Attack trees and security assessment

Sjouke Mauw

(Thanks to Barbara Kordy for the slides)

ADaCoR Industry Workshop, April 19-21, 2016

Outline

1. Attack trees
2. Quantitative analysis
3. ADTool
4. Semantics for ADTrees
5. Concluding remarks

Outline

Attack trees

Definition

Attack tree – tree-like representation of an attacker's goal recursively refined into conjunctive or disjunctive sub-goals.

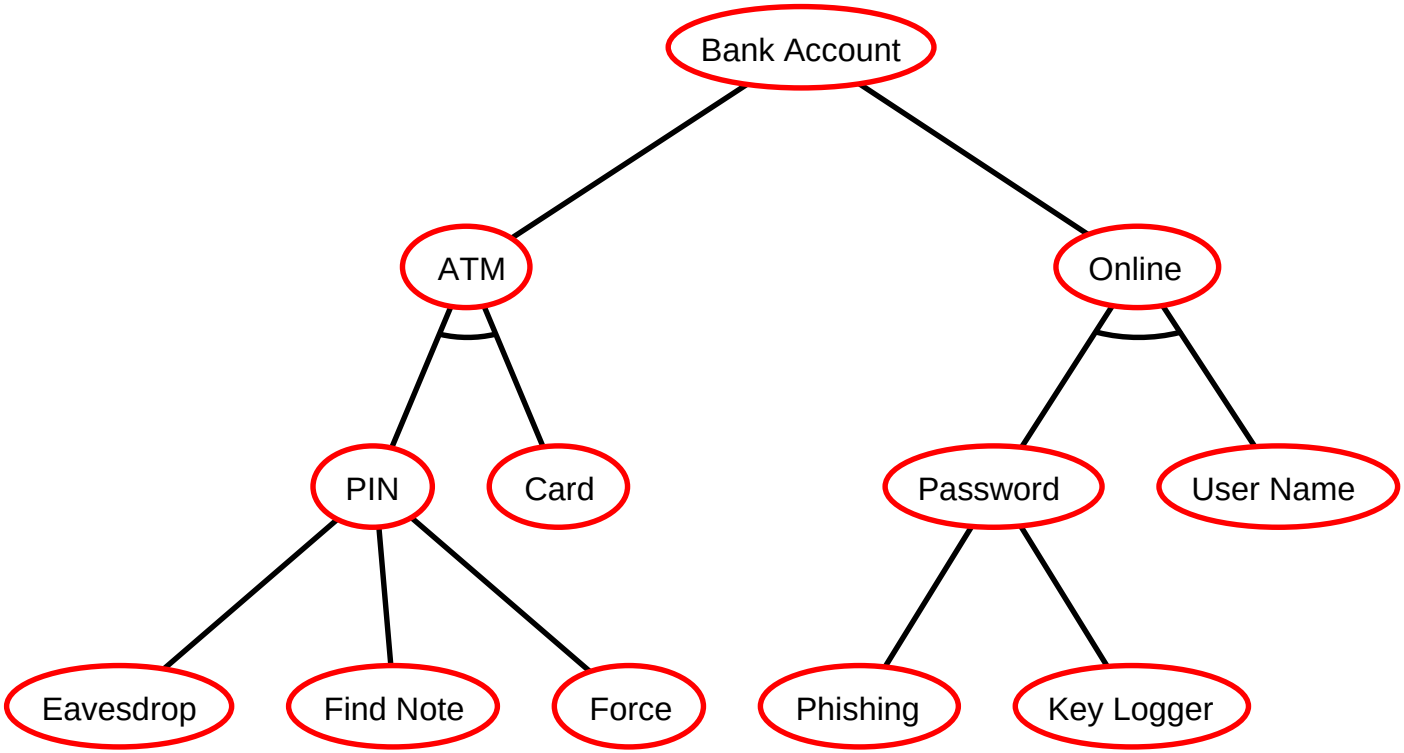
Methodology to describe security weaknesses of a system

- Proposed by Schneier

  Attack trees: Modeling Security Threats, '99
- Formalized by Mauw and Oostdijk

  Foundations of Attack Trees [ICISC'05]

Example: attacking a bank account



Limitations of attack trees

- Only attacker's point of view
- No defensive measures
- No attacker/defender interactions
- No evolutionary aspects

