Quantitative Questions and Attack–Defense Trees

Patrick Schweitzer*

Joint work with Barbara Kordy and Sjouke Mauw

* supported by the grant No. PHD-09-167 from the FNR Luxembourg

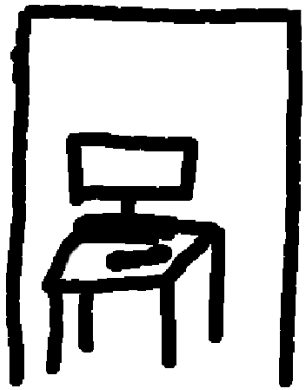
Outline

uni.lu UNIVERSITÉ DU LUXEMBOURG

Outline

Attack–Defense Trees: Introduction

Attack–Defense Trees: Introduction



Attack Trees

Methodology to describe security weaknesses of a system

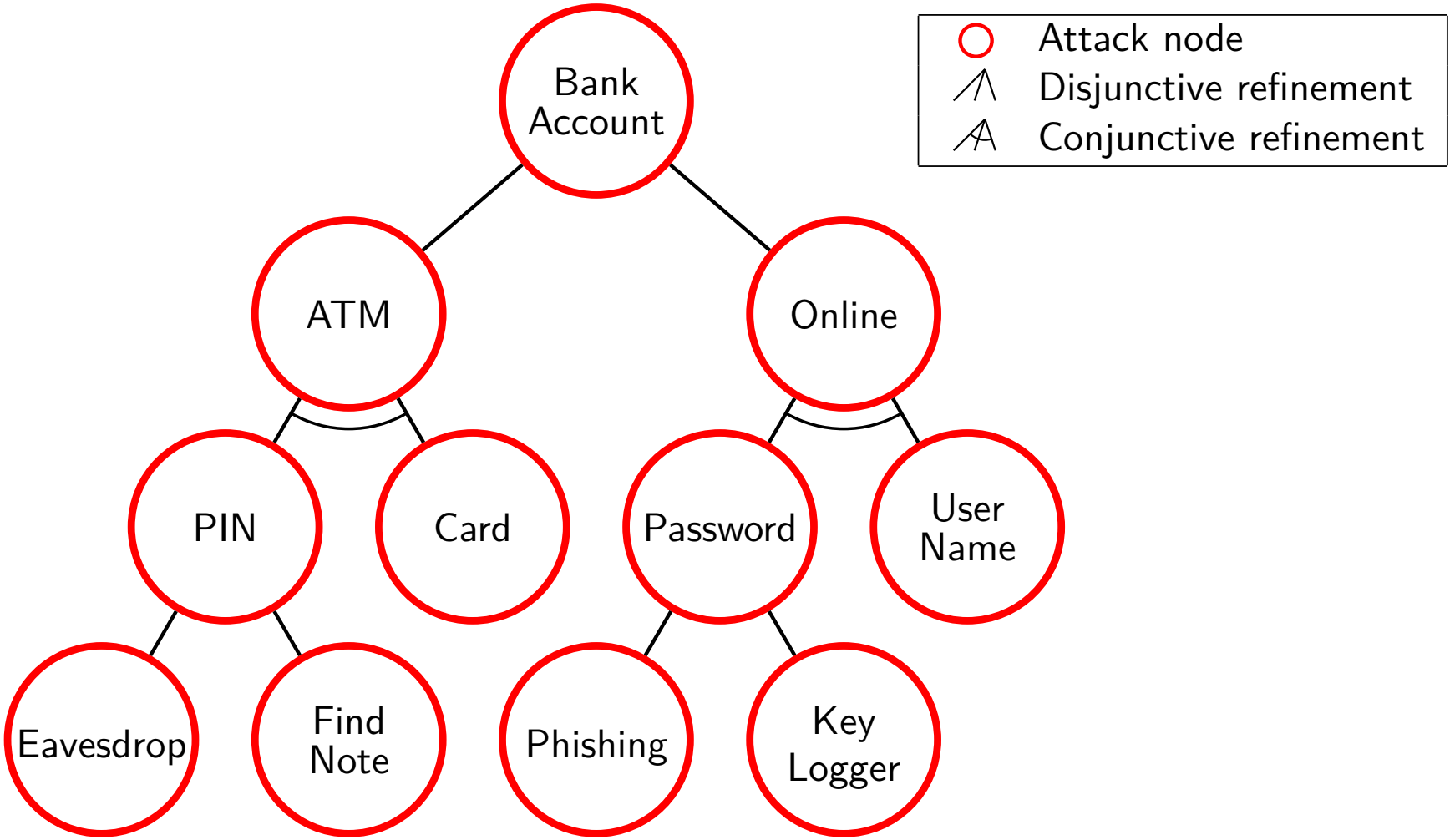
Definition

Attack Trees (ATrees): Tree-like representation of an attacker's goal recursively refined into conjunctive and disjunctive subgoals.

