Models of attacks in critical infrastructures

Peter Popov,
Centre for Software Reliability, City, University of London, UK

28 - 31 January 2020
77th IFIP WG10.4 Workshop, Reggio Calabria, Italy

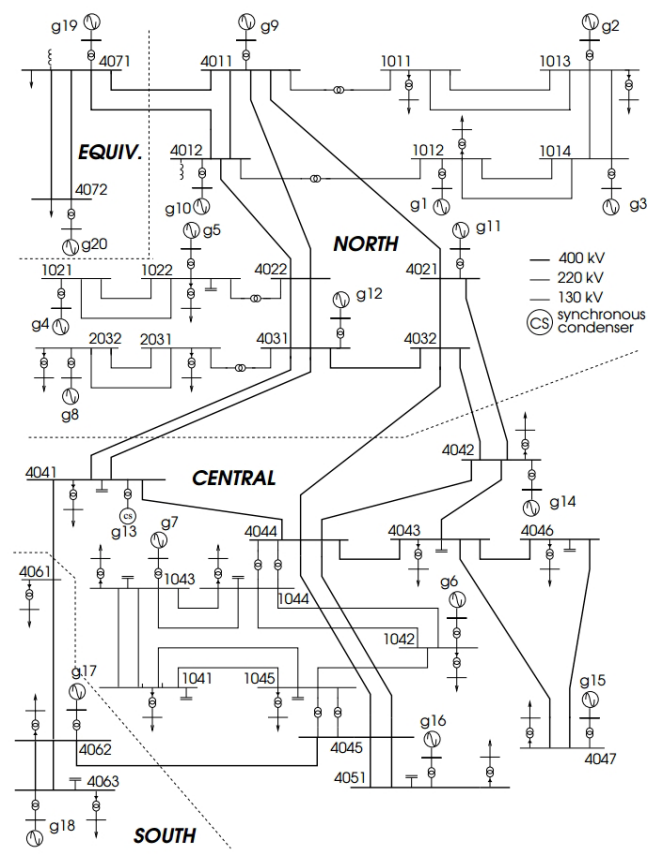
Talk Outline

- Motivation of the Work
  - Risk assessment of complex systems deployed in adverse environment
  - "Best practices" and expert judgement seem insufficient if one wants to answer questions like:
    - "Is the system good enough?", "Is the (cyber) risk acceptable?"
    - "How do I spend a given budget on resilience improvement in a cost effective manner?"
    - "I know that defence-in-Depth is a good practice. How do I apply it in the best possible way?", etc
- The Problem

  "Can we build model of unknown attacks on Critical infrastructures, which are useful for risk assessment?"

- The approach taken
  - Modelling attacks at *different level of abstraction* and looking at how the level of abstraction affects commonly used risk indexes, e.g. the *loss of power* in power transmission systems.
  - A set of simulations is conducted with a *high fidelity model* of a power transmission network (NORDIC - 32) using a set of simulation tools developed at City over the last 10+ years.
- Discussion of findings and the limitations of the approach
- Ways forward

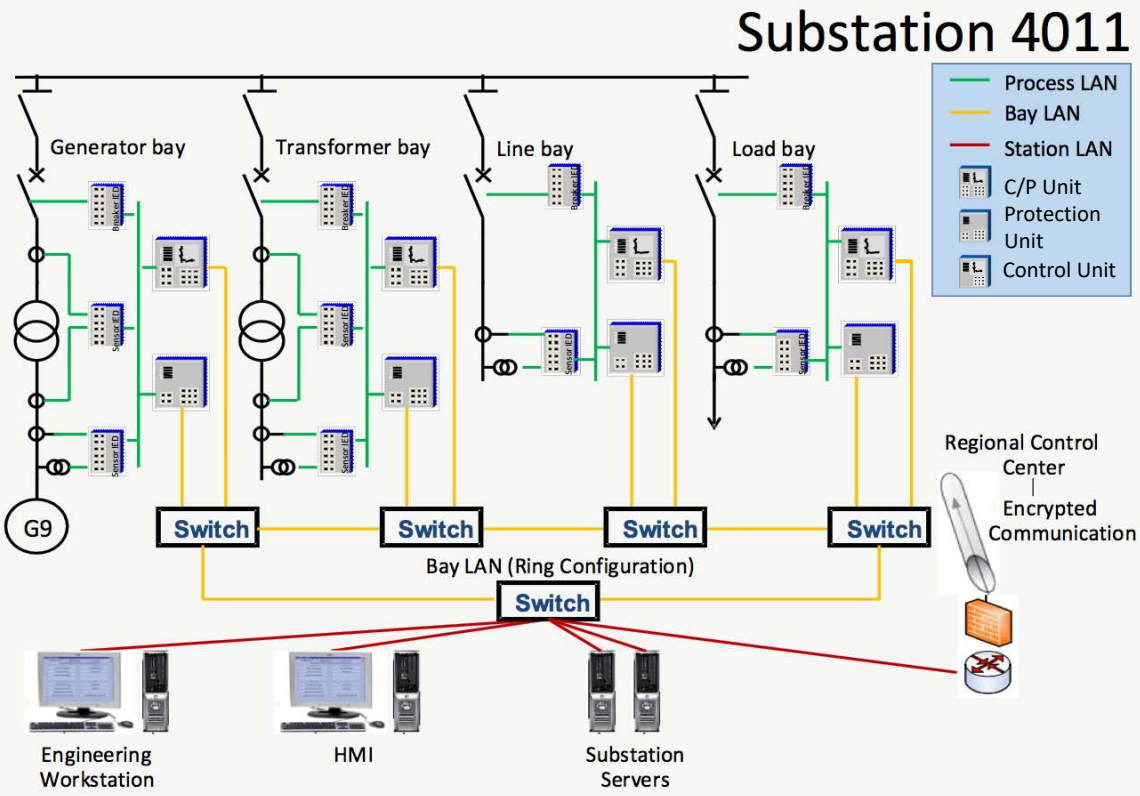
NORDIC - 32 power transmission network



The empirical simulation study is based on model of NORDIC-32, a reference architecture used by power engineers for research:

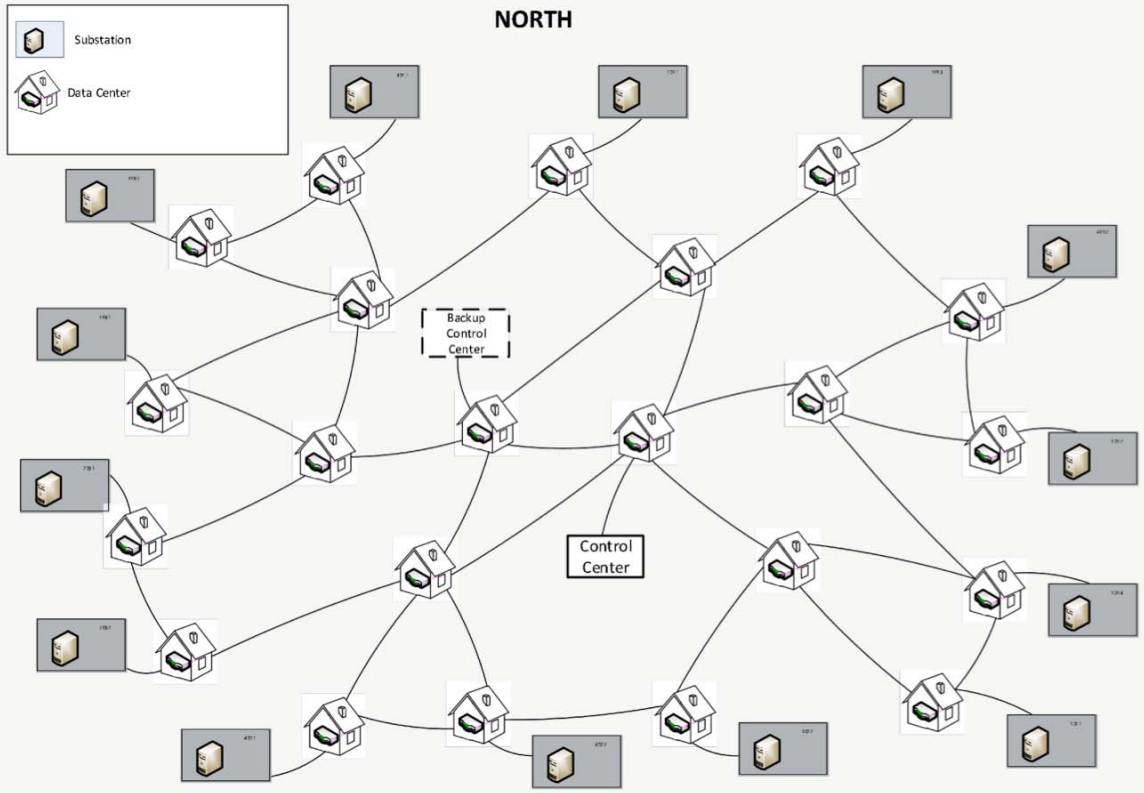
- 4 zones
- 19 generators
- A number of high voltage lines
- A number of “sinks” (connections to power distribution networks)
- A number of bus bars (in substations)

NORDIC – 32: Model of Substations compliant with IEC 61850



- Each substation has:
  - Own LAN
    - I&C devices are connected to physical assets (lines, relays, generators)
    - Control Units, or
    - Protection units
  - Switches connect "bays"
  - Remote control from Control Centres
    - Own firewall

NORDIC – 32: Model of SCADA



- Control of substations done remotely:
  - Control Centres (main and backup)
  - Communication network has redundancy (no single point of failure in the communication network)
  - Remote Control “functions” carried out if communication path exists from Control Centre to the particular sub-station.
  - State estimation and other “special purpose software” (SPS), e.g. to detect “bad data” due to sensor failures, are run in Control centres.
    - Tampered with sensor data (RTU or PMU) may remain undetected and lead to wrong outputs from SPS.