Attack–defense trees

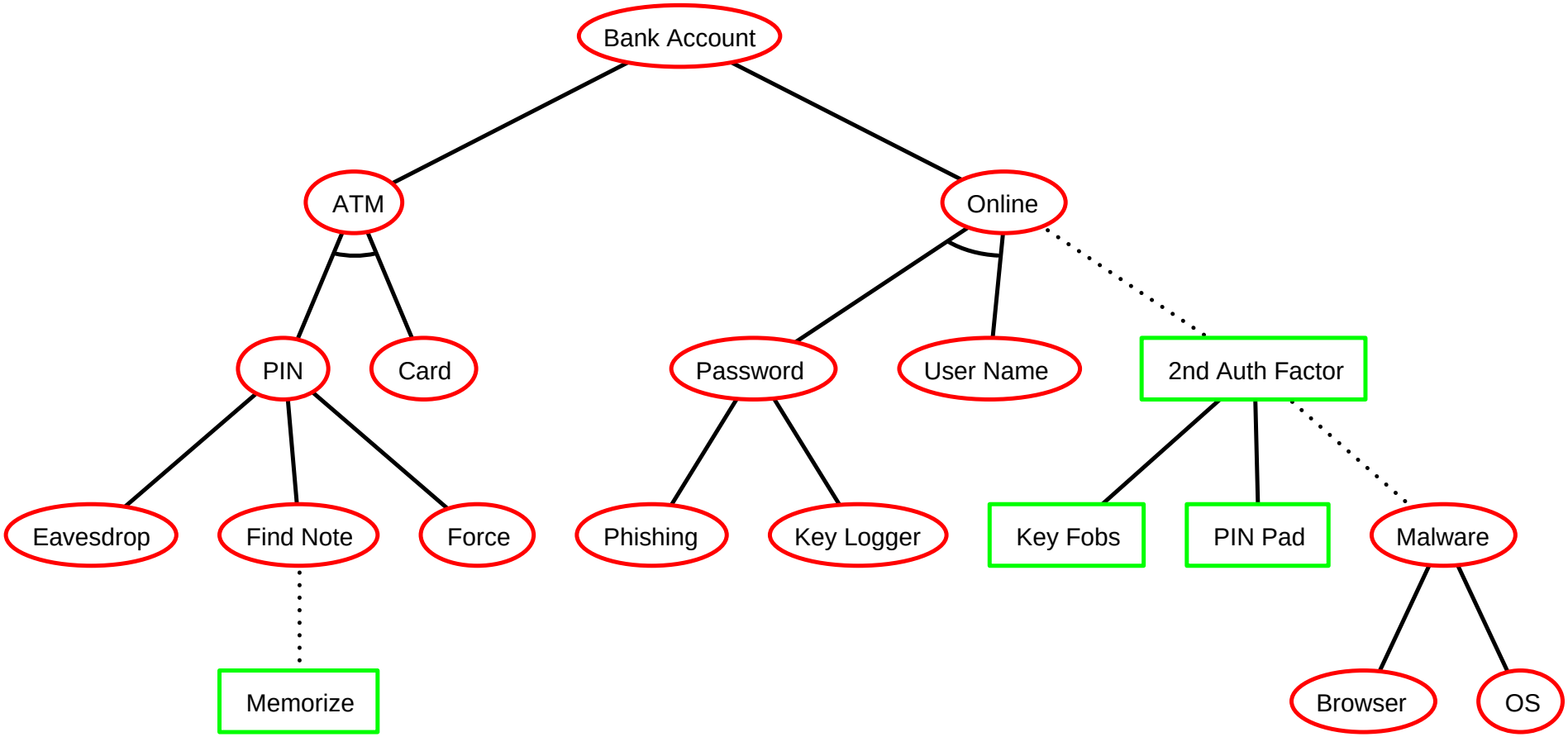
Definition

Attack–defense tree (ADTree) – attack tree extended with possibly refined or countered defensive actions.

Introduced by Kordy et al. in

Foundations of Attack–Defense Trees [FAST'10]

Example: attacking and defending a bank account



Strengths of attack–defense trees

- Defense nodes allowed at any level of a tree
- Countermeasures can be refined
- Countermeasures can be attacked, and so on
- Intuitive visual representation + term-based, formal syntax
- Numerous formal semantics
- Quantitative analysis
- Dedicated software tool

Outline

Motivation

Quantitative analysis of an attack–defense scenario

- Standard questions
  - What is the minimal cost of an attack?
  - What is the expected impact of a considered attack?
  - Is special equipment required to attack?
- Bivariate questions
  - How long does it take to secure a system, when the attacker has a limited budget?
  - How does the scenario change if both, the attacker and the defender are affected by a power outage?

Calculation of attributes

Bottom-up algorithm

- Basic assignment – values assigned to basic actions
- Attribute domain – operators specifying how to compute values for other nodes

- Intuitive idea of Schneier

  Attack trees: Modelling Security Threats, '99
- Formalization by Mauw and Oostdijk for attack trees

  Foundations of Attack Trees, [ICISC'05]
- Extension to attack–defense trees by Kordy et al.

  Foundations of Attack–Defense Trees, [FAST'10]

Attribute: minimal time of an attack

Question:

What is the **minimal time** needed for the attacker to achieve a considered attack, when actions are executed sequentially?

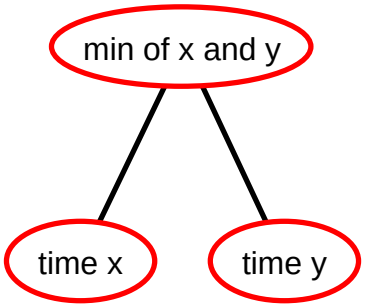
How to specify quantitative questions on attack–defense trees

Quantitative Questions on Attack–Defense Trees., [ICISC'12]

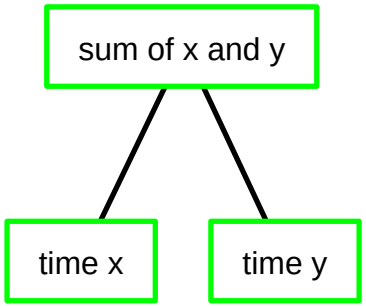
Attribute domain:

- Values from $\mathbb{N} \cup \{\infty\}$
- ∞ = action not under control of the attacker
- $(\vee^A, \wedge^A, \vee^D, \wedge^D, \mathsf{c}^A, \mathsf{c}^D) = (\min, +, +, \min, +, \min)$

