ATTACK TREES

Olga Gadyatskaya

MSSI: Communication, traitement et persistance des informations 1

23 March 2018

AGENDA

- Context: risk management and threat modelling
- Attack trees
- Quantitative analysis of attack trees
- ADTool
- Gap between theory and practice

SECURITY RISK PARLANCE: RECAP

- `Asset` — smth of value to an organisation.
- `Vulnerability` — a weakness of an `asset` or `control` that can be exploited by a `threat agent`.
- `Threat` — exploitation of a `vulnerability` by a `threat agent` that may lead to an `unwanted incident`.
- `Unwanted incident` — creates damage to an `asset`.
- `Risk` — quantification of a `threat` (probability and impact).
- `Control` — a measure that reduces `risk` .

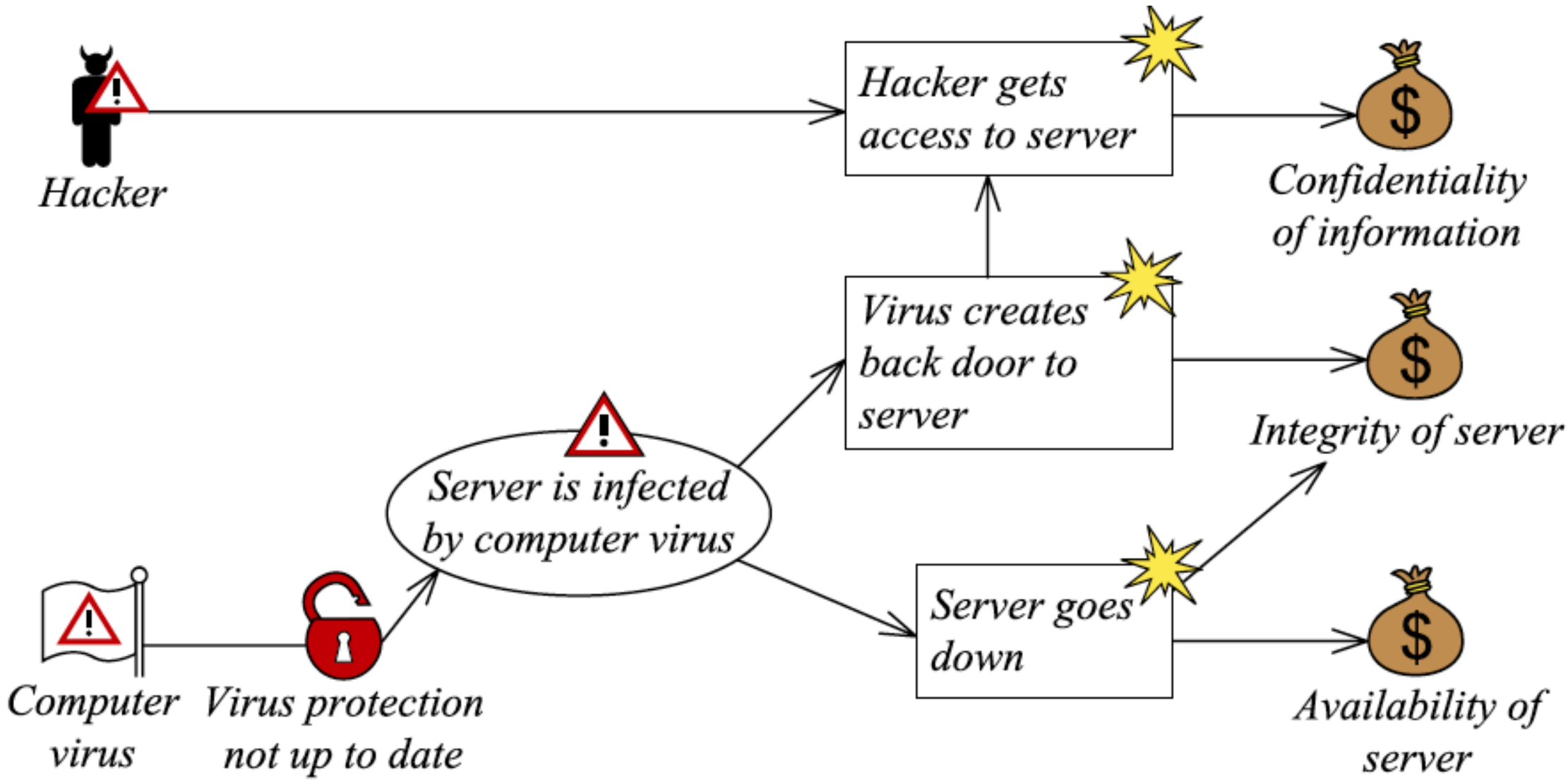
EXAMPLE THREAT SCENARIO

Scenario: Facebook friend Bob discloses your very personal, friends-only post

EXAMPLE THREAT SCENARIO

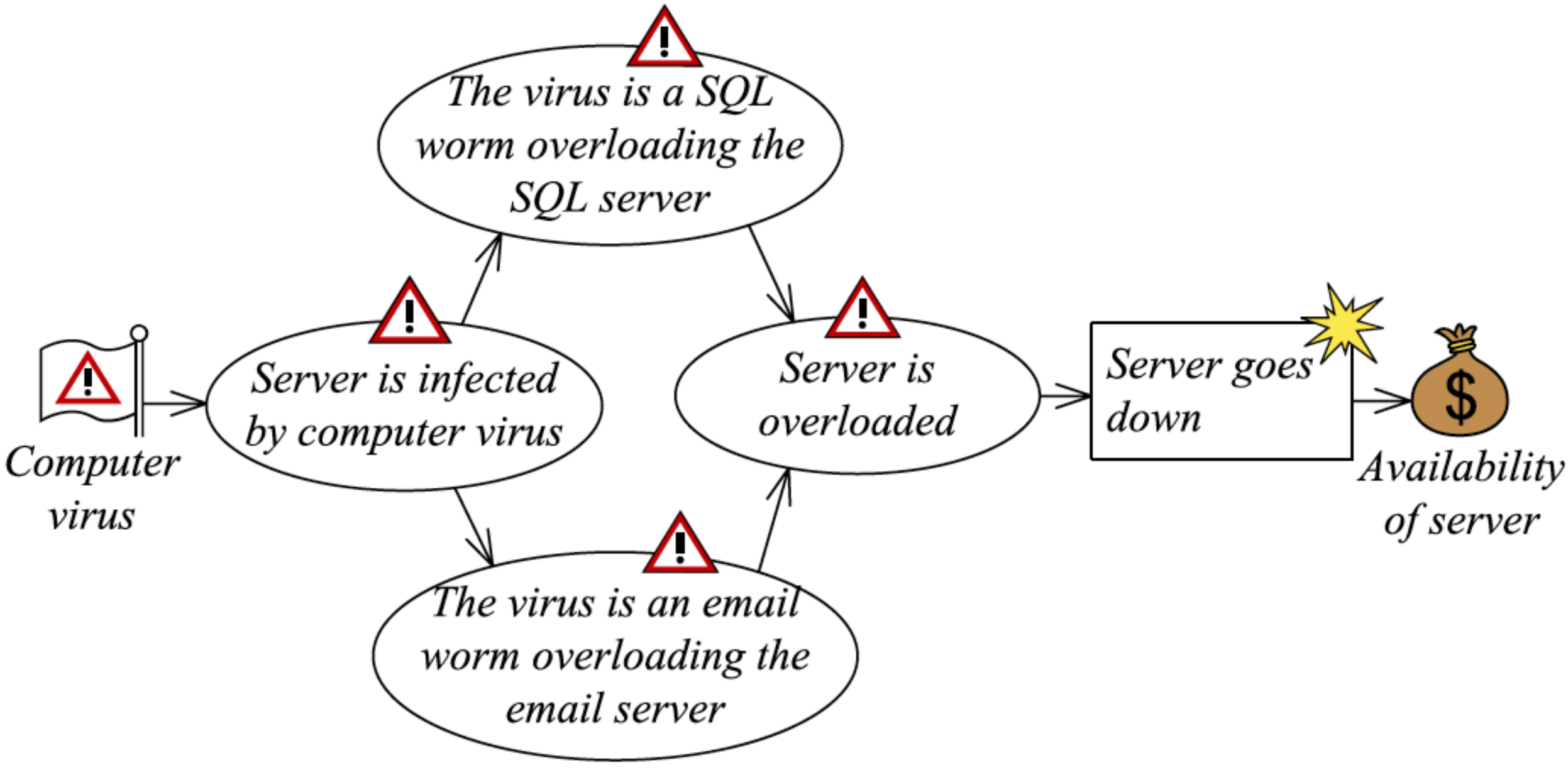
Scenario: friend drank your beer at a party

EXAMPLE OF THREAT SCENARIOS: CORAS DIAGRAMS



http://coras.sourceforge.net/

ANOTHER THREAT DIAGRAM

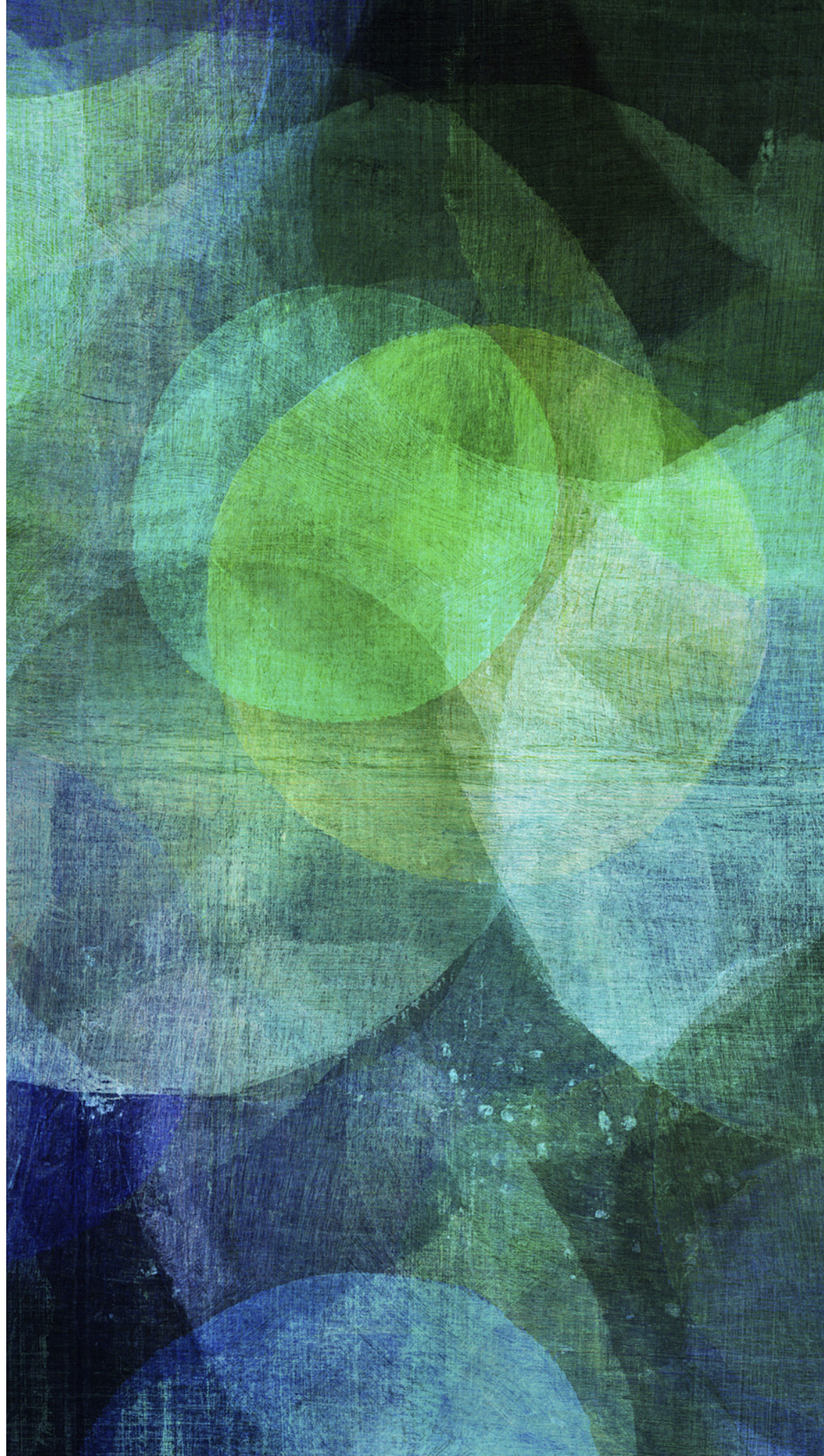


http://coras.sourceforge.net/

THREAT MODELLING: ACTIVITY TO MODEL THREAT SCENARIOS

- Point of view:
  - **System-centric**
    - What are the threat agents?
    - Which vulnerabilities are present?
    - What kinds of threats are relevant?
  - **Attacker-centric**
    - What is the goal?
    - Who is the attacker?
    - What are the attack steps?

