from executable to Python script
  - added visualisation of attacks and RTU interventions
- Hacker Tools
  - Hacker Tools CMD
    - simple command line shell
    - manipulate sensor data or change switch states
    - TCP communication with RTUs' servers and WebVis
  - Hacker Tools Script Interpreter
    - automating attacks through scripts
    - self-developed script language

Implementation in mosaik

- Hacker Tools Script Interpreter
  - set and get for variables
  - if - then - else
  - for-loop
    - over a range of values
    - over an array
  - random-function
    - number in range
    - element from array
  - array length function
  - wait function (waits a given amount of seconds)

Implementation in mosaik

```
for i in 0 to 1000
  for server in get listservers
    connect server
    for branch in get listbranches
      set v get sensordata of getstate branch False
      setsensor branch, v*1.01
      wait 0.5
    forEnd
  forEnd
forEnd
```

Implementation in mosaik

```
for i in 0 to 1000
  # choose random RTU
  connect random get listservers False
  # iterate through all branches
  for branch in get listbranches
    # per cent to modify sensordata
    set a random 25 300
    # get sensor value of current branch
    set v get sensordata of getstate branch False
    if random 0 1 > 0
      setsensor branch, v*(1+a/100)
    else
      setsensor branch, v*(a/100)
    ifEnd
    wait 0.5
  forEnd
forEnd
```

Implementation in mosaik

- Operator Tools
  - simple GUI showing RTU attack warning messages
  - button to reset RTUs’ trust-label

Attack Scenarios

- Deterministic attacks
  - easy to implement
  - predetermined sequence of commands

Attack Scenarios

- Deterministic attacks
  - easy to implement
  - predetermined sequence of commands
- Random attacks
  - no pattern
  - tries to circumvent pattern recognition

Attack Scenarios

- Deterministic attacks
  - easy to implement
  - predetermined sequence of commands
- Random attacks
  - no pattern
  - tries to circumvent pattern recognition
- Defence mechanism specialized attacks
  - Kirchhoff’s Law
  - mimic natural gradients
  - and more

Attack Scenarios

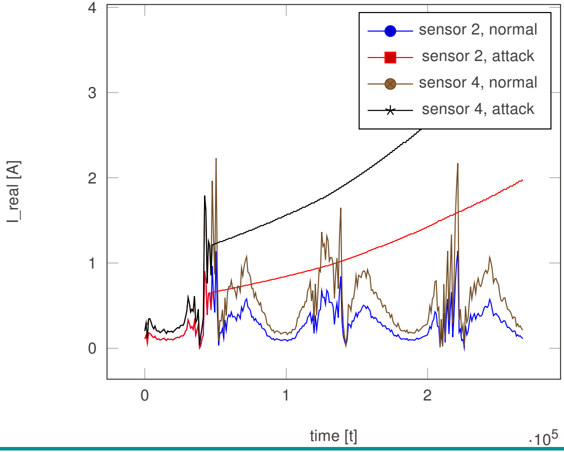
- Attack to kill the IDS
  - heavy attack → IDS declares all sensors as unsafe
  - grid is not controlled any more
  - can reach unsafe states on its own without the IDS noticing

Discussion

- Evaluation
  - sensor value logging
  - specific and random attack
  - executed on topology 1

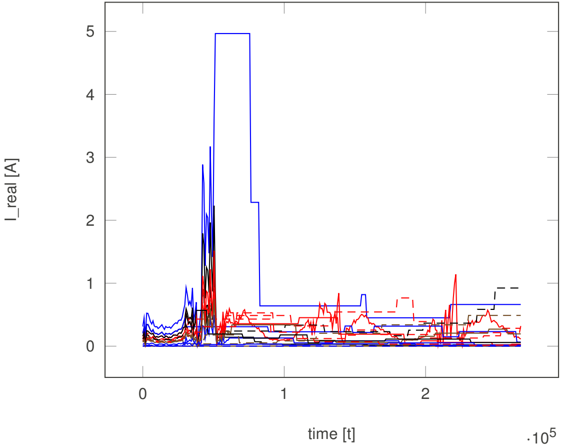
Discussion

- Specific attack



Discussion

- Random attack



Discussion

- Conclusion
  - Kirchhoff’s Law is hard to trick
    - many false positives if a sensor on a node is attacked
    - consider majority rule for improvement
  - overall very accurate attack detection
  - low number of false positives

Discussion

- Future Work
  - consider that current decreases in in the grid
  - more extensive command validation
  - take current readings of PVs and houses into account
  - testing if supplementary pattern based attack recognition would be useful
  - maybe add rules to restore the trust of a sensor
  - syntax error checks for script interpreter

Demonstration

Thank you for your attention!
Any questions?