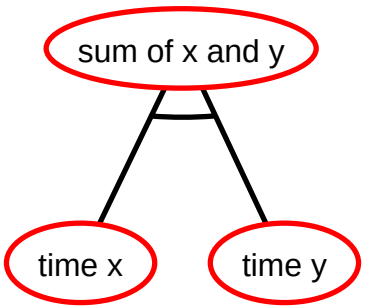
Attribute domain for minimal time



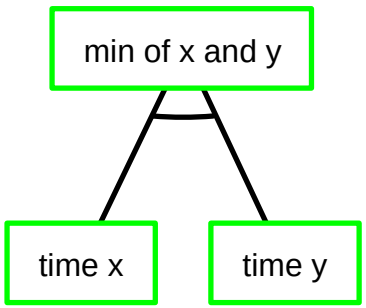
$\vee^A$: $\min\{x, y\}$



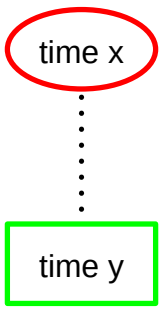
$\vee^D$: $x + y$



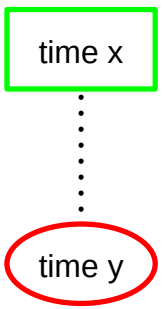
$\wedge^A$: $x + y$



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

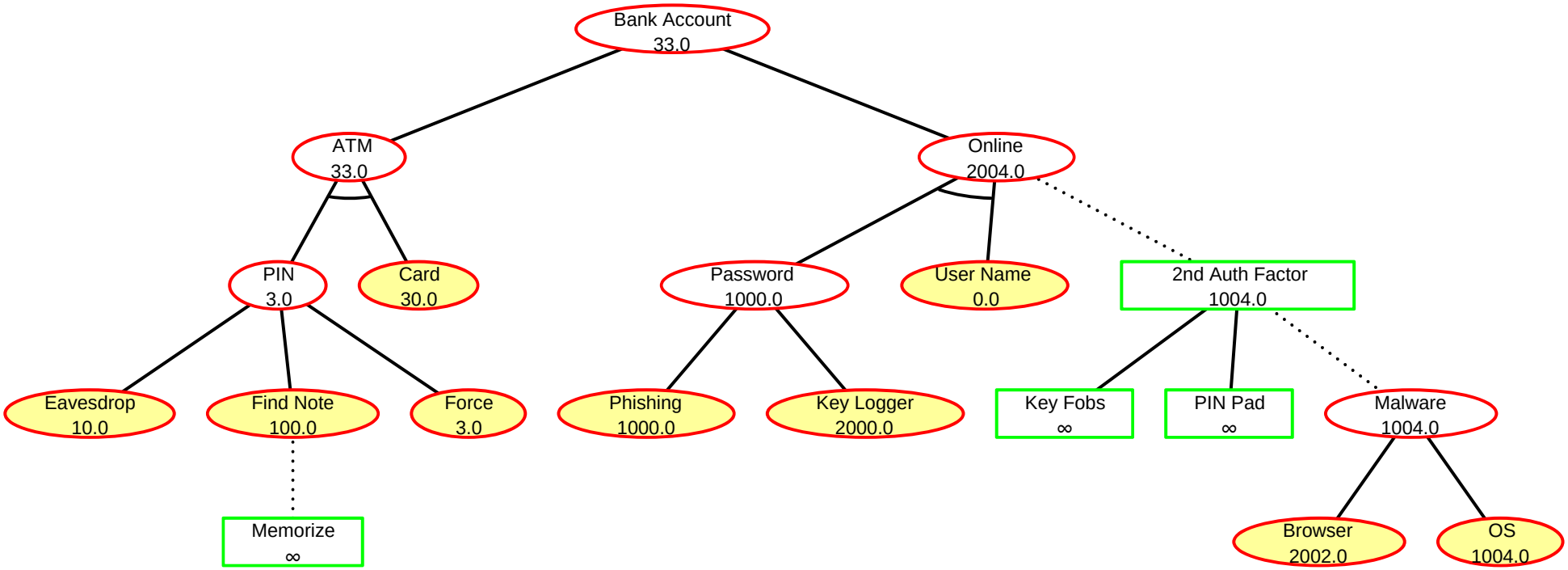


c^A: $x + y$



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

Example: computation of minimal time on an ADTree



Outline

Software for attack–defense trees

ADTool

Free software tool supporting the attack–defense tree methodology

ADTool: Security Analysis with Attack–Defense Trees [QEST'13]