ATTACK TREES



GRAPHICAL THREAT MODELLING: ATTACK TREES

- **Goal**: represent a *collection of attacks* in a **tree structure**, with the main attacker's goal as the root node, and different ways of achieving this goal as sub-nodes

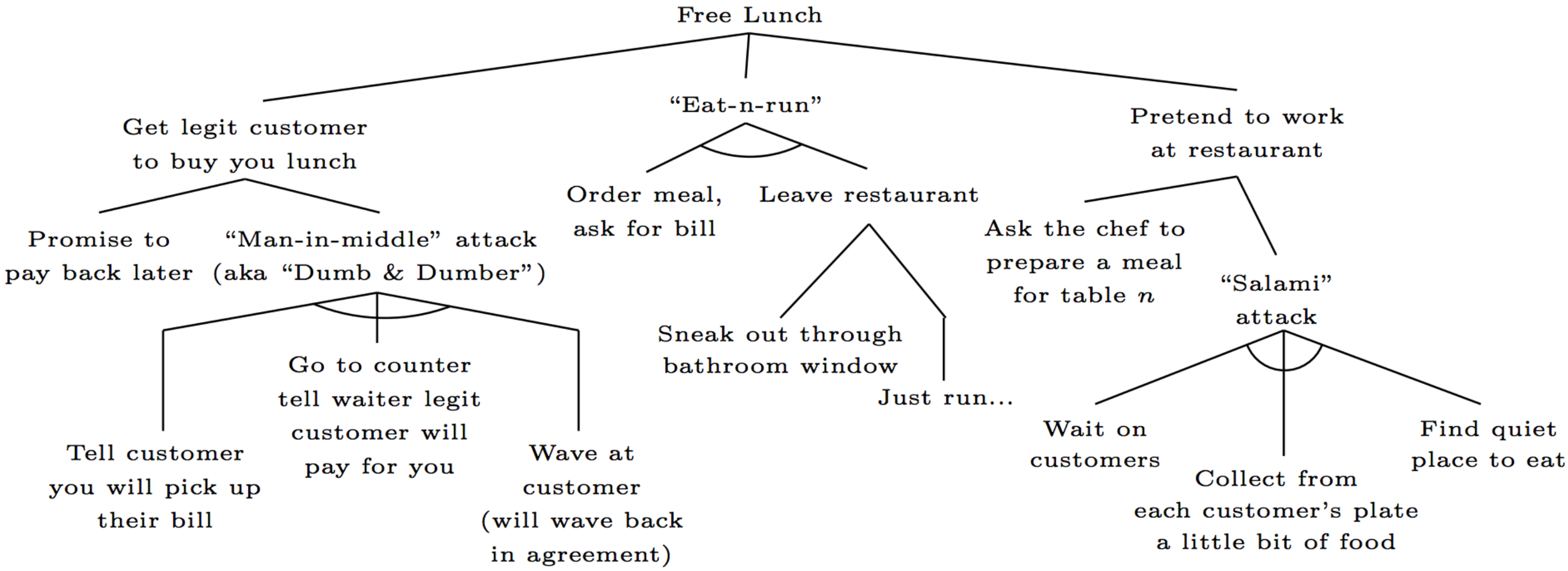
- Originally proposed by Bruce Schneier in "*Attack trees. Modelling security threats*", Dr. Dobb's Journal, 1999.

https://www.schneier.com/academic/archives/1999/12/attack_trees.html

- Formalisation defined by Mauw and Oostdijk in "*Foundations of attack trees*", ICISC'2005

- *Threat trees* are close siblings of attack trees
- *Fault trees* are cousins of attack trees

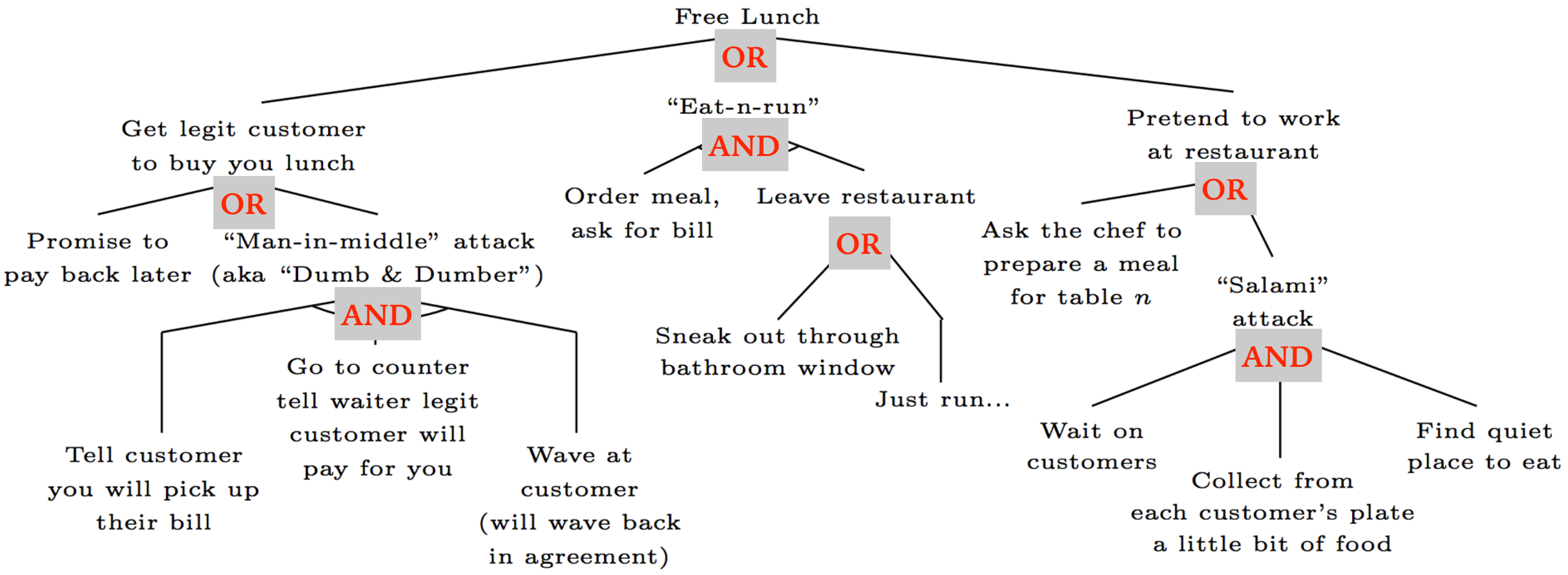
GRAPHICAL THREAT MODELLING: ATTACK TREES II



Mauw and Oostdijk "*Foundations of attack trees*" in ICISC'2005

REFINEMENT

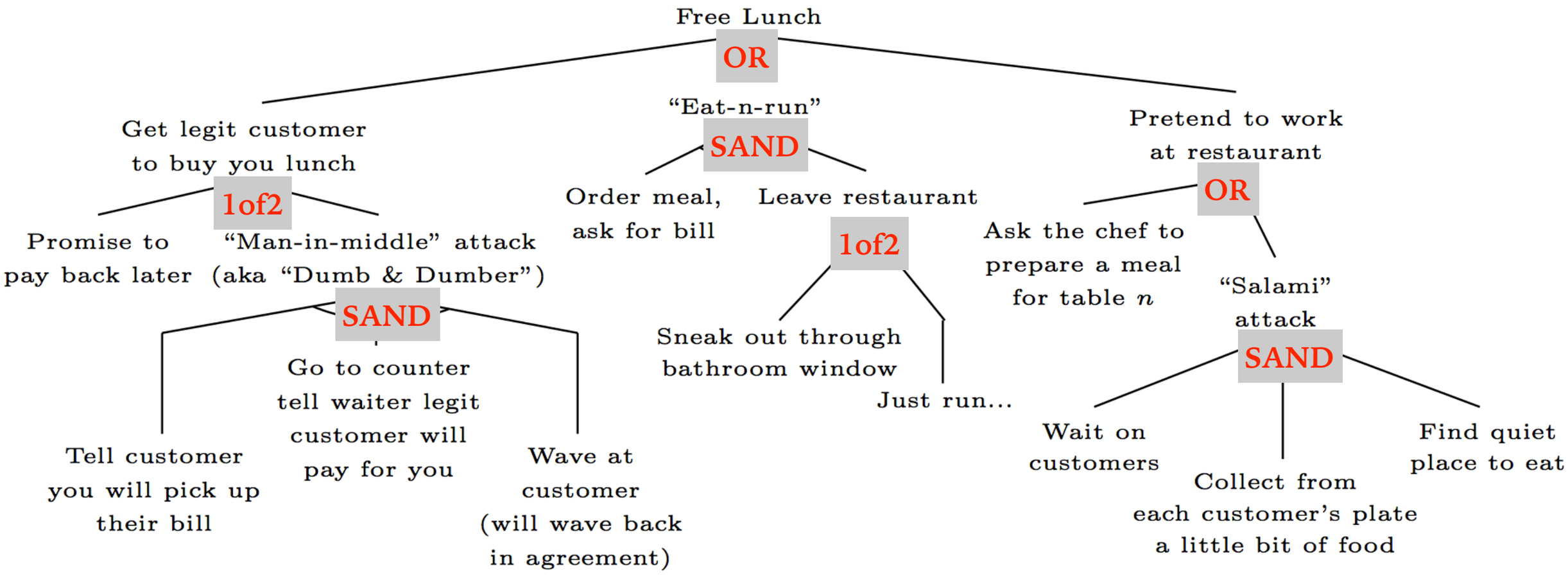
- **Refinement structure**: one of the biggest advantages of attack trees
- Classical refinement operators: AND and OR



Mauw and Oostdijk "*Foundations of attack trees*" in ICISC'2005

REFINEMENT II

- **Refinement structure**: one of the biggest advantages of attack trees
- More refinement operators: `SAND`, `KofN`