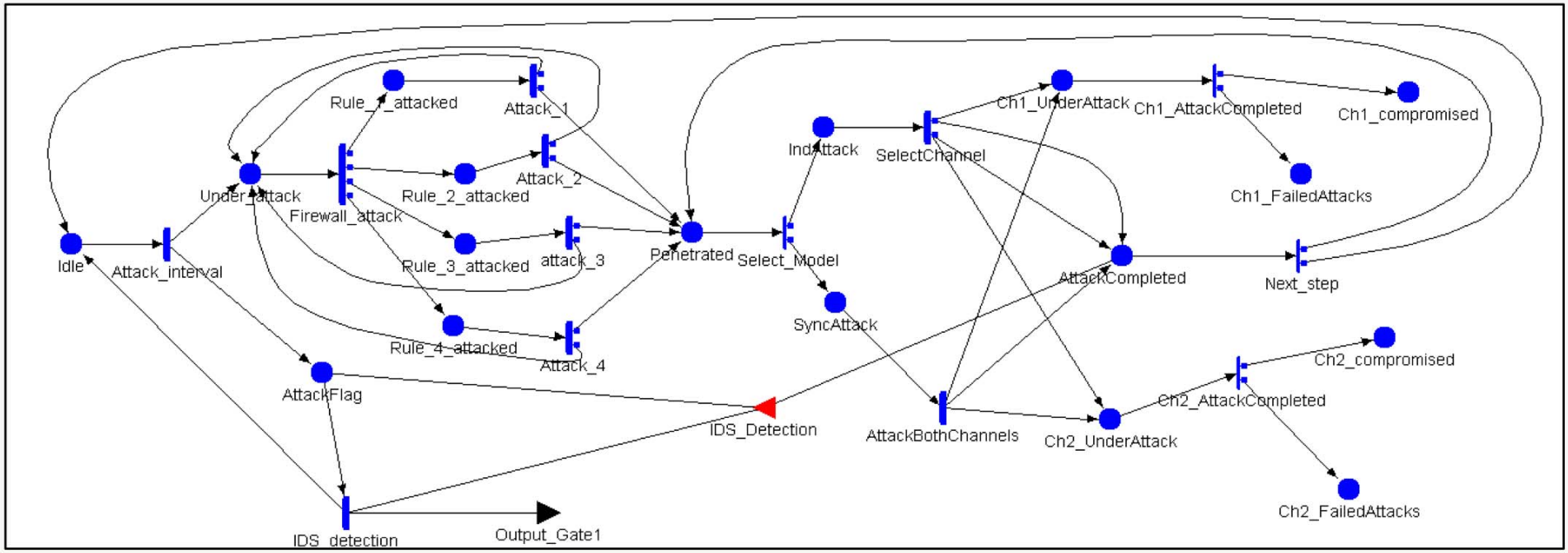
NORDIC – 32: Attack vector

- Large perimeter to defend:
  - Attacks on substations,
    - via firewalls sending wrong commands, wrong readings from the sensors (*will be used for illustration*)
    - Attacking directly the sensors (RTU/PMUs) and make them supply incorrect state values to control centres, thus misleading the SPS (special purpose software) and operators;
  - Attacks on SCADA itself (from DoS to low level protocol specific attacks)
  - Attacks on Control Centres
  - Etc.

Modelling cyber attacks

- Different formalisms to model attacks
  - Attack trees
  - Attack graphs
  - Attack/defence graphs – complex probabilistic extension of attack graphs accounting explicitly for the uncertainty in probabilistic parameters (Bayesian approach)
  - ADVISE – models the impact of attacks at high level, does not go to the level of detail (e.g. power loss) typically used by operators
  - Stochastic Activity Networks – a generic formalism for building state-based probabilistic models. Many similar formalisms.
  - Many others
    - SysML/UML Sequence diagrams (“misuse” cases)
    - UML Activity diagrams
    - Etc.

SAN model of attacks on NORDIC – 32 sub-stn



Models an attack on a NORDIC-32 substation via a firewall

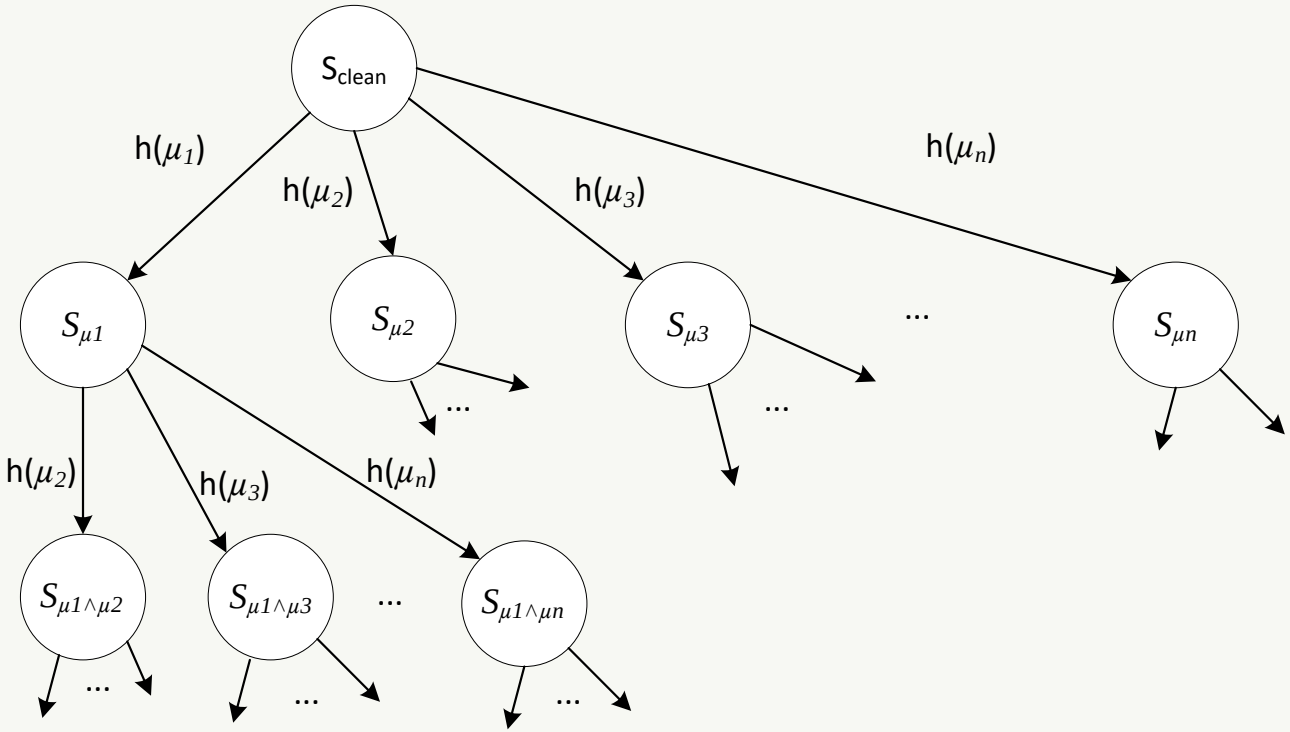
- Intensity of attacks (*Attack_interval*)
- Likelihood of success (*Attack_X*)
- What harm (*CHX_compromised*)
- Possibility of being detected (IDS/IPS) – *IDS_Detection* gate

Known vs. Unknown attacks

- For *known* attacks the model can be very *detailed*
- What do we do about *unknown* attacks?
  - New attacks are emerging all the time. We need to assess the risk from them before we even know they are possible.
  - *Option 1:* Take the "proactive recovery" view
    - No need to model attacks at all as the defence is "perfect"!
    - But this is true only if a set of *necessary conditions* are satisfied.
      - This is NOT always the case. A really very competent Adversary can overcome proactive recovery! What is the cyber-risk for a system with "proactive recovery"?
  - *Option 2:* Select an *abstract model* of the Adversary, such that it is guaranteed to be *applicable to unknown attacks*.
    - What this abstract model would look like?
    - Is it going to be *useful*?