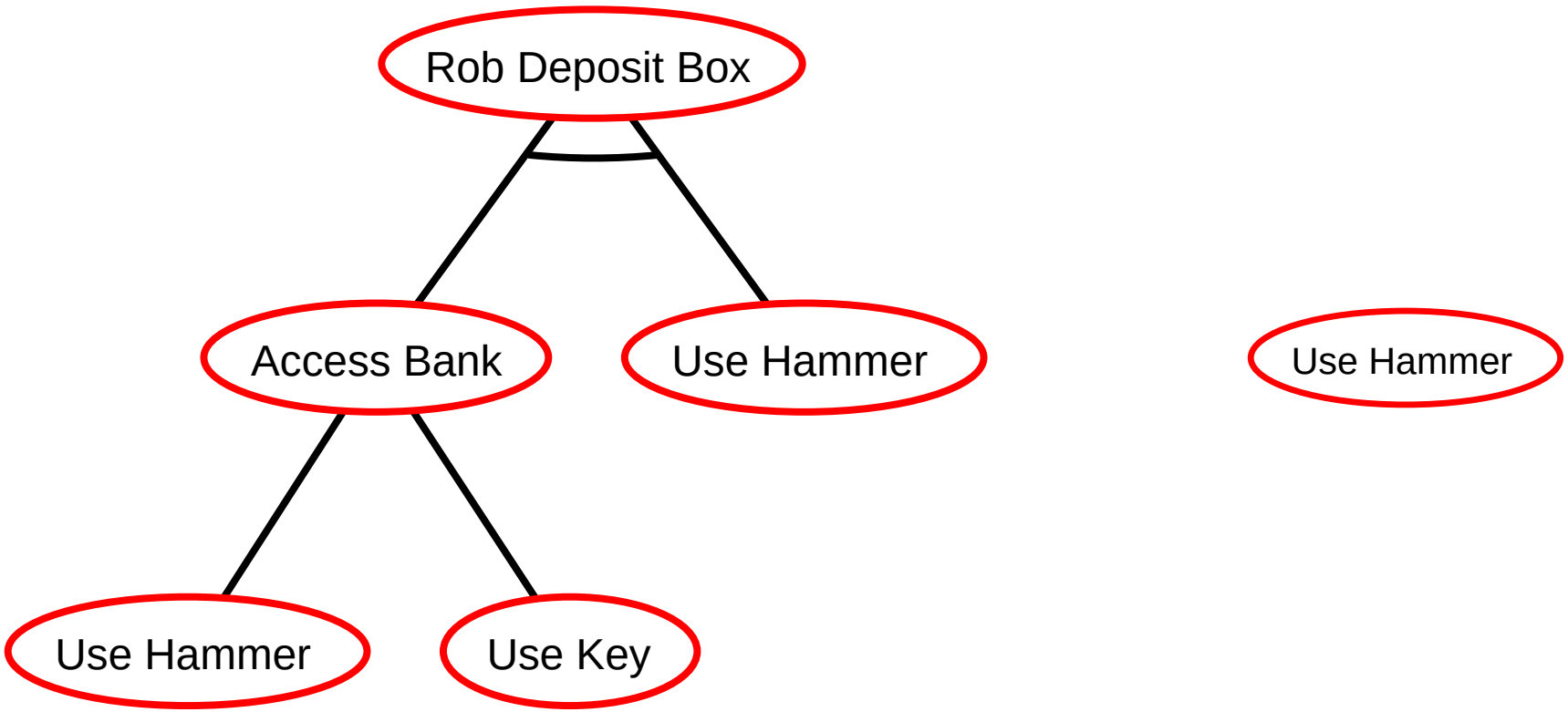
ADTool specification

- Implemented in Java
- Compatible with Windows, Linux, MC OS
- Download
  http://satoss.uni.lu/projects/atrees/adtool
- ADTool documentation and user manual
  http://satoss.uni.lu/projects/atrees/adtool/manual.pdf

Outline

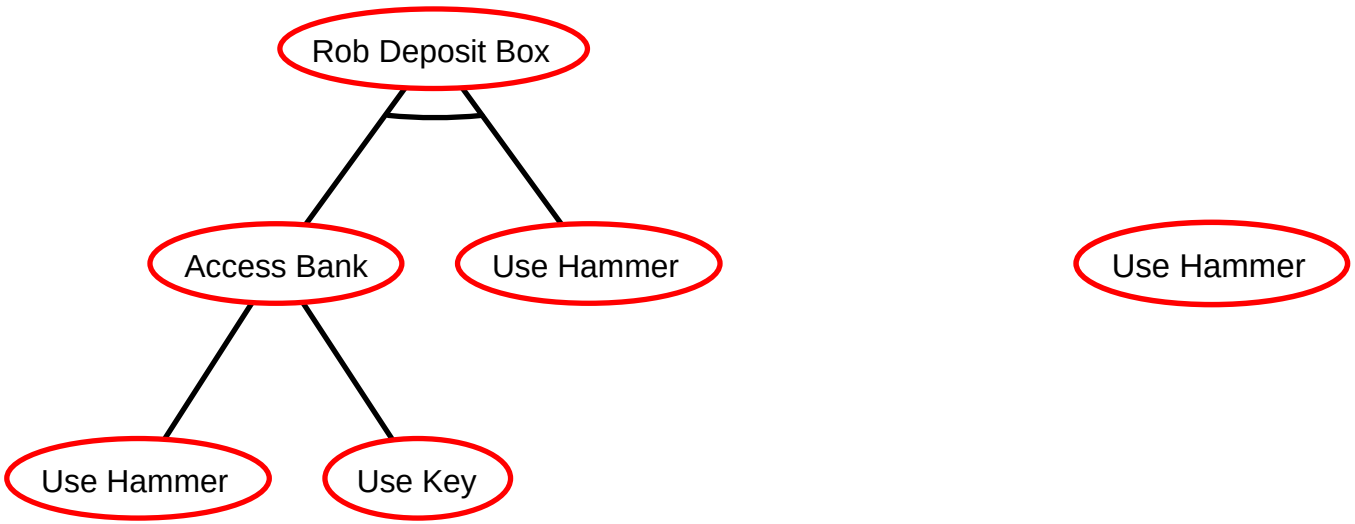
Motivation

Do the two trees represent the same scenario?



Do the two trees represent the same scenario?