Mauw and Oostdijk "*Foundations of attack trees*" in ICISC'2005

BANK ACCOUNT ATTACK EXAMPLE

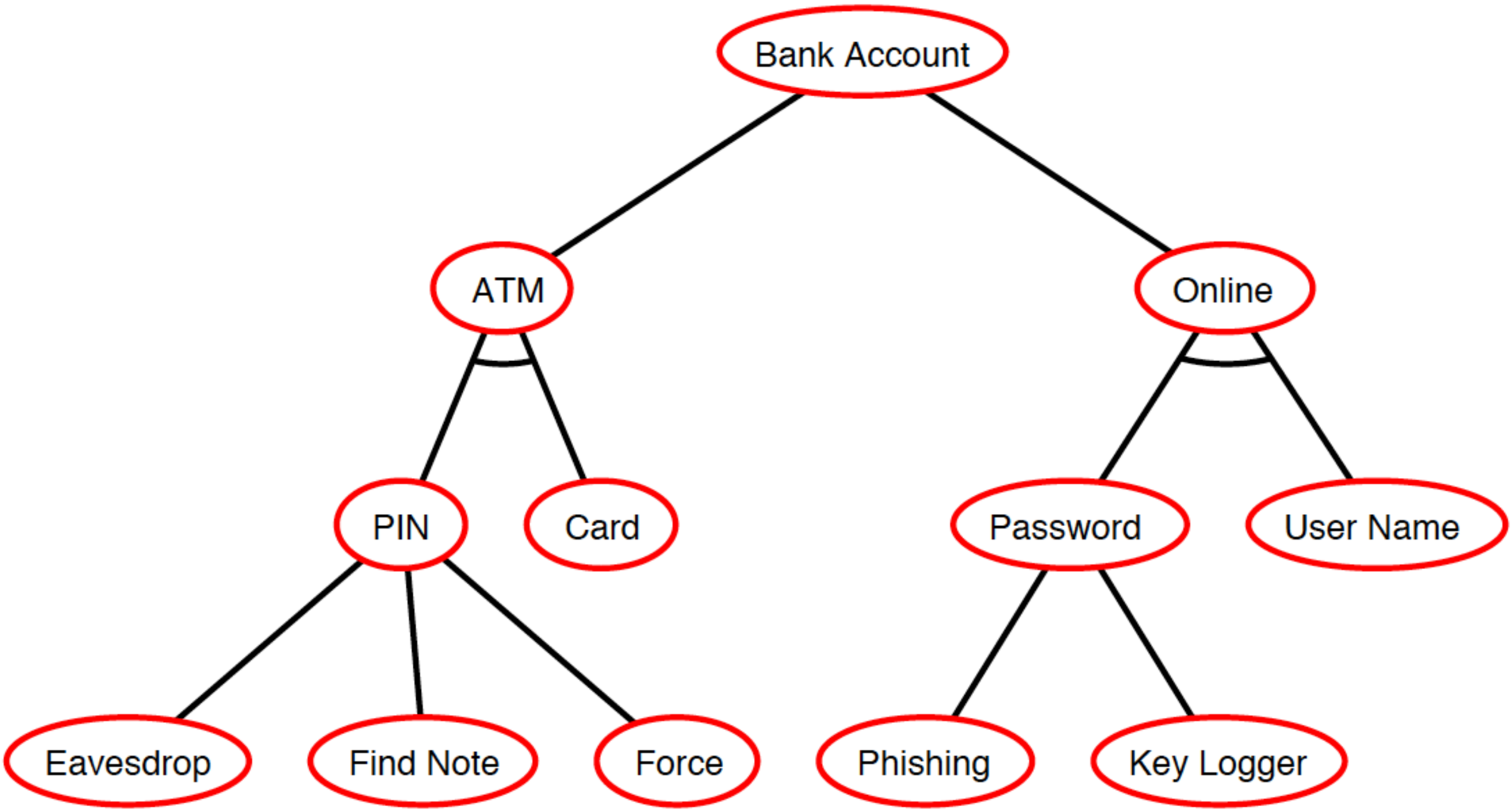


Image courtesy: Sjouke Mauw

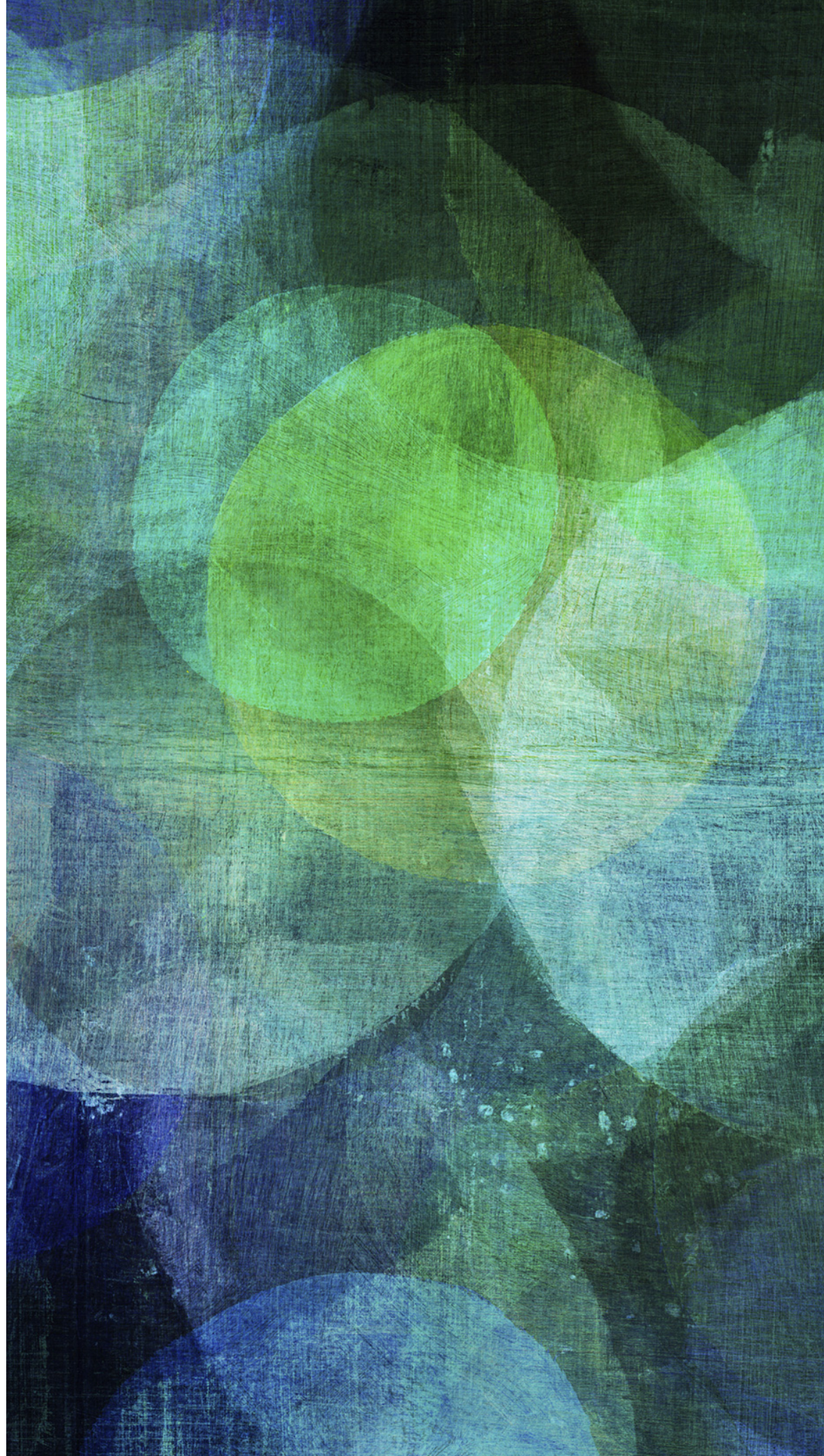
WHY ATTACK TREES: INDUSTRY

- Structured brainstorming means
  - think *mind-maps*
- Facilitate communication across stakeholders
- Allow to reason about quality of the analysis
- Enable *what-if* analysis
  - before and after estimations for scenarios

WHY ATTACK TREES: RESEARCH

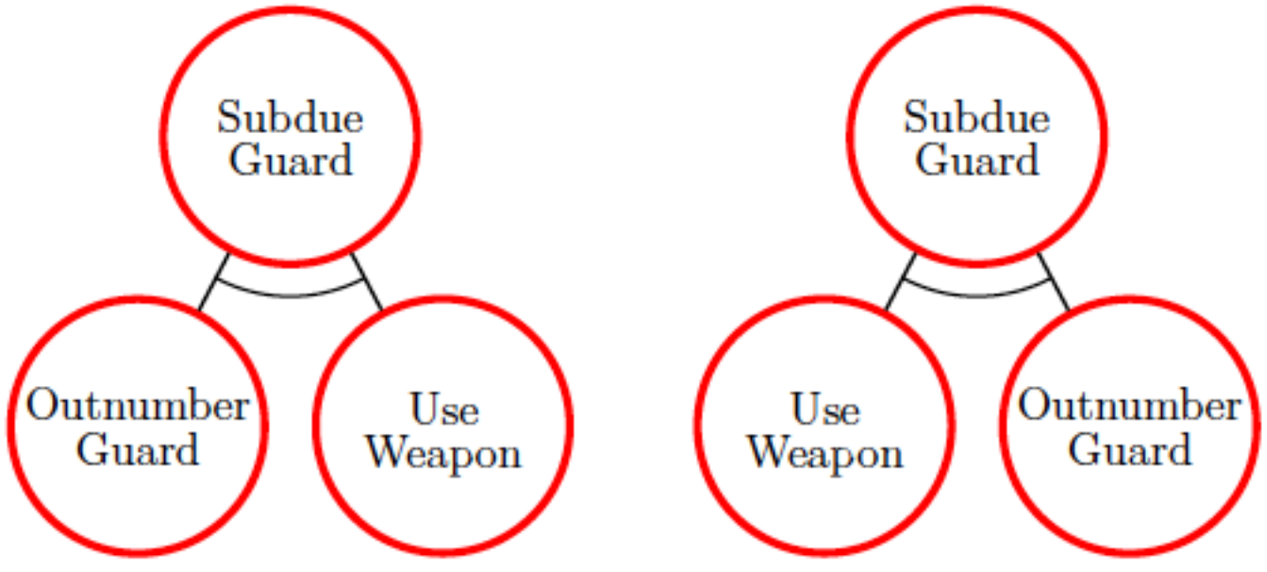
- Allow to develop underlying theoretical models that precisely define meaning (*semantics*)
  - several semantics exist already!
- Semantics enable further studies of the attack tree formalism

SEMANTICS



MEANING OF ATTACK TREES

- How do we know if two attack trees represent the same collection of attacks?



P. Schweitzer "*Attack-defense trees*" PhD thesis, University of Luxembourg, 2013

ARE THESE TWO TREES EQUIVALENT?