Abstract Model of Attacks (ISSRE'2017)

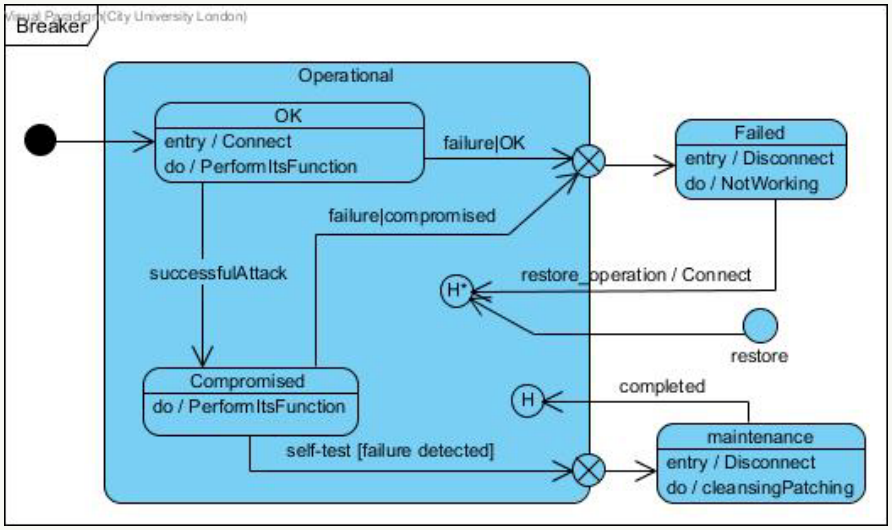
- Malicious demands (MD), $\{\mu_1, \mu_2, \ldots, \mu_n\}$ can be applied to each software component.
- MD are either successful or blocked (e.g. by an intrusion protection system).
- Demands are *serialized* (i.e. at most one demand is applied at a time).

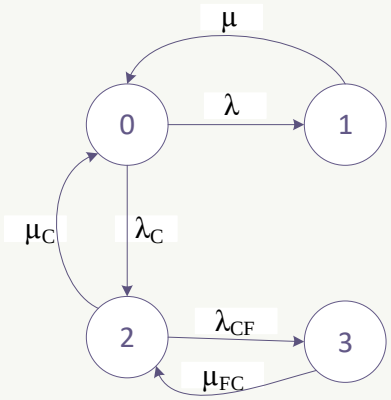


Abstract Model of Attacks (2)

- The view ***taken by many*** is:
  - "once an attack succeeds, the game is over: the Adversary can do whatever they please".
  - The consequences of successful attacks are often not modelled in detail and "the worst" consequences are assumed.
- In this work a ***different view*** is taken: Successful attacks ***merely*** *increase the probability of failure* of the compromised software
  - An Immediate failure after a successful attack ("the game is over scenario") becomes a special case of an ***extreme reliability decay:***
  - A new model parameter emerges with this model: how much worse has software reliability become of the *compromised* software.
- This model is compatible with "proactive recovery" – software "cleansing" restores software reliability to what it was before the compromise.

Abstract Model of Attacks (3)

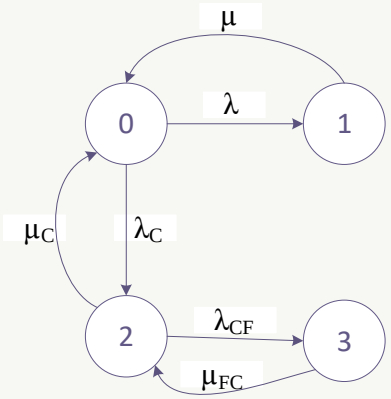




0 – channel is OK
1 – channel is failed
2 – channel is compromised
3 – channel is Failed Compromised

- The model of software behaviour is captured in two diagrams:
  - UML state machine (of a breaker component in NORDIC - 32):
    - The operational state consists of 2 states:
      - OK (Normal operation)
      - Compromised (after a successful attack)
  - Markov chain
    - State 1 and 2 are the usual states (OK and Failure due to accidental fault)
    - State 2 and 3 emerge as a result of a software compromise.

Abstract Model of Attacks (4)



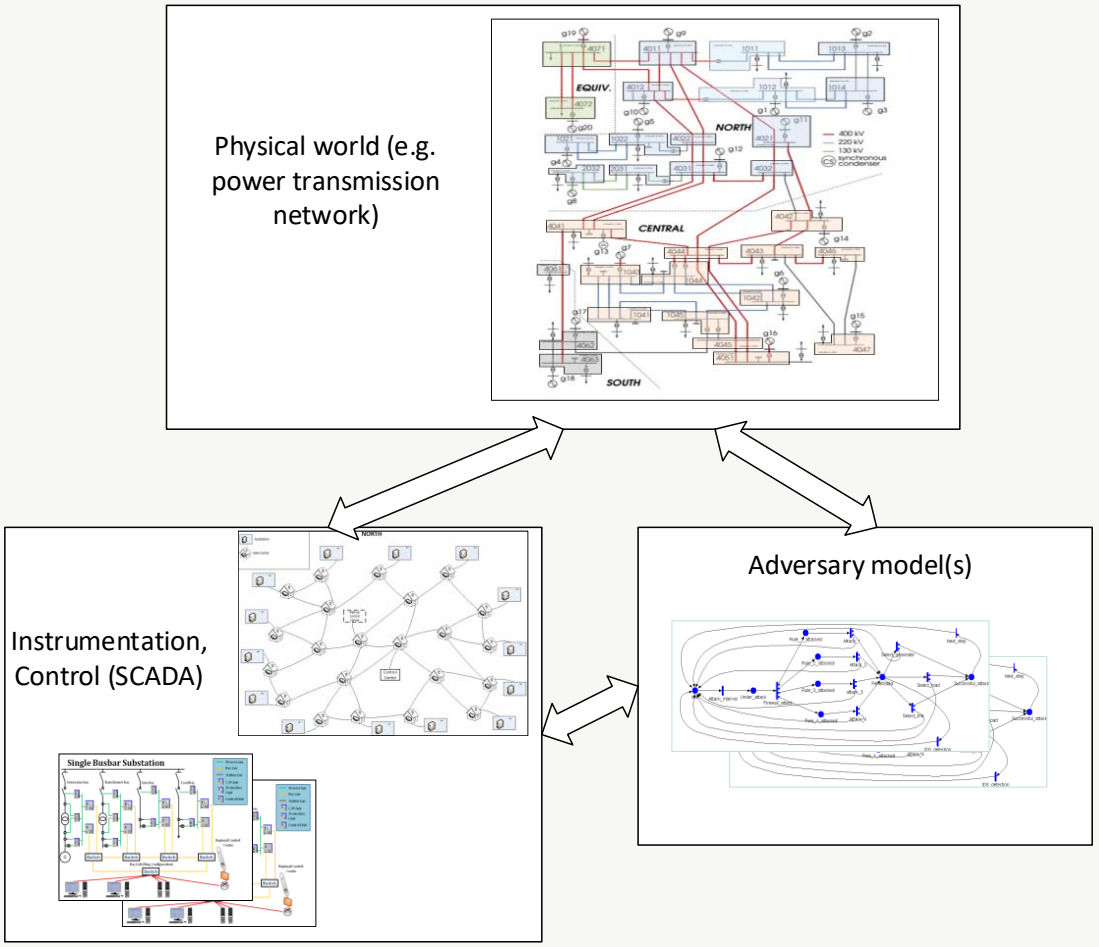
0 – channel is OK
1 – channel is failed
2 – channel is compromised
3 – channel is Failed Compromised