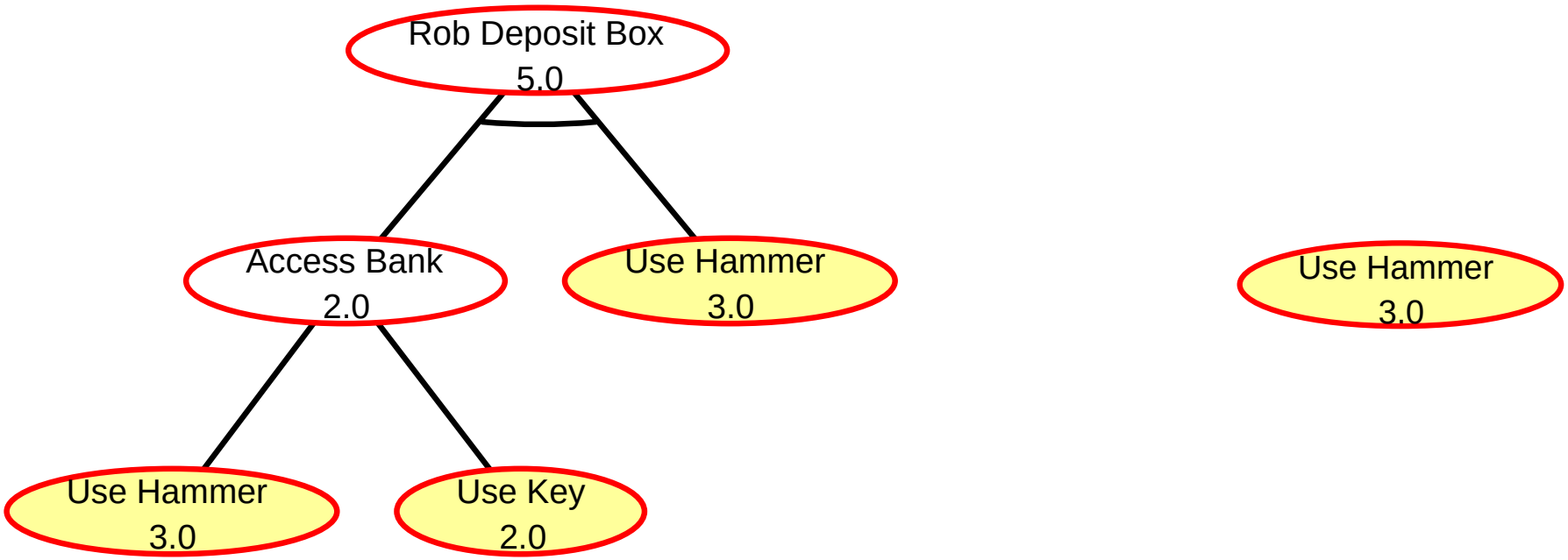
Yes, if we are interested in **which components are necessary**



In both scenarios, the necessary component is having a hammer

Do the two trees represent the same scenario?

No, if we are interested in what is the **minimal attack time**



Definition

Semantics define which ADTrees represent the same scenario.

Definition

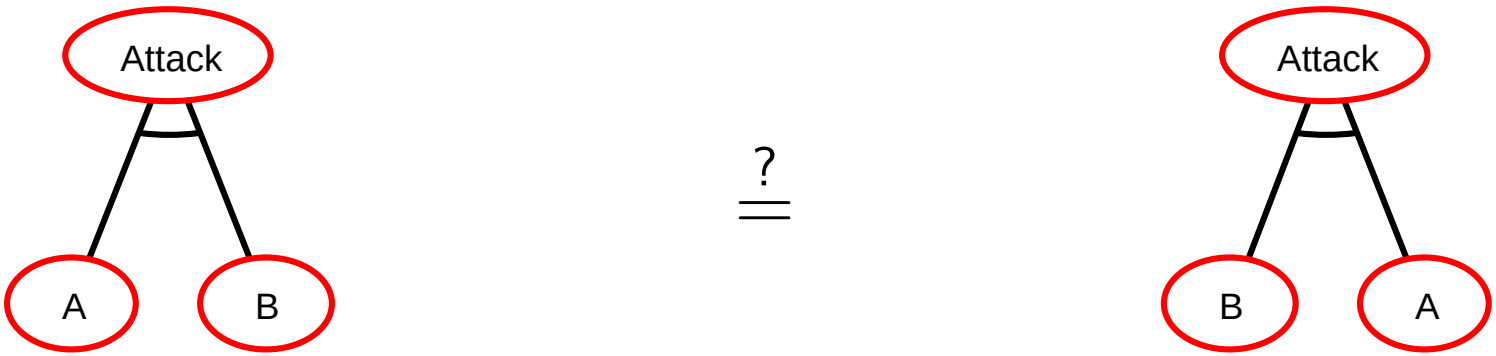
Semantics for ADTrees – equivalence relation on ADTrees

- Propositional semantics
- Semantics induced by a De Morgan lattice
- Multiset semantics
- Equational semantics

Role of formal semantics

Formal semantics for attack trees

- Define what is the meaning of used components
- Model used assumptions
- Express which trees represent the same scenario
- Define allowed transformations of trees