- Successor of fault trees (Safety modeling)
- Security adaptation as threat trees (Weiss, Amoroso)
- Name coined in 1998 (Schneier)
- Formalization in 2005 (Mauw, Oostdijk)

uni.lu UNIVERSITÉ DU LUXEMBOURG

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

uni.lu
UNIVERSITÉ DU LUXEMBOURG

Attack–Defense Trees

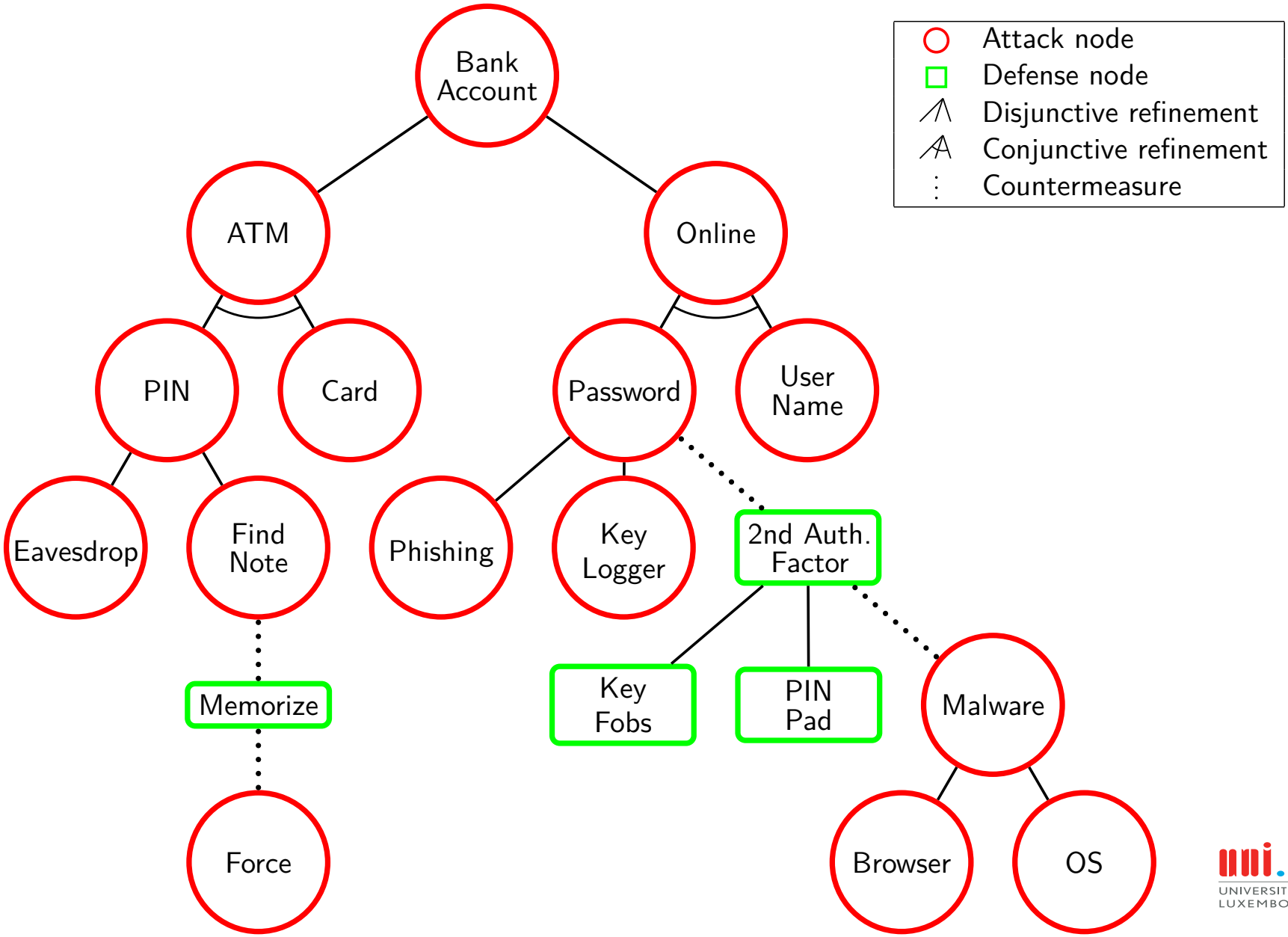
Definition

Attack–Defense Trees (ADTrees): ATrees extended with possibly refined or countered defensive actions.