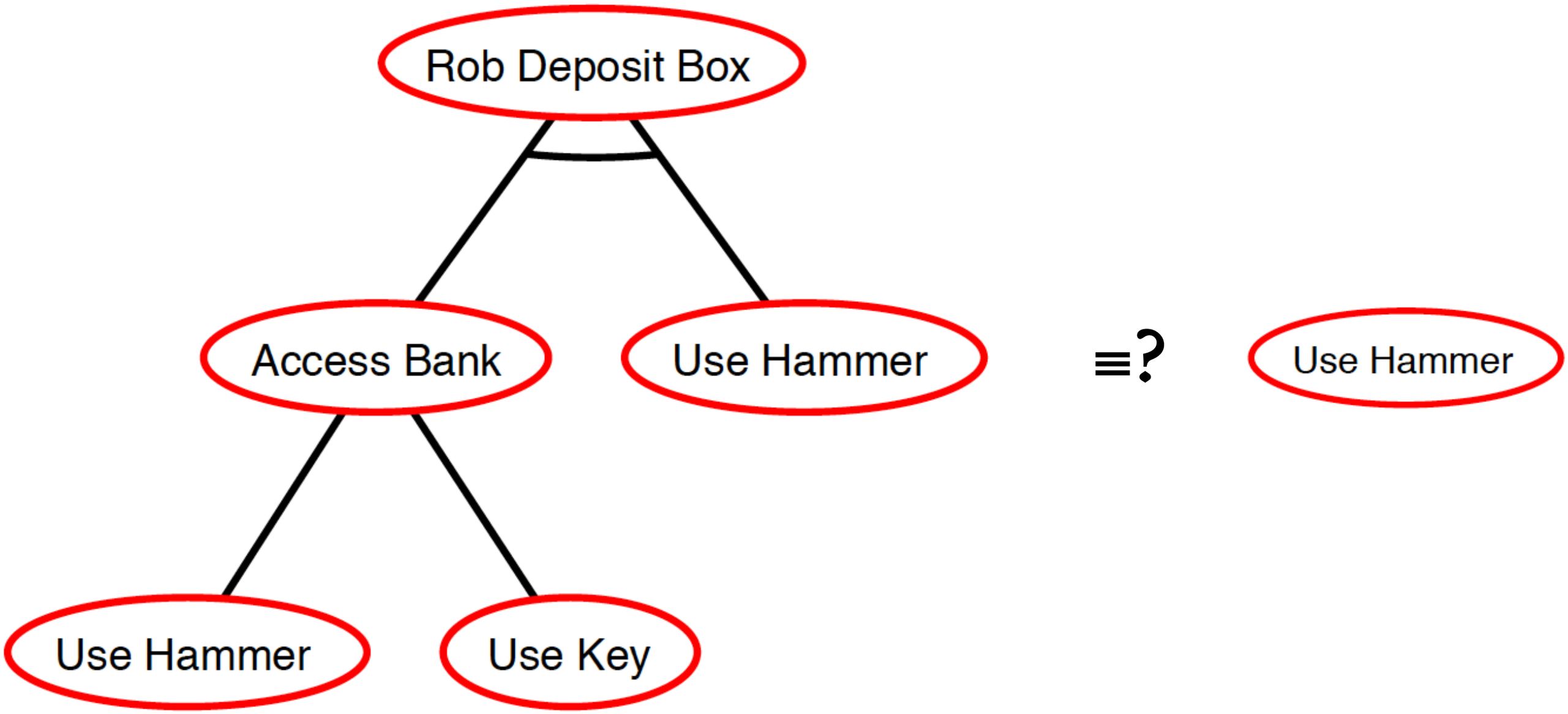


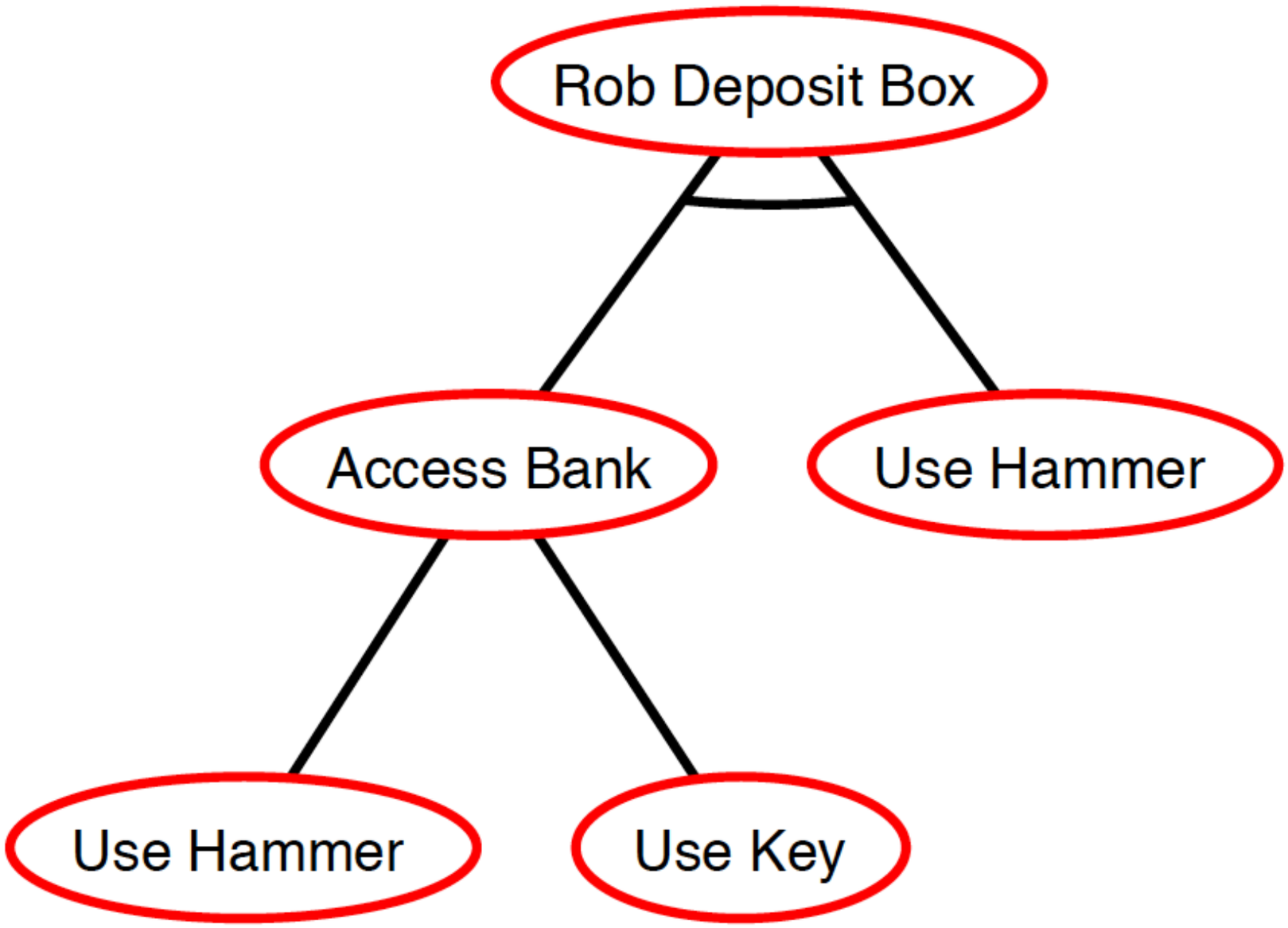
Image courtesy: Sjouke Mauw

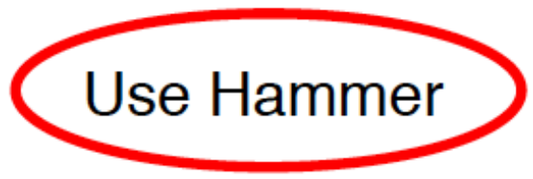
SEMANTICS OF ATTACK TREES II

- Meaning of a tree is typically defined through a combination of its *leaf nodes*

- `Propositional semantics`:
  - an attack tree is defined as a propositional formula
  - two trees are *equivalent* if corresponding propositional formulae are equivalent

PROPOSITIONAL SEMANTICS



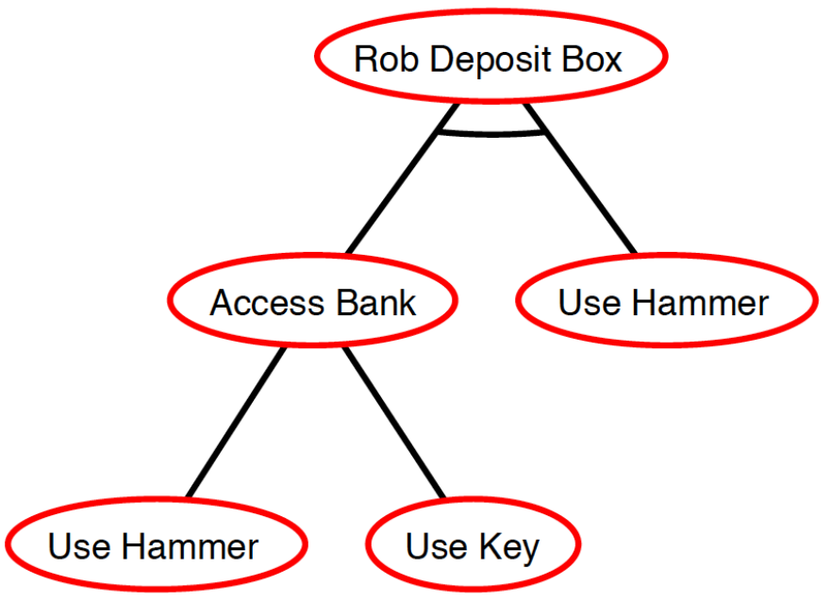


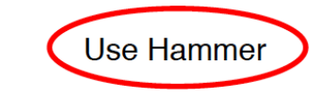
(hammer **OR** key) **AND** hammer $\equiv_P$ hammer

Image courtesy: Sjouke Mauw

MULTISET SEMANTICS

- `Multiset semantics`:
  - an attack tree is a set of multisets. Each multiset is a possible way to attack the system.
  - two attack trees are equivalent if the corresponding sets of multisets are equal.





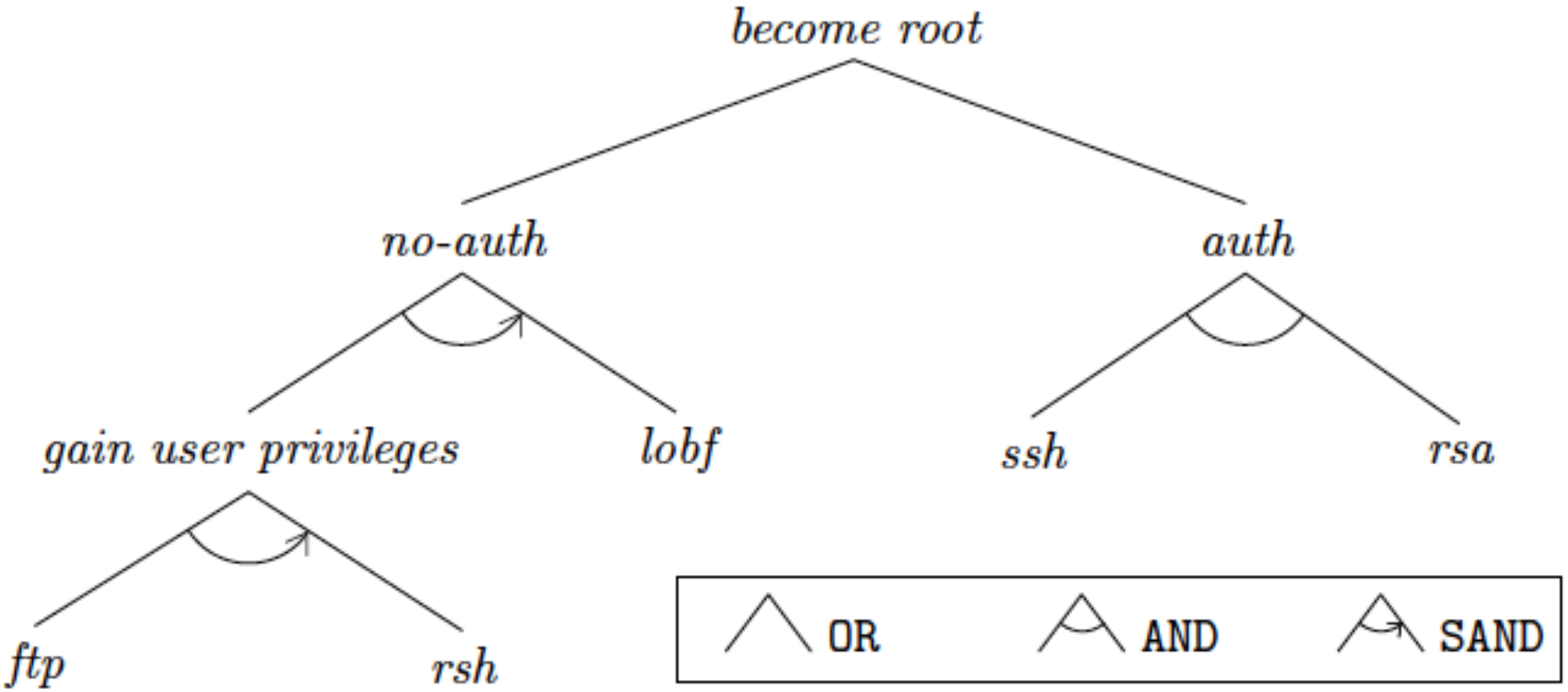
`{|hammer|}`

`{{|hammer, hammer|}, {|key, hammer|}}`

SERIES-PARALLEL GRAPHS SEMANTICS

- For SAND refinement operator we need an order of events
- Captured by series-parallel graphs (SP graphs)
  - SAND: actions are done in sequence
  - AND: actions can be done in parallel
  - OR: any of the actions is done
- Jhawar et al. "*Attack trees with sequential conjunction*" in SEC'2015

EXAMPLE OF SAND TREE



The SP semantics of the attack tree t depicted in Figure 1 is

$$[\![t]\!]_{SP} = \{\xrightarrow{ftp}\xrightarrow{rsh}\xrightarrow{lobf}, \xrightarrow{ssh}\|\xrightarrow{rsa}\}.$$

Jhawar et al. "*Attack trees with sequential conjunction*" in SEC'2015

ADDING CONTROLS TO THE PICTURE

- Attack trees show only attacker's view
- **Attack-defense trees** allow to add also defender's perspective in the same model
  - attack and defence nodes can be interleaved
  - attack tree semantics extended for attack-defence trees
- Kordy et al. "*Foundations of attack-defence trees*" in FAST'2010
- Alternatives: attack-countermeasure trees