- The abstract model of software behaviour under attack seems quite *general* and applicable to MANY (possibly ANY) attack, affecting the behaviour of a software component/device.
- The real question is: “Is this model useful”?
  - This question is address by conducting an extensive simulation study.

The Simulator

At City we built a *Simulator of hybrid systems*, which includes:

1. A model of the physical assets and I&C of a cyber – physical system (CPS)
2. An Adversary model.
   - Different Adversary models can be attached to the same CPS model.



The Simulator (2)

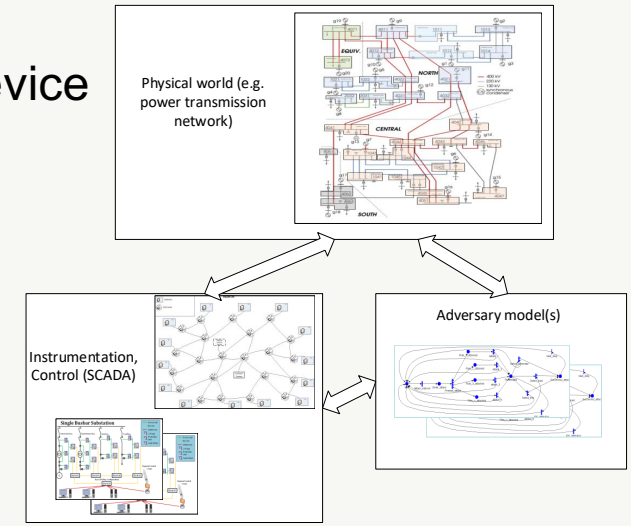
A model of the physical assets and I&C of a CPS includes:

- A set of components (e.g. a generator, a control device, etc.). Each is modelled by a ***stochastic state machine (SSM)***
  - captures accidental failures/repairs and are quite generic.
- Additional ***properties*** are defined for components (e.g. "capacity" of a line, "output power" for a generator, etc.)
- Various *deterministic models* (e.g. power flow calculations, cut sets for the control functions run from a control centre, etc.)
  - These capture the specifics of the application domain
- Dependencies between SSMs are captured by:
  - Probabilistic models (transition rates may be dependent on the states of *other* SSMs),
  - Deterministic models (e.g. tripping lines, if overloading occurs, etc.)
- Models of maintenance (e.g. restoring from component failures, software "cleansing", etc.)

The Simulator (3): Adversary model

Adversary model is tightly coupled with the SSMs

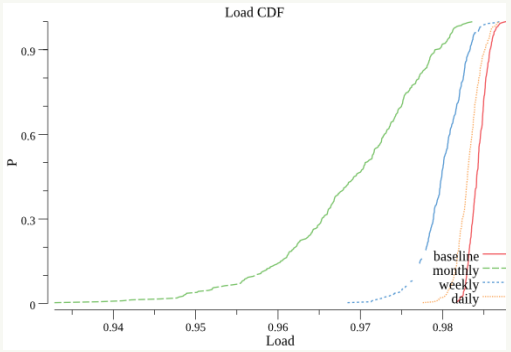
- Successful attacks affect the state of the affected s/w
  - may change the properties of a component, e.g. the protection threshold of a voltage sensitive protection device
  - May lead to an instantaneous transition from OK to “compromised”
  - Etc.
- Secondary effects
  - Successful attacks may change the topology of the respective networks by disconnecting assets (lines, generators, sinks),
  - Affect availability of control functions run from control centres
  - Trigger further effects (e.g. cascade of tripping events through power flows).

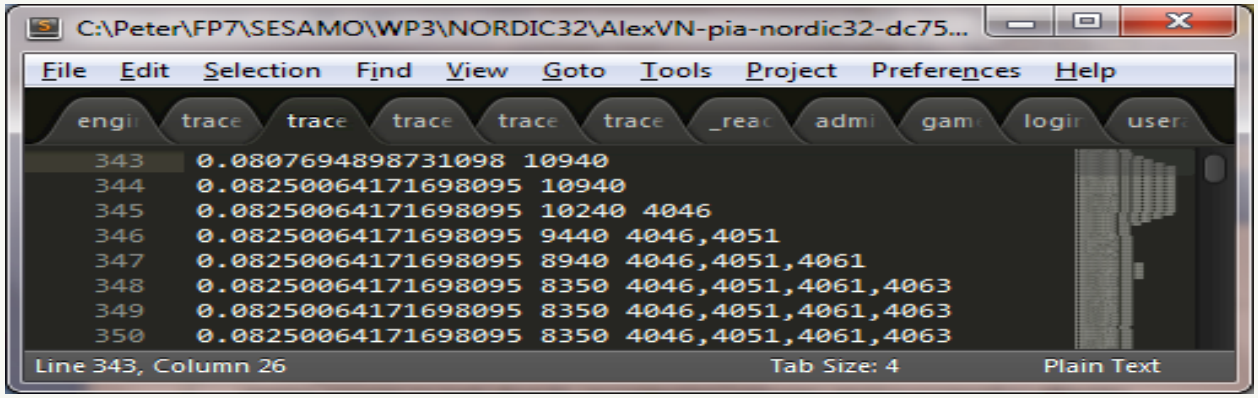


The Simulator (4): Rewards

"Rewards" (i.e. utility function) are implemented by a dedicated plugin which can be changed to account the specific research questions:

- The default reward for NORDIC-32 is the *mean supplied power* as a fraction of the MAX power in NORDIC-32 (10,400 MW).
- Another useful reward plugin is storing a *complete simulation trace* of all events generated (with timestamps) during a simulation run and the complete state of the system (~1500 SSMs with their respective properties).





The Simulator (5): Studies and Solvers

Solver

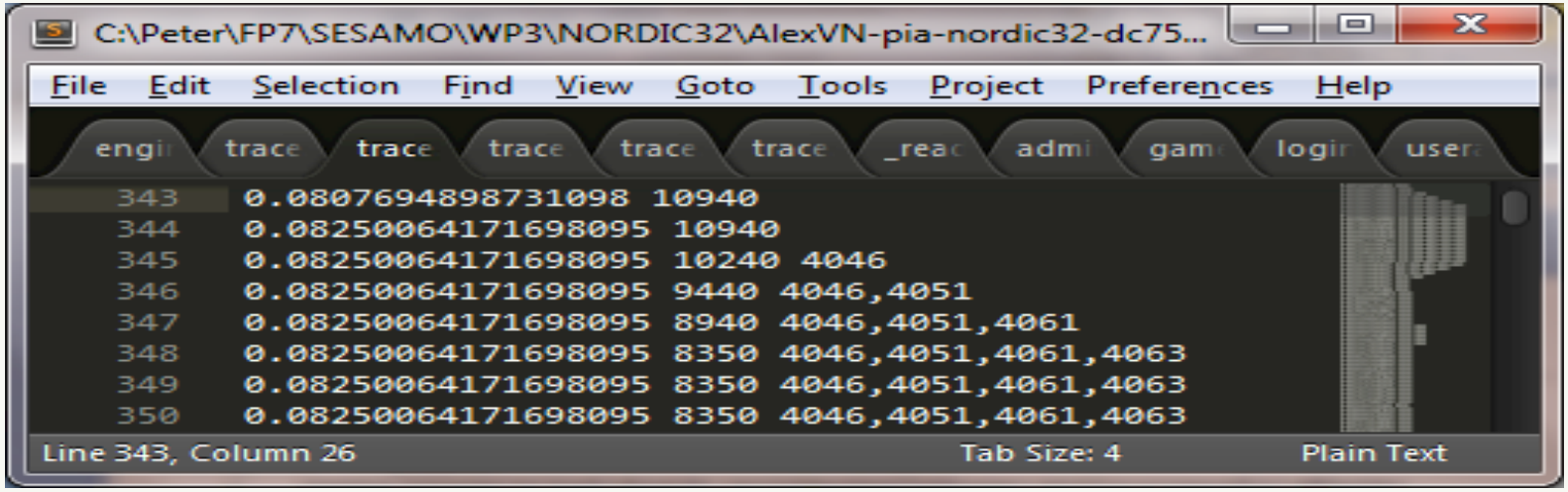
Models are solved via *Monte Carlo simulation.*

Studies

- Similar to the concept of studies in Mobius SAN.
  - A study is the label given to a model and a set of model parameters for this model (topology, probabilistic parameters, property values, etc.)
  - A number of “runs” (i.e. repetitions) is specified for each study:
    - *300 runs* for a study are used in the work reported today
  - Duration of each run is also defined for the study
    - For the studies reported today the duration was set to *10 years of operation* of the NORDIC-32 system.

Further details on reward: Power Loss

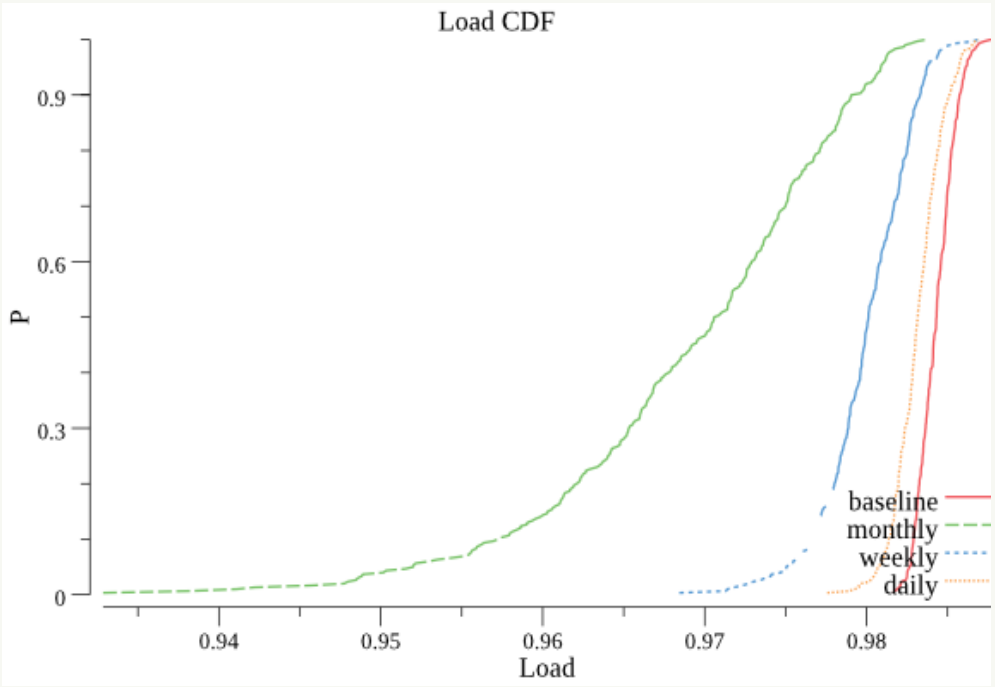
For each run we compute the *average power supplied*, Pi (i= 1,…300), which is a ***random variable*** (varies with i).