- Introduced by Kordy, Mauw, Radomirović and Schweitzer in *Foundations of Attack–Defense Trees* [FAST'10]
- Related approaches by
  – Attack-Countermeasure trees: Roy et al. ('10)
  – Augmented attack trees: Wang et al. ('10)

uni.lu UNIVERSITÉ DU LUXEMBOURG

Example: Attacking and Defending a Bank Account



Interesting Questions

- Equivalent representations of the same scenario (semantics)
  PS in 2010
- Practical applications (case studies)
  PS in 2011
- Computational complexity of ATrees and ADTrees (querying)
  BK in 2011
- Quantitative analysis (attributes)
  Today

uni.lu UNIVERSITÉ DU LUXEMBOURG

Outline

Quantitative Analysis: Introduction

- Standard questions (ATrees)
  - What are the costs of attacking the system?
  - How many specialists are needed to attack a system?
  - Can the attack succeed?

Quantitative Analysis: Introduction

- Standard questions (ATrees)
  - What are the costs of attacking the system?
  - How many specialists are needed to attack a system?
  - Can the attack succeed?
- Bivariate questions (ADTrees)
  - What are the minimal costs of an attack when the defender tries to maximally protect his system?
  - How long does it take the defender to secure a system, assuming independent work on defenses, when the attacker has a limited budget?

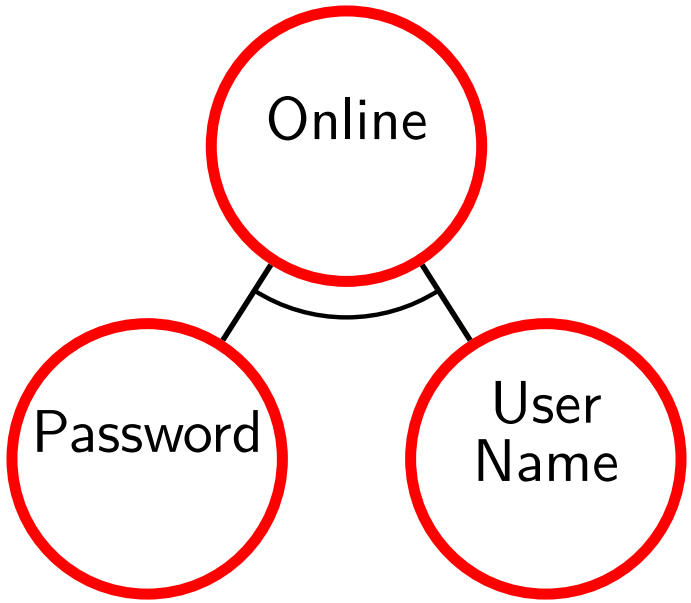
Quantitative Analysis: Introduction

- Standard questions (ATrees)
  - What are the costs of attacking the system?
  - How many specialists are needed to attack a system?
  - Can the attack succeed?
- Bivariate questions (ADTrees)
  - What are the minimal costs of an attack when the defender tries to maximally protect his system?
  - How long does it take the defender to secure a system, assuming independent work on defenses, when the attacker has a limited budget?

Underspecified Questions

How many specialists are needed to attack the system?