BANK ACCOUNT ATTACK WITH COUNTERMEASURES

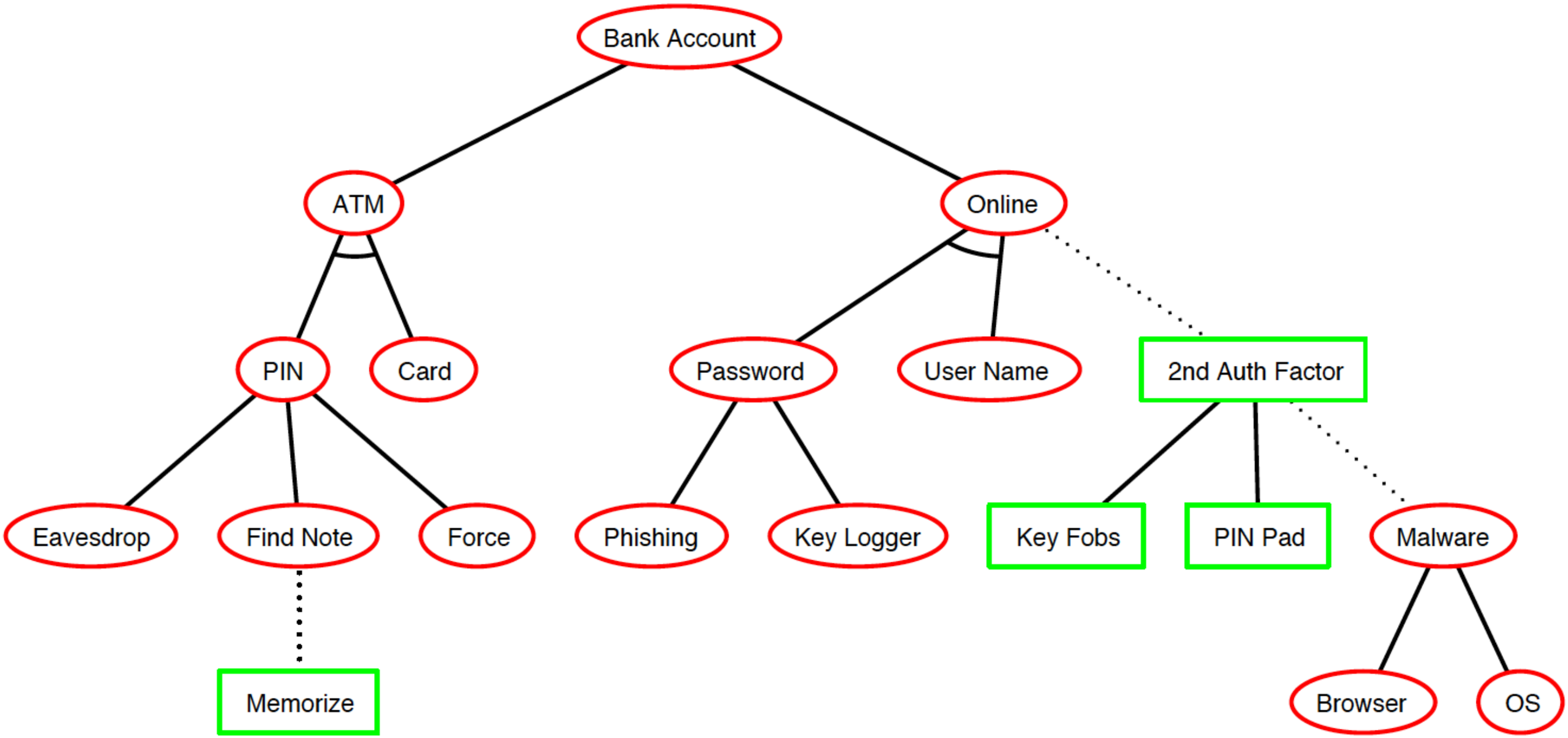
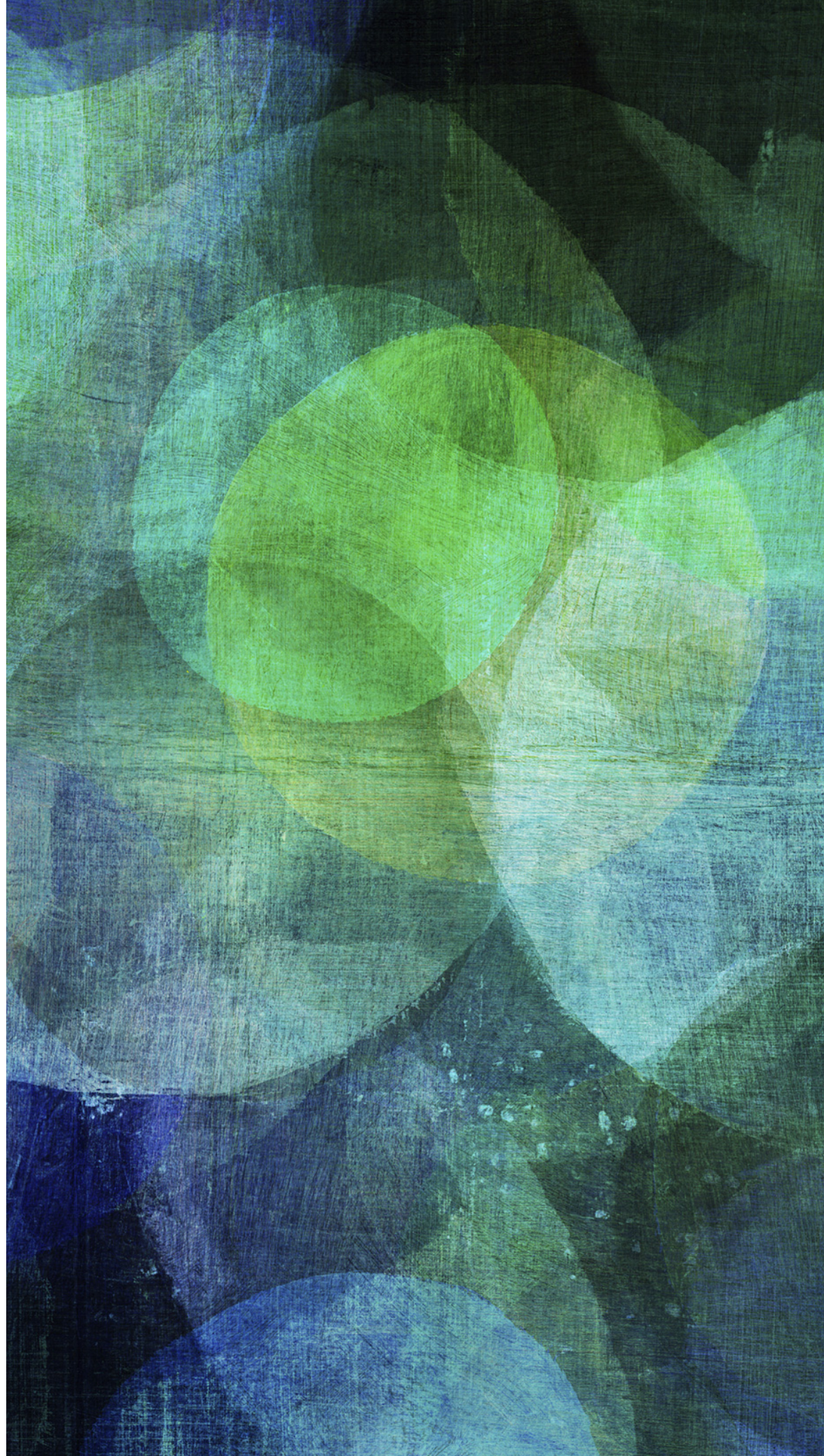


Image courtesy: Sjouke Mauw

QUANTITATIVE ANALYSIS



HOW TO ANALYSE ATTACK TREES?

- Propositional semantics allows to analyse satisfiability of attack scenarios
- What about other properties of attack trees?
- We may want to know:
  - *probability*
  - *cost*
  - *time*
  - …

COMPUTING ATTRIBUTES

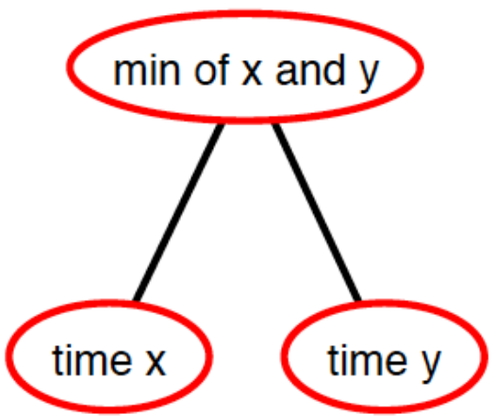
- **Bottom-up algorithm**
  - Values assigned to leaf nodes
  - *Attribute domain* — rules specifying how to compute values for other nodes

- Example: minimal `cost` domain for attack trees
  - `cost` (a **OR** b) = **min**(`cost`(a), `cost`(b))
  - `cost` (a **AND** b) = `cost`(a) + `cost`(b)

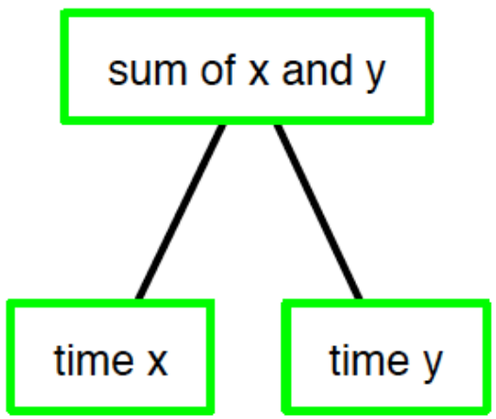
PROBABILITY DOMAIN FOR ATTACK TREES

- $\mathtt{Pr}(a \textbf{ OR } b) = 1 - (1-\mathtt{Pr}(a))(1-\mathtt{Pr}(b)) = \mathtt{Pr}(a) + \mathtt{Pr}(b) - \mathtt{Pr}(a)\mathtt{Pr}(b)$
- $\mathtt{Pr}(a \textbf{ AND } b) = \mathtt{Pr}(a)\mathtt{Pr}(b)$