We looked at:

- The *distribution* (over 300 runs) of Pi
- The ***average*** supplied power over the chosen interval of 10 years, $E[P_i]$
- The ***standard deviation***, $StD(P_i)$ is a measure of spread of supplied power.
  - Greater value indicates ***greater variability*** of power supply, i.e. more ***unstable*** power supply.

NORDIC – 32: Illustration of Power Loss Distribution



- We used this distribution in the past to rank different attacks on NORDIC – 32:
  - Attacks, which lead to *switching off* some of the assets (generators, lines, sinks)
  - Attacks, which instead *tamper with the configuration of protection devices* (i.e. change the protection threshold)
  - Attacks which lead to tampering with the sensor data (from RTU/PMU) – done by others.

Abstract attack model validation: the setup

- We conduct a number of studies with *two paired system models (PSM)* of NORDIC-32 which differ only in the way the consequences of successful attacks are modelled.

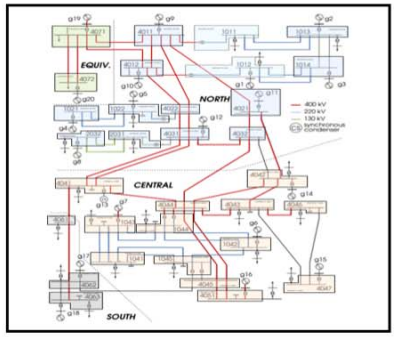
- The system models share the *same model of physical assets:*
  - Topology is the same (i.e. NORDIC - 32)
  - SSMs of the components are the same
    - including their parameters (failure/repair rates, dependencies)
  - Model of recovery (from failure) is the same
    - The rate of recovery is the same

  - *The adversary models of PSM are similar, but not the same.*

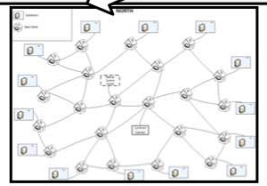
Paired System Models: An Illustration

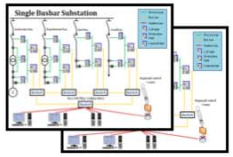
System Model 1

Physical world (e.g. power transmission network)

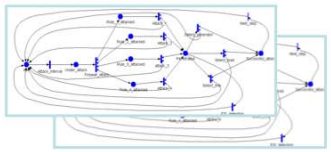


Instrumentation, Control (SCADA)



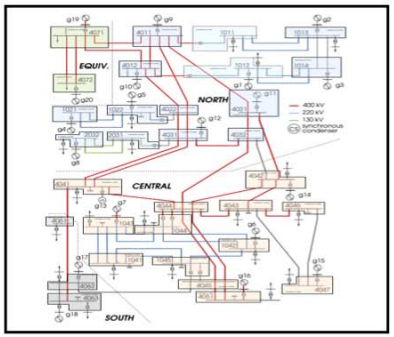


Adversary model(s) *HiFi*

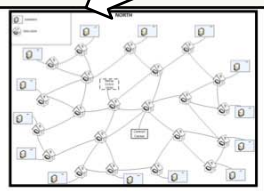


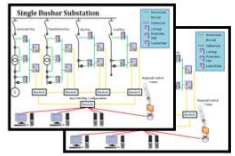
System Model 2

Physical world (e.g. power transmission network)

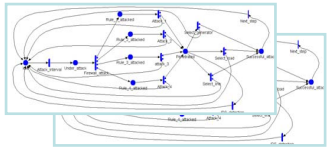


Instrumentation, Control (SCADA)





Adversary model(s) *Abstract*



Attack Models in PSM

- Adversary models used in PSM:
  - Share:
    - The same objectives
      - what assets are being attacked
      - what is the intended harm (e.g. tamper with the configuration of a protection device)
    - The same intensity of attacks
    - The same probability of attack success
    - The same “cleansing” intensity. Cleansing is assumed always successful.
  - Differ
    - In terms of how they capture the *consequences of a successful attack*.
      - *(Option 1) High fidelity model*: Successful attack is modelled in details including the steps taken, effects on SSMs/properties, topology, etc.
      - (*Option 2*) *Abstract model: A* successful attack merely increases (*K-fold*) the rate of failure of the compromised software (device).
        - When the compromised software/device fails, the result is the same as with a successful attack under Option 1.

An attack EXAMPLE:

Tampering with the configuration of a protection device

- *(Option 1) High fidelity model*: Successful attack changed the protection value from default (linked to capacity of power lines/generators/sinks) to 10% above the current flow (i.e. no immediate effect from successful attack)
- (*Option 2*) *Abstract model: A* successful attack increases (*K-fold*) the rate of failure of a protection device. Failure of the protection device leads to disconnecting the protected assets (as it does when the protection device "trips").
  - Using Option 2 leads to an *extra parameter, K.*
  - "Sensitivity analysis": We varied this parameter *K* (K = 1, 10, 100, … 5000…) until the reward E[Pi] of the reward obtained with the Option 1 and Option 2 become similar (very close).