The choice of an appropriate semantics depends on considered applications

Propositional semantics for ADTrees

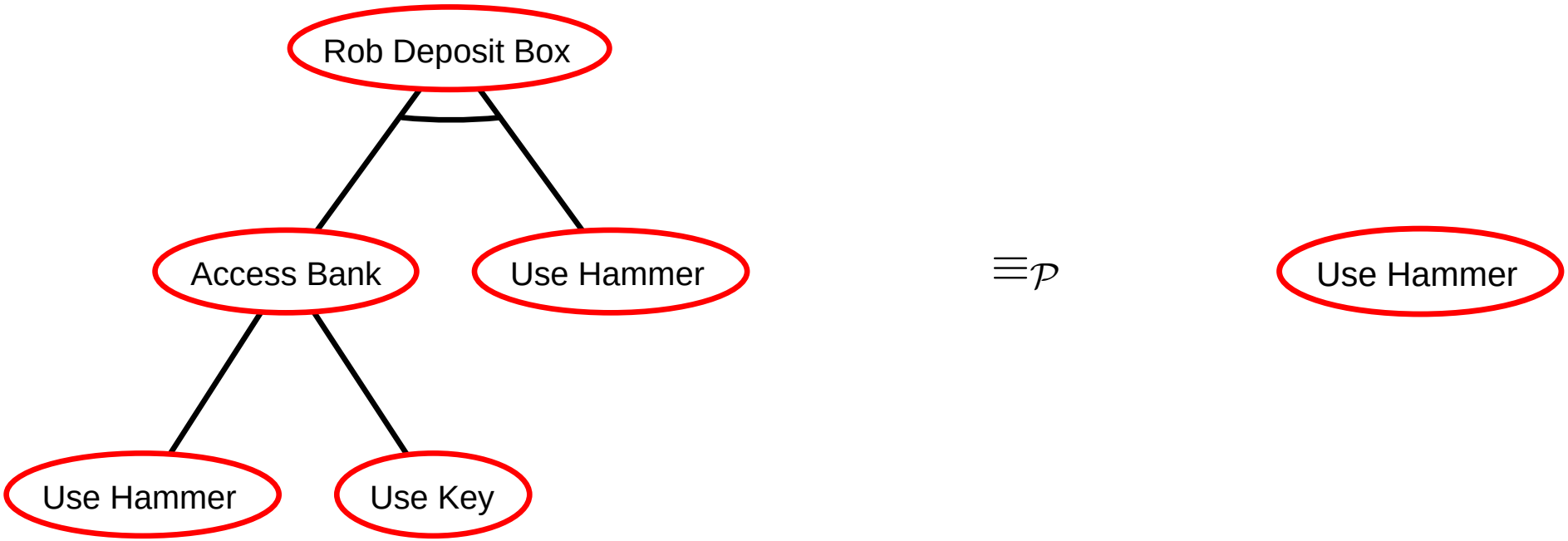
In the propositional semantics

ADTrees are interpreted as propositional formulas.

Equivalent ADTrees

ADTrees represent the same scenario if the corresponding propositional formulas are equivalent.

Example: propositionally equivalent ADTrees



$$(\text{hammer} \vee \text{key}) \wedge \text{hammer} \equiv \text{hammer}$$

Absorption law implies that the two trees are equivalent in the propositional semantics

Multiset semantics $\equiv_{\mathcal{M}}$

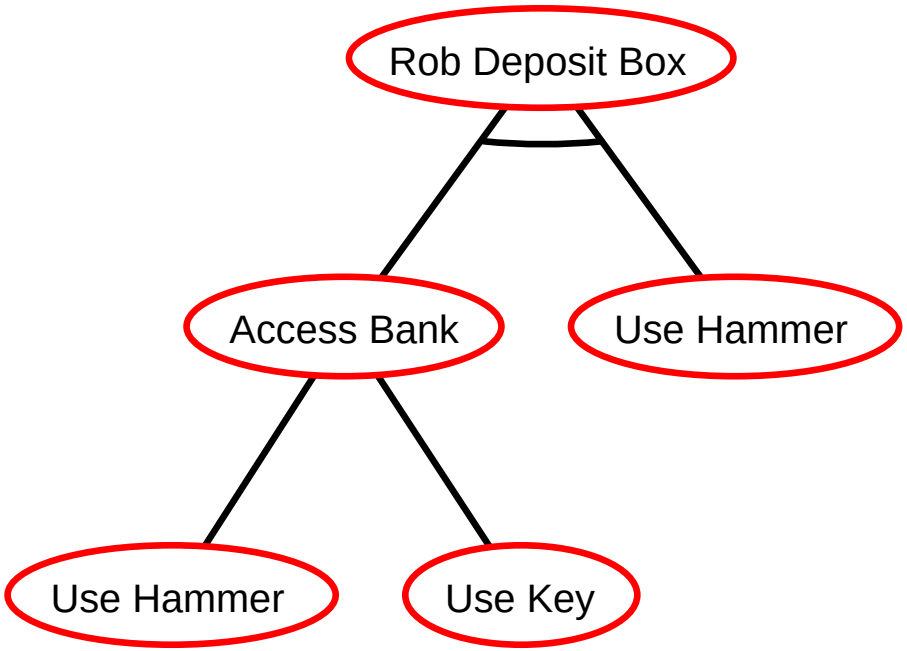
In the multiset semantics

ADTrees are interpreted as sets of multisets.
Each multiset represents a possible way of attacking.

Equivalent ADTrees

ADTrees represent the same scenario if the corresponding sets of multisets are equal.

Example: ADTrees not equivalent in the multiset semantics



$\not\equiv_{\mathcal{M}}$

$$\{\{\!|\text{hammer}, \text{hammer}|\!\}, \{\!|\text{key}, \text{hammer}|\!\}\} \neq \{\{\!|\text{hammer}|\!\}\}$$