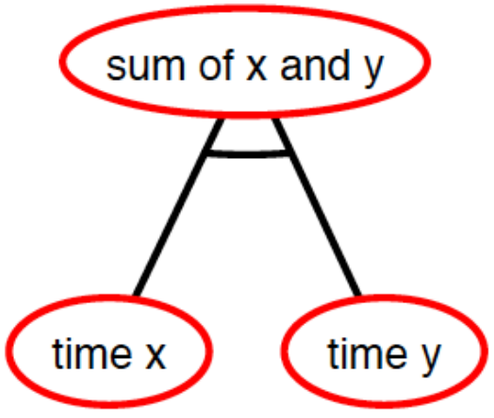
MINIMAL ATTACK TIME DOMAIN FOR ATTACK-DEFENCE TREES



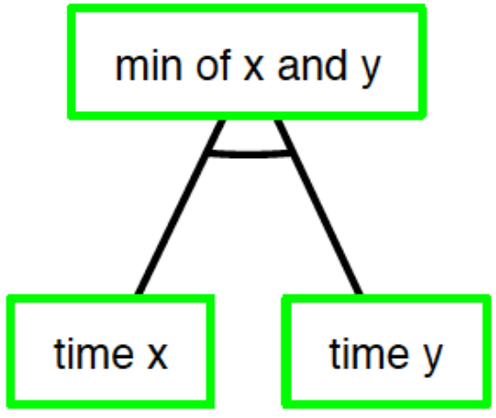
$\vee^A$: $\min\{x, y\}$



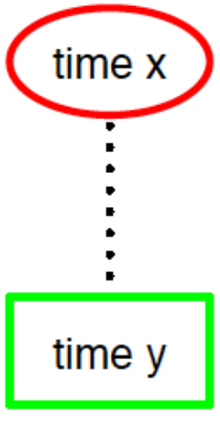
$\vee^D$: $x + y$



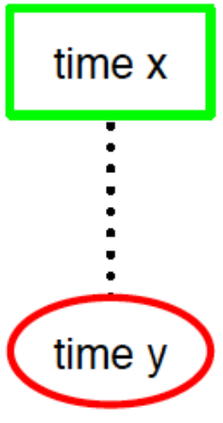
$\wedge^A$: $x + y$



$\wedge^D$: $\min\{x, y\}$



c^A: $x + y$



c^D: $\min\{x, y\}$

Image courtesy: Sjouke Mauw

MIN TIME FOR THE BANK ACCOUNT ATTACK EXAMPLE

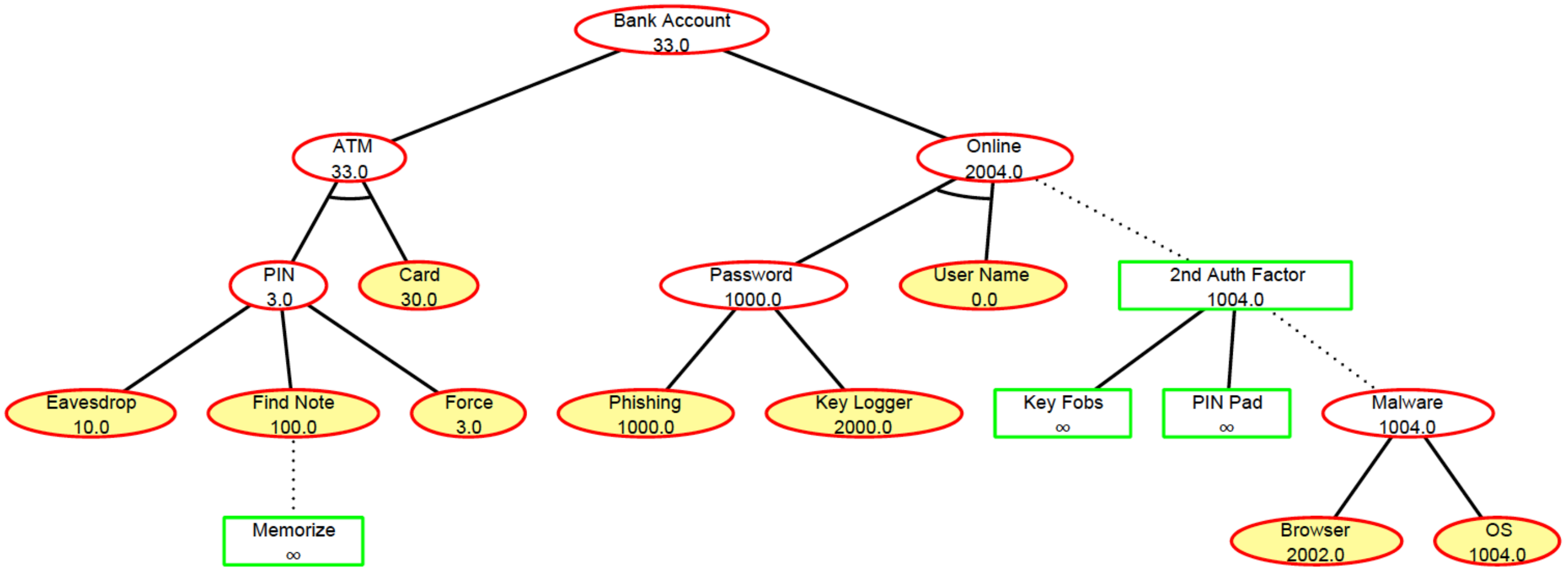


Image courtesy: Sjouke Mauw

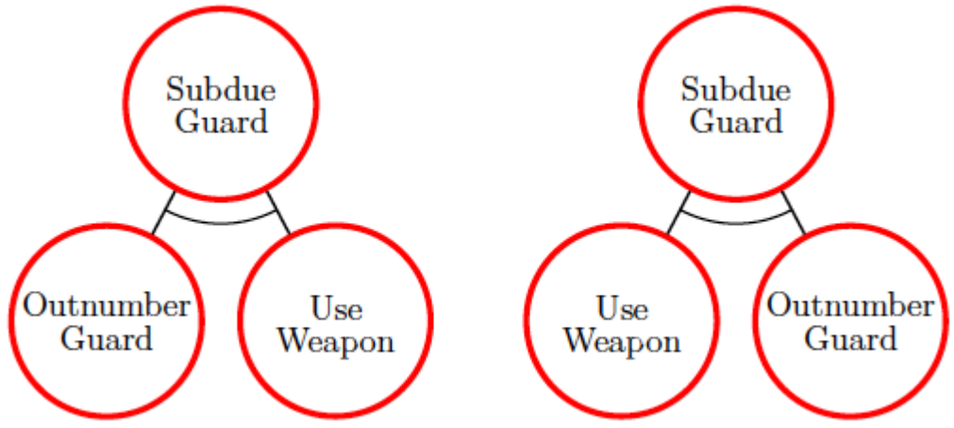
ADTOOL

- Open source Java software to work with attack trees
- http://satoss.uni.lu/members/piotr/adtool/
- Supports:
  - attack-defence trees and SAND-trees
  - quantitative analysis with many attributes
  - ranking of attacks

ADTOOL LIVE

COMPATIBILITY OF SEMANTICS AND ATTRIBUTE DOMAINS

- Semantics defines equivalence relation on attack trees
- Intuition: same trees should yield the same value



- Attribute domain *D* is **compatible** with semantics *S* if all trees equivalent in *S* result in the same value for *D*
- Kordy et al. "*Attack-defence trees*" in Oxford Journal of Logic 2014

COMPATIBILITY EXAMPLE

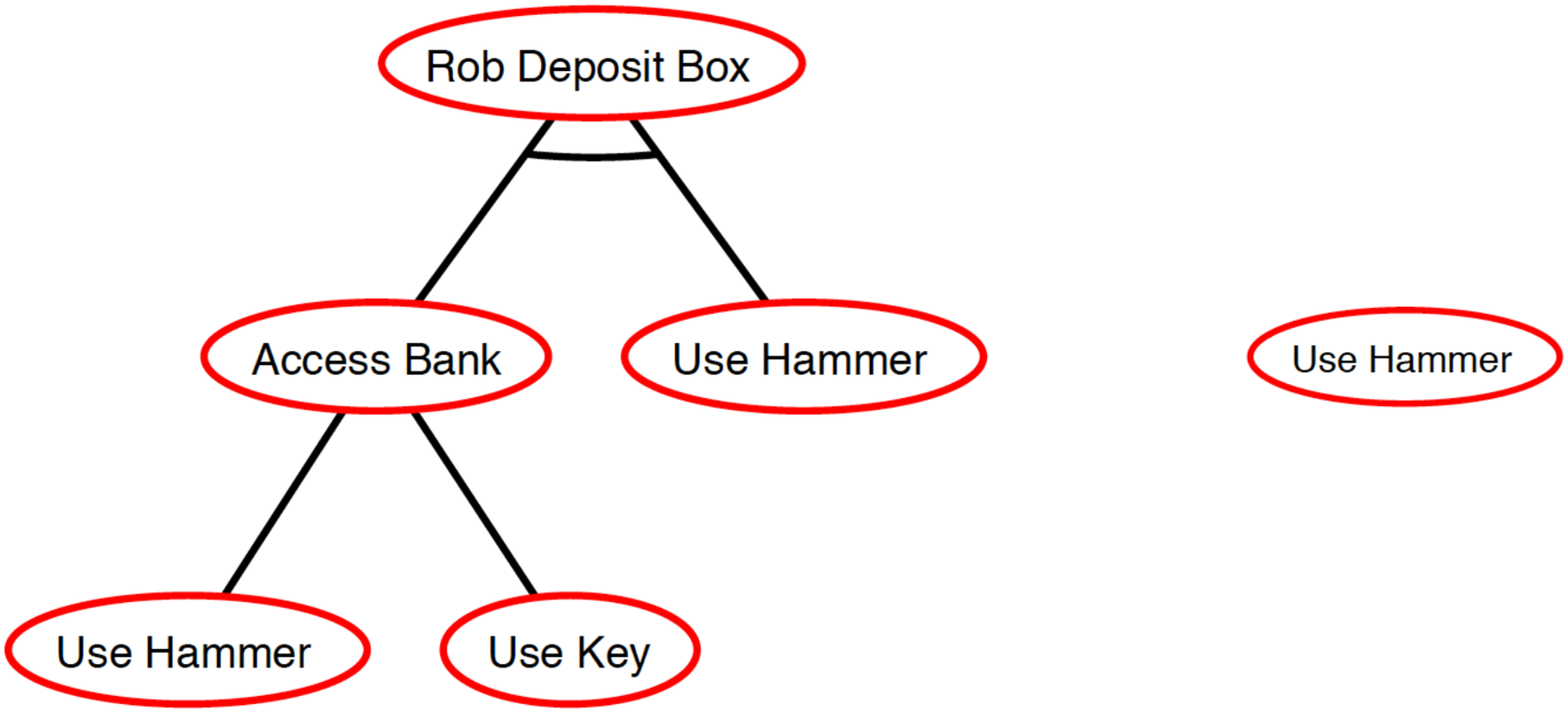
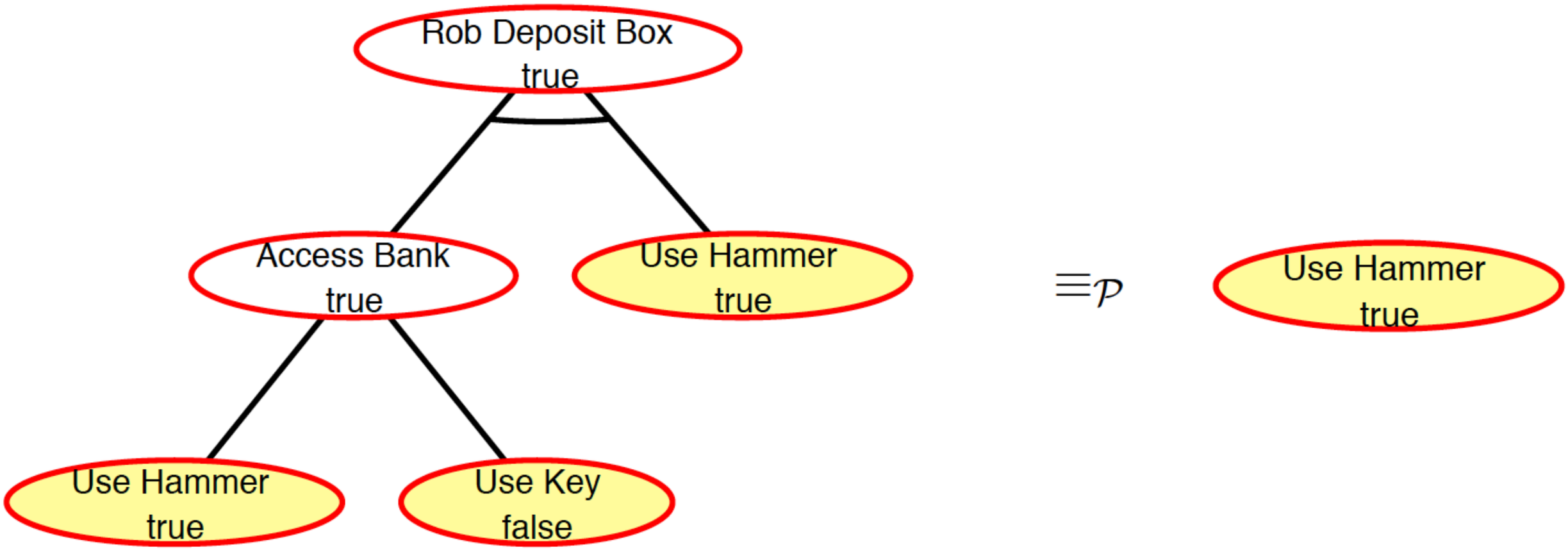