HiFi vs. Abstract Adversary model Results

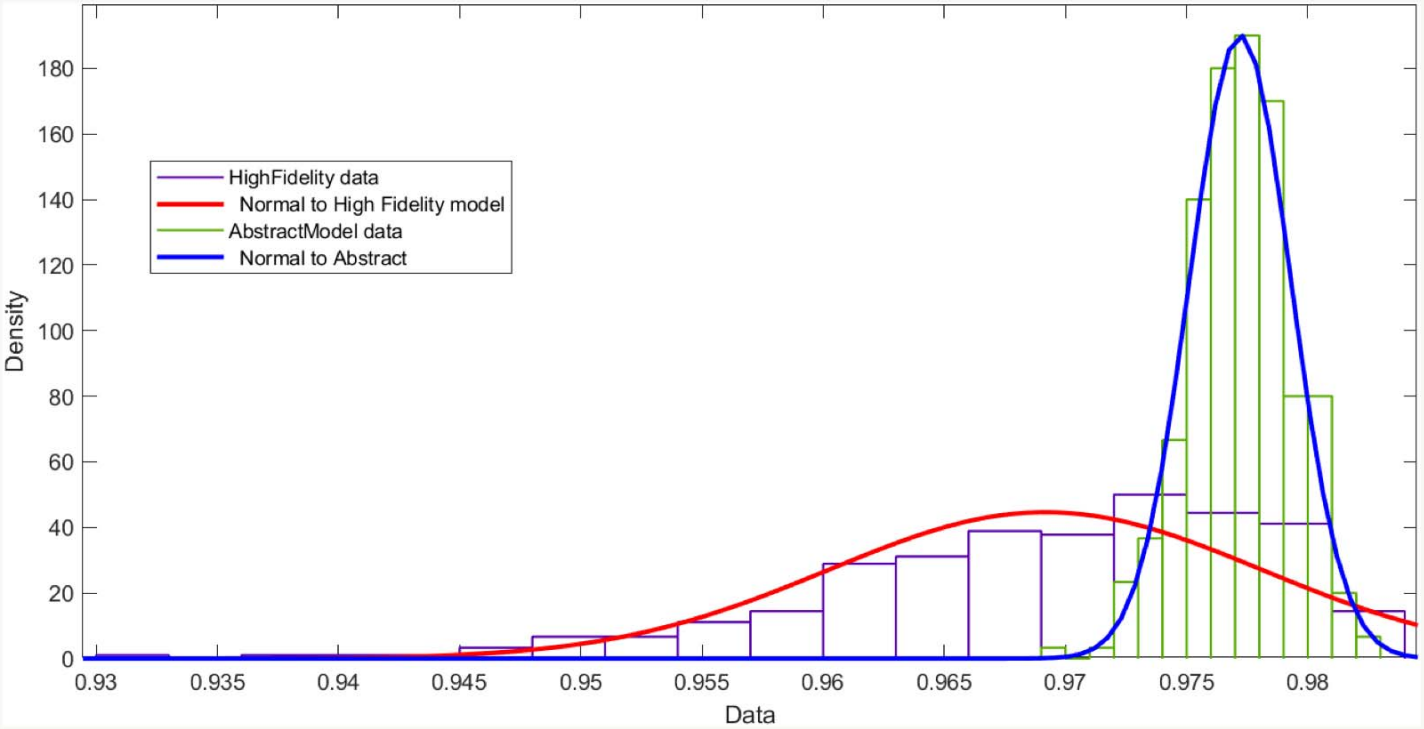
- Parameterisation for the Example attack (Tampering with the configuration of a protection device):
  - Intensity (exponentially distributed with mean “once a week”)
  - Probability of failure = 0.8
  - “Cleansing”: exponentially distributed with mean “once a month”
  - *K* for “abstract model” varied 10, 100, 3650.

Model	E[Pi]	Std. Dev. (Pi)	K
High Fidelity	0.969	0.00893731	N/A
Abstract	0.977	0.00209841	3650

≈ ≠

- The values of E[Pi] with the two models are very close (*difference is < 1%*), although the difference is statistically significant.
- The values of StdDev with both models, however, differ *significantly*.

High Fidelity vs. Abstract model



Fitting a normal distribution to the two samples of 300 observations:

- A (statistically) good fit with the Abstract model
- Poor fit with the High Fidelity model. Testing statistically the hypothesis of normal distribution was rejected.

Clearly the two distributions of Pi are *quite different* despite their means, $E_{abst}[Pi]$ and $E_{HiFi}[Pi]$ being very close.

Discussion

Observation 1

- Looking at the ***expected loss alone***, one can claim that the ***particular*** abstract model is *adequate*. The expected loss can be made as close as needed to the true loss (produced by the HiFi attack model).
  - It seems that this trick is ***always possible*** (one can prove it under broad assumptions)
- This exercise can be repeated with all ***known attacks***. ***K*** is ***likely to vary*** with attack type (to be confirmed)
  - We will identify a range of values of K for these known attacks.

- What K should we use for unknown attacks?
  - Conjecture: Can we use the range established for K on the known attacks?
    - This is a ***bit of a stretch***, but the hypothesis can be checked …
  - *Positives:* In the absence of anything better, the abstract model can be used to get an estimate of the expected loss for the unknown attacks.

Discussion (2)

- *Observation 2*
- Looking at the reward distribution (i.e. of the power loss in the study) and even at the standard deviation, indicates that the *particular* abstract model is NOT very good
  - *Negatives.* This observation is a bad new
  - In *risk assessment* the most interesting part of the distribution would typically be the “tail” of the distribution, which the abstract model seems to *underestimate*.

  - *Positive* (spin) Being optimistic is potentially a useful property of the model!
    - Risk assessment with the particular abstract model can be used to establish a “lower bound” for the risk index: the true value of the risk index will be worse.

Ways forward

- The preliminary results that the particular abstract model of attacks *underestimates* the power loss variability is intriguing.
  - Checking if this property is “universal” (i.e. applies to all known attacks) is clearly worthy.
  - Confirming the property would turn the observations from a minor curiosity to a potentially very useful tool in risk assessment with unknown attacks..

- Look for *better abstract models*, such that lead to greater variability of Pi (e.g. greater standard variation). With such model we may be able to model the risk from unknown attacks more accurately!
  - In fact, the model used here is the *simplest version* of the model developed in ISSRE 2017. Scope for a bit morework here, too.

Questions

- Thank you!

City, University of London
Northampton Square
London
EC1V 0HB
United Kingdom

T: +44 (0)20 7040 8963
E: p.t.popov@city.ac.uk
www.city.ac.uk/people/academics/peter-popov