Thus, the two trees are not equivalent in the multiset semantics

Compatibility of an attribute with a semantics

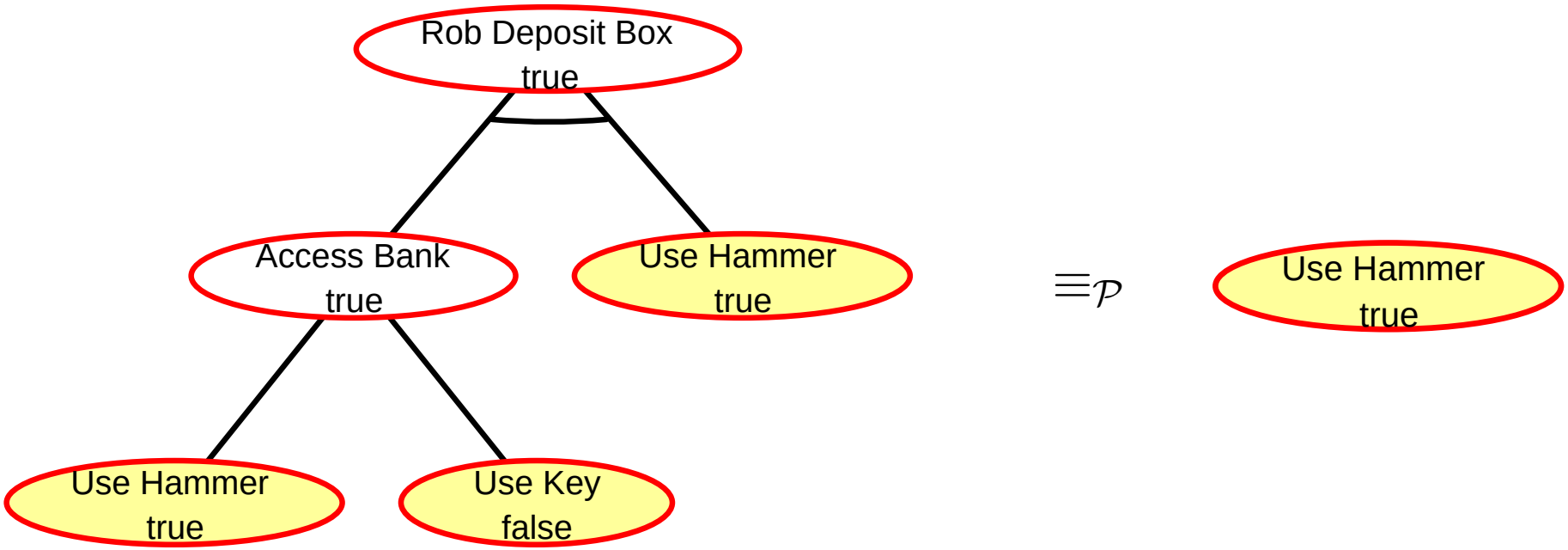
Compatibility defines which semantics should be used in combination with which attribute.

Definition

Attribute α is compatible with semantics $\mathcal{S}$ iff all ADTrees equivalent in $\mathcal{S}$ result in the same value for α.

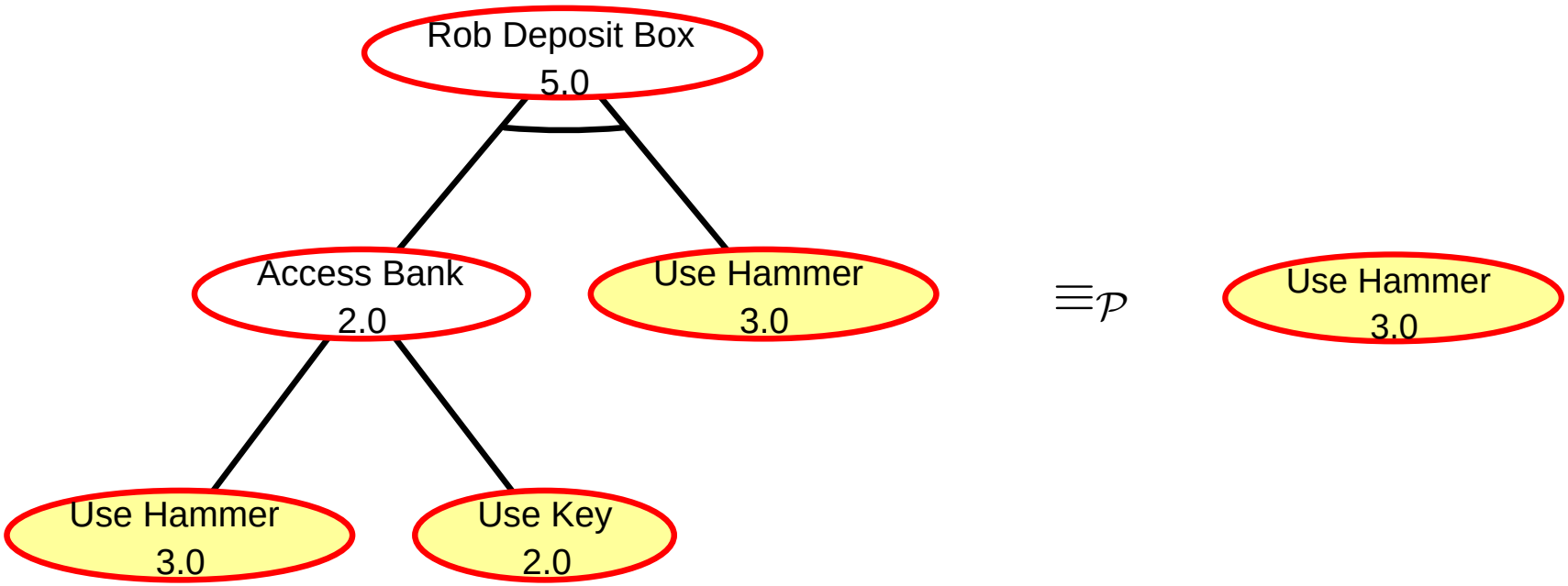
Methods for checking compatibility have been developed by Kordy et al., in *Attack–Defense Trees*, [JLC'14] and *Computational Aspects of Attack–Defense Trees* [SIIS'11]