Image courtesy: Sjouke Mauw

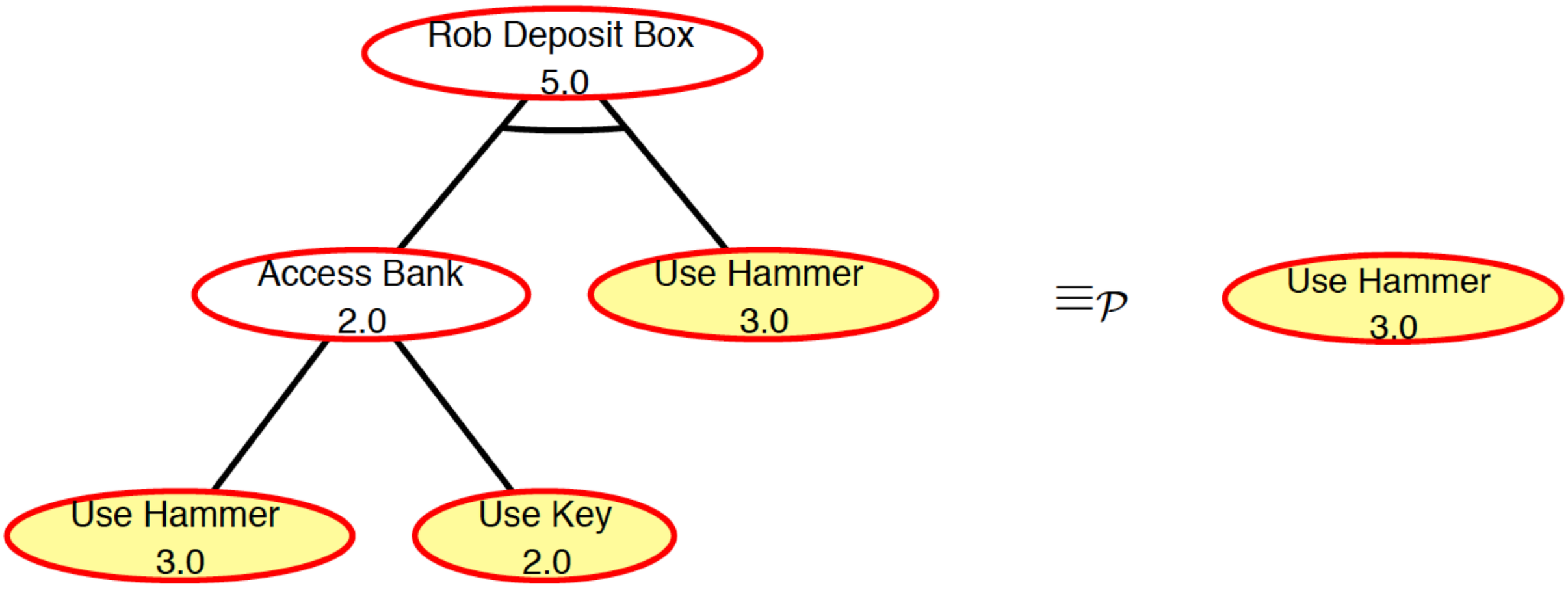
COMPATIBILITY EXAMPLE II



- Propositional semantics is compatible with the satisfiability attribute domain

Image courtesy: Sjouke Mauw

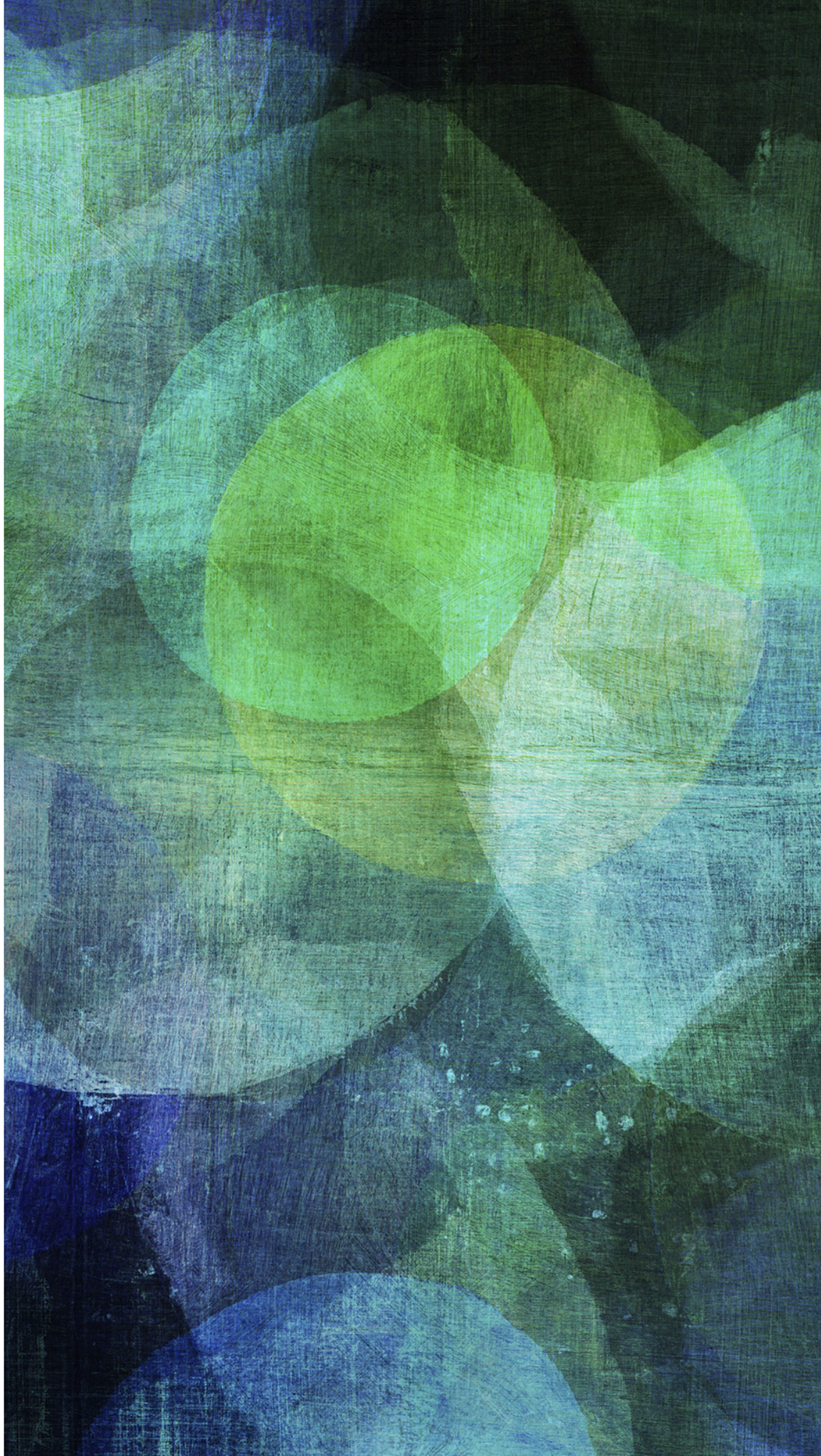
COMPATIBILITY EXAMPLE III



- Propositional semantics is not compatible with the minimal cost domain

Image courtesy: Sjouke Mauw

BRIDGING THE GAP BETWEEN THEORY AND PRACTICE



ISSUES WITH ATTACK TREES

- Huge effort
- Completeness
- Data quality

Image courtesy: causecast.com

REDUCE EFFORT

- **Generate trees automatically from system models**
  - Gadyatskaya et al. "*Refinement-Aware Generation of Attack Trees*" in STM'2017
- **Generate trees from libraries of attacks**
  - Paul "*Towards Automating the Construction & Maintenance of Attack Trees: a Feasibility Study*" in GraMSec'2014

COMPLETENESS

- **Can be formally ensured for generated trees**

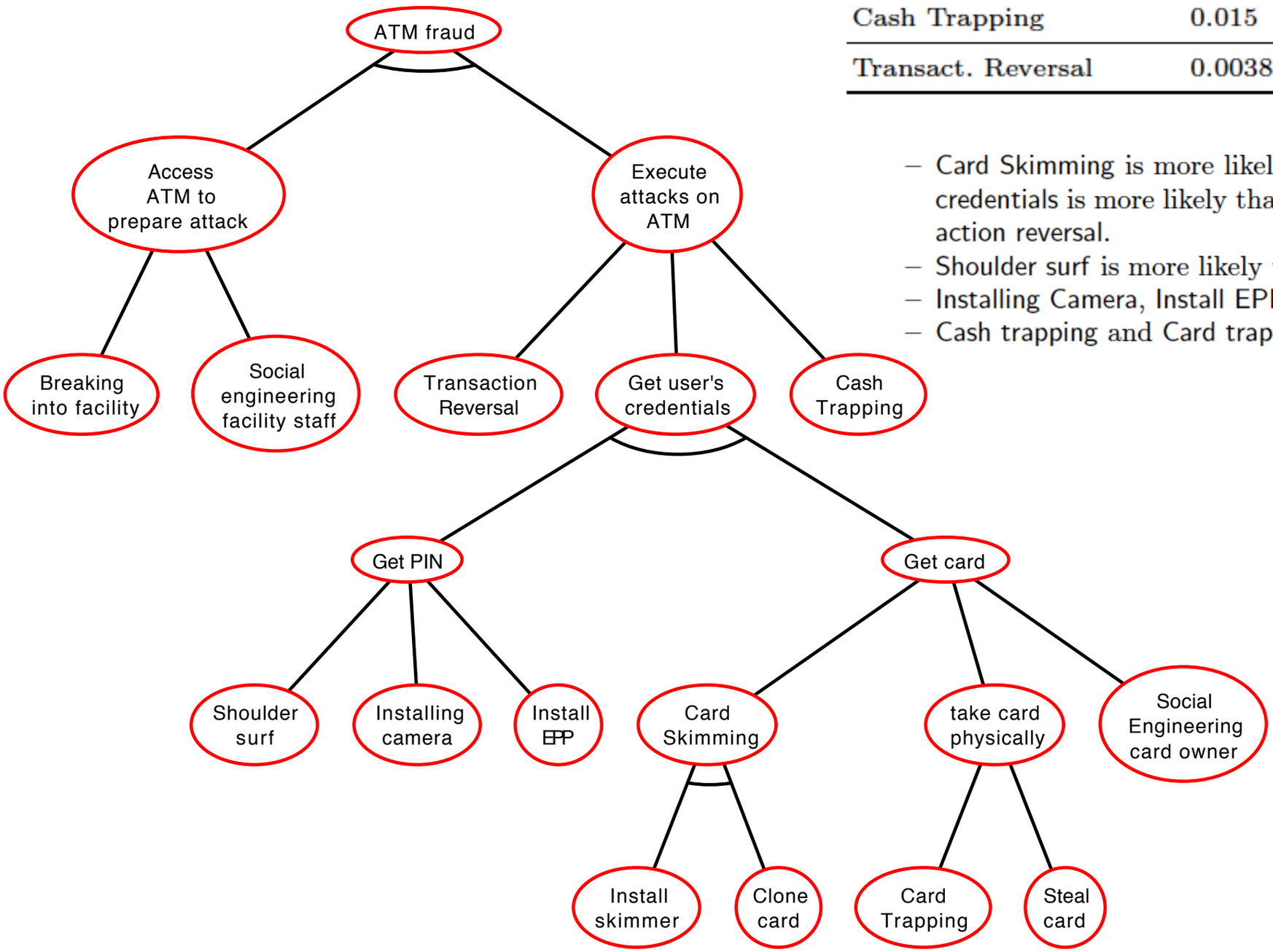
- **Rely on industry catalogues of threats**
  - Fraile et al. "*Using attack-defence trees to analyse threats and countermeasures in an ATM: A case study*" in PoEM'2016

DATA QUALITY

- Quantitative analysis is as good as the data used
  - but data for low-level actions are not available

- Solution: use available statistical data values
  - requires to change the bottom-up approach

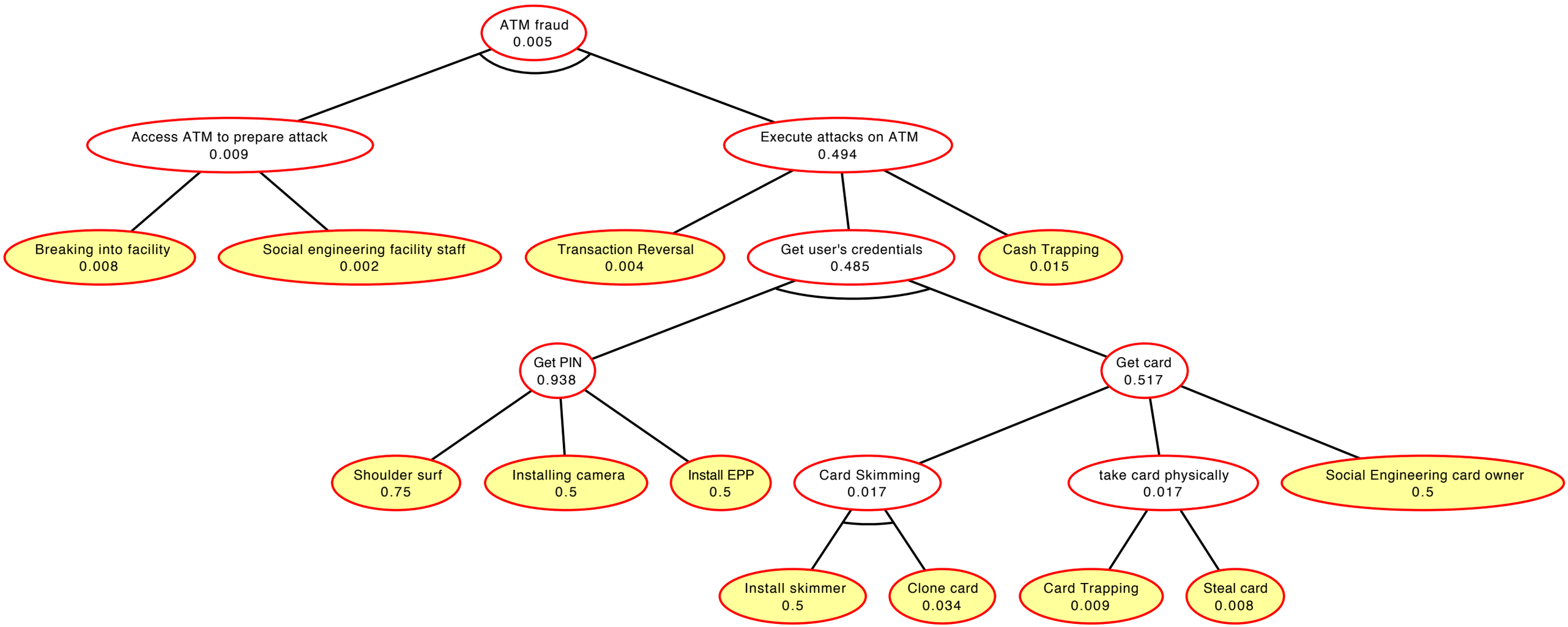
EXAMPLE OF CONSISTENT DECORATION FROM HISTORICAL DATA



Node	Prob.	Source
ATM fraud	0.0046	ATM Crime Report 2015 (EAST)
Card Skimming	0.0172	ATM Crime Report 2015 (EAST)
Card Trapping	0.0094	ATM Crime Report 2015 (EAST)
Cash Trapping	0.015	ATM Crime Report 2015 (EAST)
Transact. Reversal	0.0038	ATM Crime Report 2015 (EAST)

- Card Skimming is more likely than take card physically. Moreover, Get user's credentials is more likely than Cash trapping which is more likely than Transaction reversal.
- Shoulder surf is more likely than Installing Camera
- Installing Camera, Install EPP, and Install Skimmer, are all equally likely.
- Cash trapping and Card trapping are equally likely.

DECORATED TREE



APPLICATIONS OF ATTACK TREES IN LUXEMBOURG

- **Cost-benefit analysis** is used to select cost-effective countermeasures
  - controls that optimally reduce risks
- Gadyatskaya et al. *"Bridging two worlds: Reconciling practical risk assessment methodologies with theory of attack trees"* in GraMSec'2016
  - integrated attack trees with the TRICK Service
  - https://www.itrust.lu/trick-service/
    - expert designs an attack tree
    - controls are selected from catalogue and inserted into attack tree

EXAMPLE: ORIGINAL ATTACK TREE

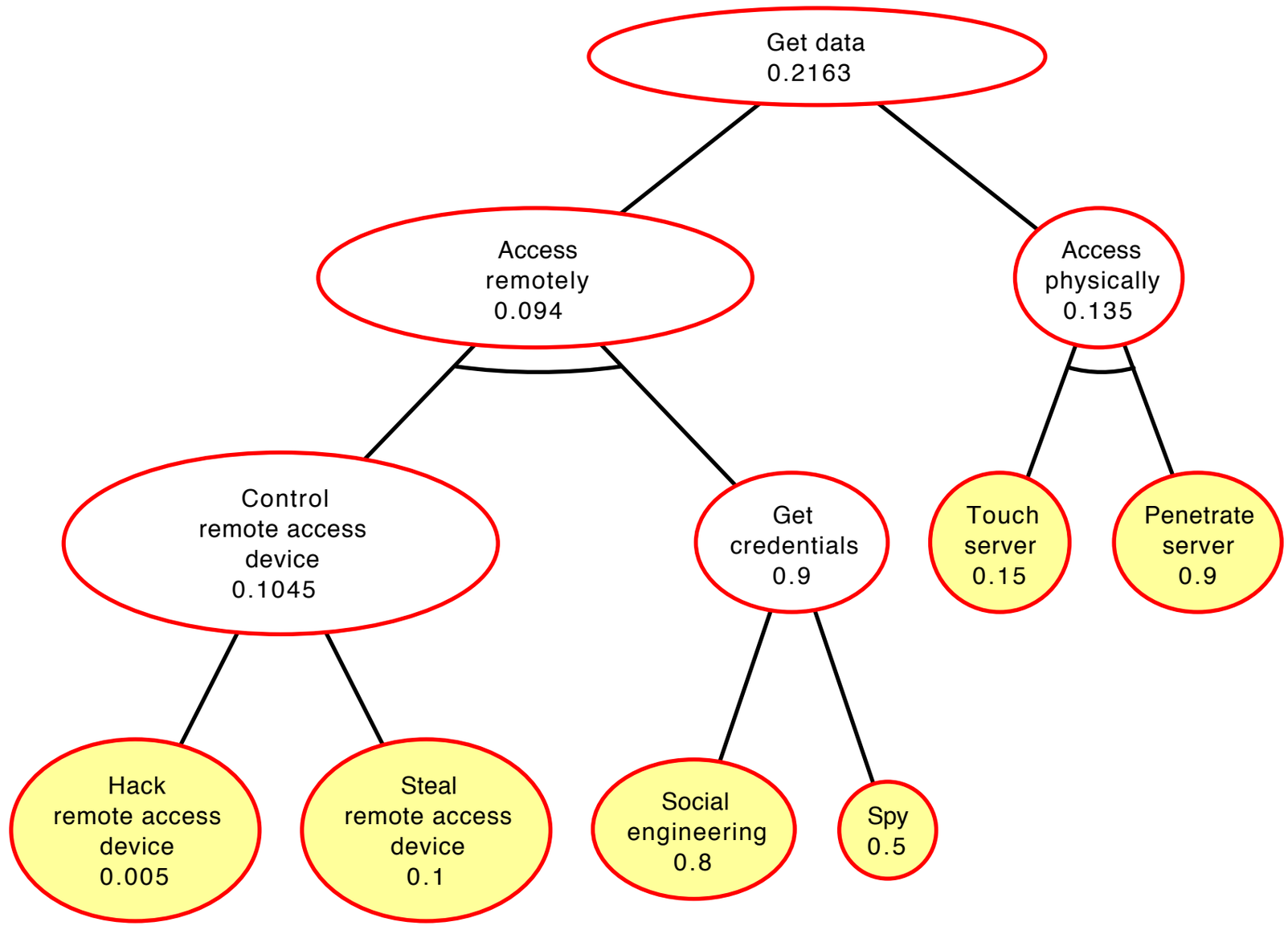


Image courtesy: Cedrick Muller

EXAMPLE: ATTACK TREE WITH OPTIMAL COUNTERMEASURES

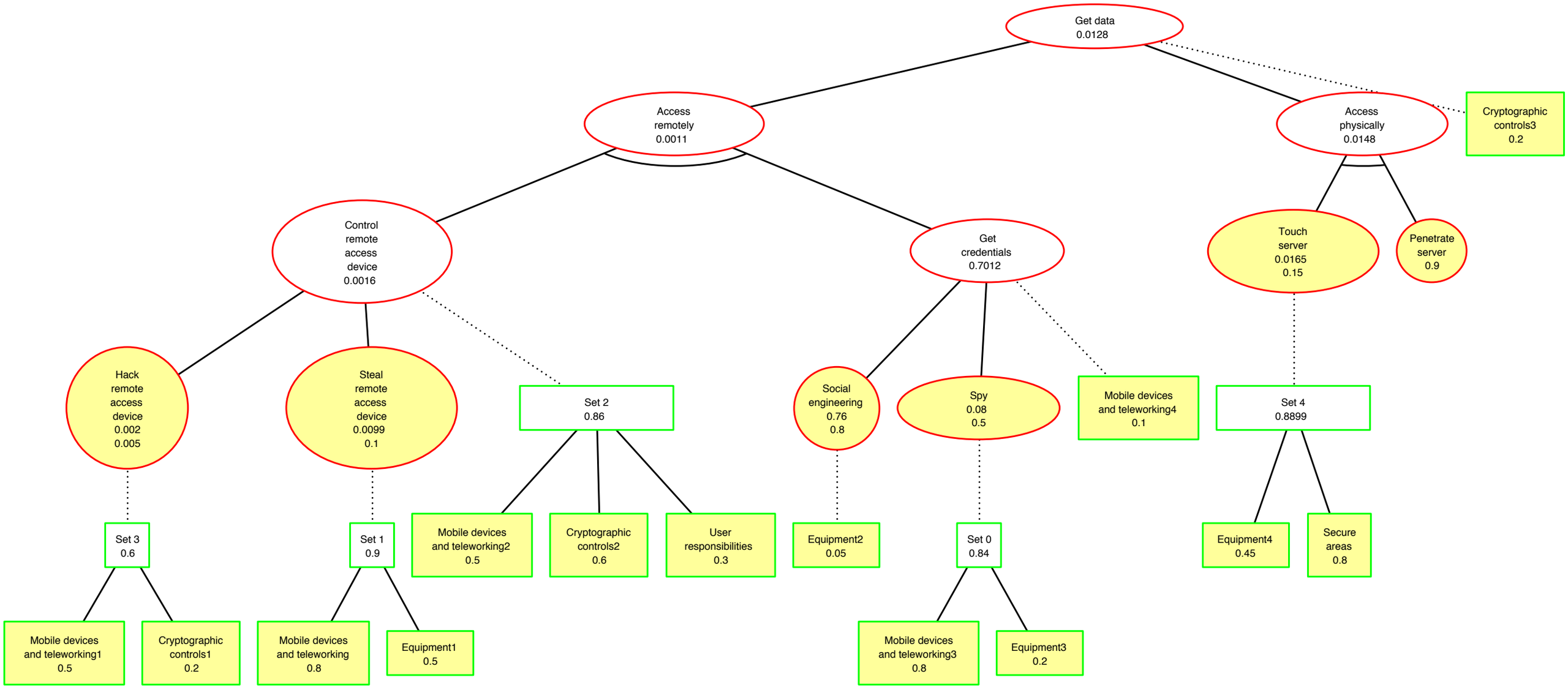


Image courtesy: Cedrick Muller

OPEN CHALLENGES

- Best practices for attack trees
  - *cognitive complexity* versus the *formalism power*

- Methodology for automated attack trees generation
  - integrated with risk management standards

- Theory of attack-defence trees with different controls
  - *cost-effective* countermeasure selection in this theory

THE END

Contact me at

olga.gadyatskaya@uni.lu

Master thesis projects available