Example: compatibility



Satisfiability attribute is compatible with $\mathcal{P}$

Counterexample: compatibility



Minimal attack time attribute is not compatible with $\mathcal{P}$

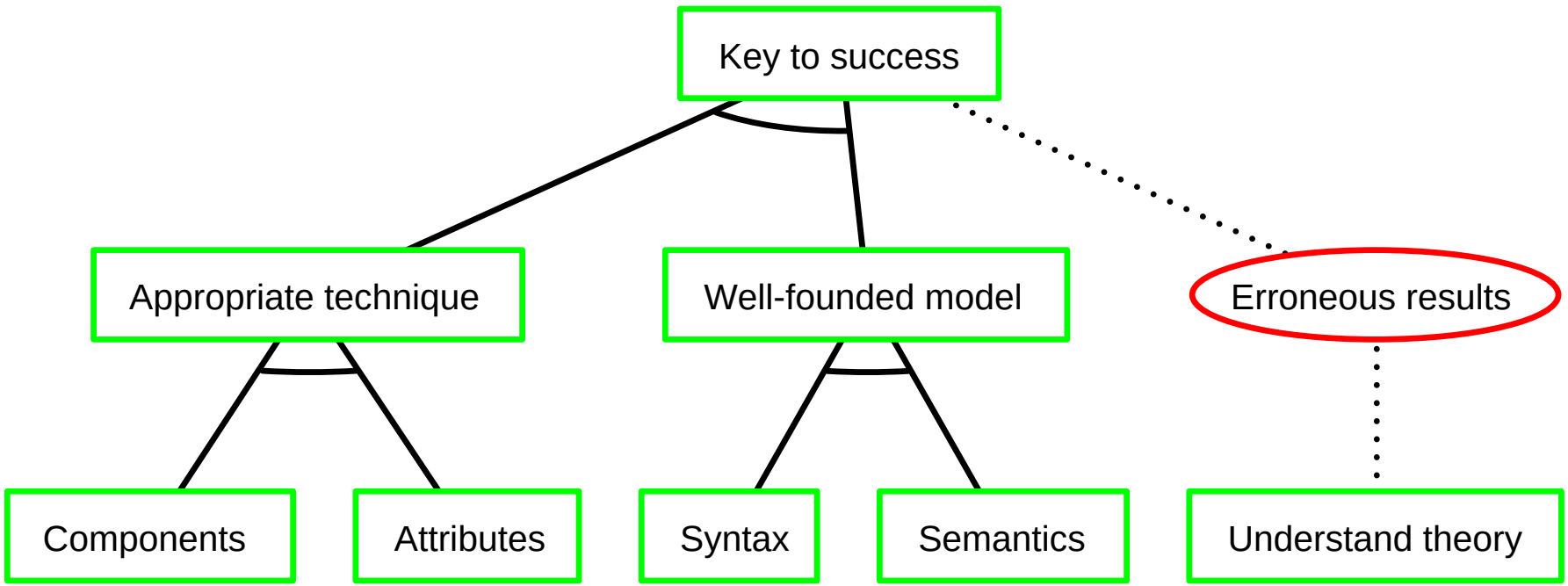
Outline

Active research questions

- Sequential AND.
- Extending with Markov chains.
- Defining libraries.
- Factorizing attack trees.
- Generating attack trees.
- Countermeasure selection.
- Application in Moving Target Defense.
- Application in Cyber Insurance.

Outline

Take home message



References

Barbara Kordy, Sjouke Mauw, Saša Radomirović, and Patrick Schweitzer.
Foundations of Attack–Defense Trees.
In *FAST'10, volume 6561 of* LNCS, *pages 80–95. Springer, 2011.*

Barbara Kordy, Sjouke Mauw, Saša Radomirović, and Patrick Schweitzer.
Attack–Defense Trees.
Journal of Logic and Computation , *24(1), pages 55–87, (2014),*
http://logcom.oxfordjournals.org/content/24/1/55

Barbara Kordy, Piotr Kordy, Sjouke Mauw, and Patrick Schweitzer.
ADTool: Security Analysis with Attack–Defense Trees.
In QEST'13. *To appear in Springer, 2013.*
http://satoss.uni.lu/projects/atrees/adtool.

Barbara Kordy, Sjouke Mauw, and Patrick Schweitzer.
Quantitative Questions on Attack–Defense Trees.
In ICISC'12, *volume 7839 of* LNCS *pages 49–64. Springer, 2013.*

Barbara Kordy, Ludovic Piètre-Cambacédès, and Patrick Schweitzer.
DAG-Based Attack and Defense Modeling: Don't Miss the Forest for the Attack Trees.
In CoRR abs/1303.7397, 2013.
http://arxiv.org/abs/1303.7397 (under submission)

References

Alessandra Bagnato, Barbara Kordy, Per Håkon Meland, and Patrick Schweitzer.
Attribute Decoration of Attack–Defense Trees.
International Journal of Secure Software Engineering, Special Issue on Security Modeling 3, 2 (2012), 1–35.

Barbara Kordy, Marc Pouly, and Patrick Schweitzer.
A Probabilistic Framework for Security Scenarios with Dependent Actions.
In *iFM'14*, *LNCS*. Springer, 2014.

Barbara Kordy, Marc Pouly, and Patrick Schweitzer.
Computational Aspects of Attack–Defense Trees.
In *S&IIS'11*, volume 7053 of *LNCS*, pages 103–116. Springer, 2011.

Barbara Kordy, Sjouke Mauw, Matthijs Melissen, and Patrick Schweitzer.
Attack–Defense Trees and Two-Player Binary Zero-Sum Extensive Form Games Are Equivalent.
In *GameSec'10*, volume 6442 of *LNCS*, pages 245–256. Springer, 2010.