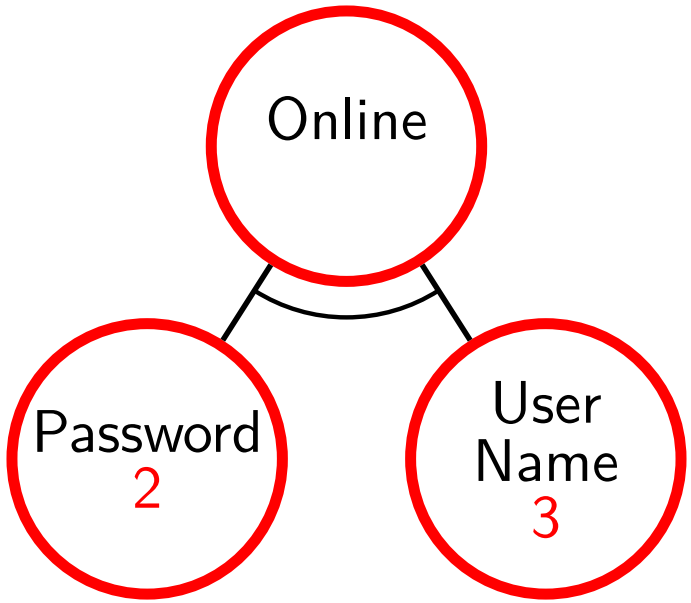
Intuitive answers welcome



Underspecified Questions

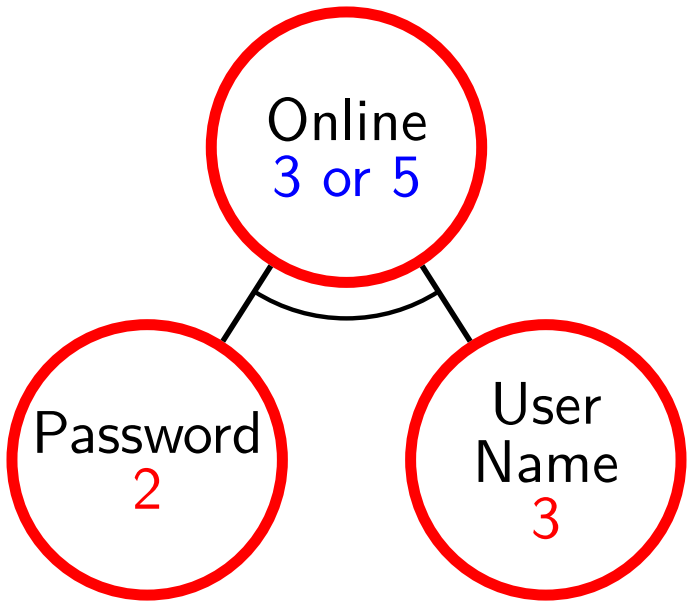
How many specialists are needed to attack the system?

Intuitive approaches welcome



Underspecified Questions

How many specialists are needed to attack the system?



$\longrightarrow$ The question is underspecified

Calculation of Attributes

Bottom-up algorithm

- Attribute domain: operators specifying how to compute values for refined nodes
- Basic assignment: values assigned to non-refined nodes

uni.lu UNIVERSITÉ DU LUXEMBOURG

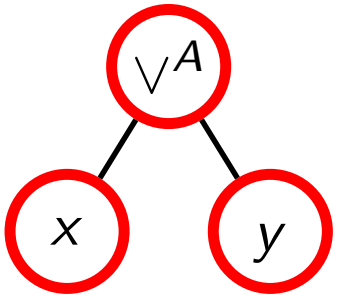
Minimal Number of Specialists Needed to Attack

Question:

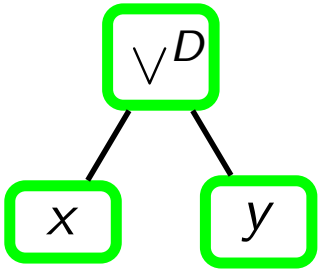
What are the minimal number of specialists needed to attack?

- Values from $\mathbb{N} \cup \{\infty\}$
- $\infty =$ action not under control of the attacker
- $(\vee^A, \wedge^A, \vee^D, \wedge^D, \mathsf{c}^A, \mathsf{c}^D) \mapsto (\min, +, +, \min, +, \min)$
- Attribute domain A consists of a value domain plus mapping of the six functional operators.

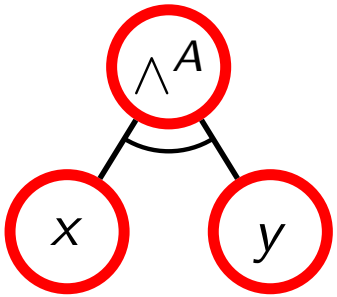
Attribute Domain for Minimal Number of Specialists (Ind.)



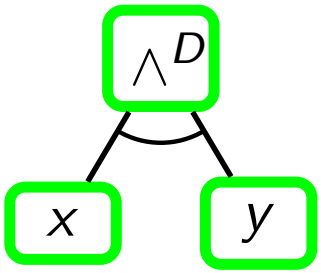
$\vee^A$: $\min\{x, y\}$



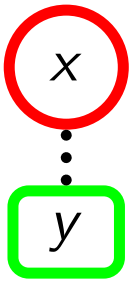
$\vee^D$: $x + y$



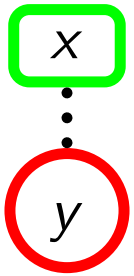
$\wedge^A$: $x + y$



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



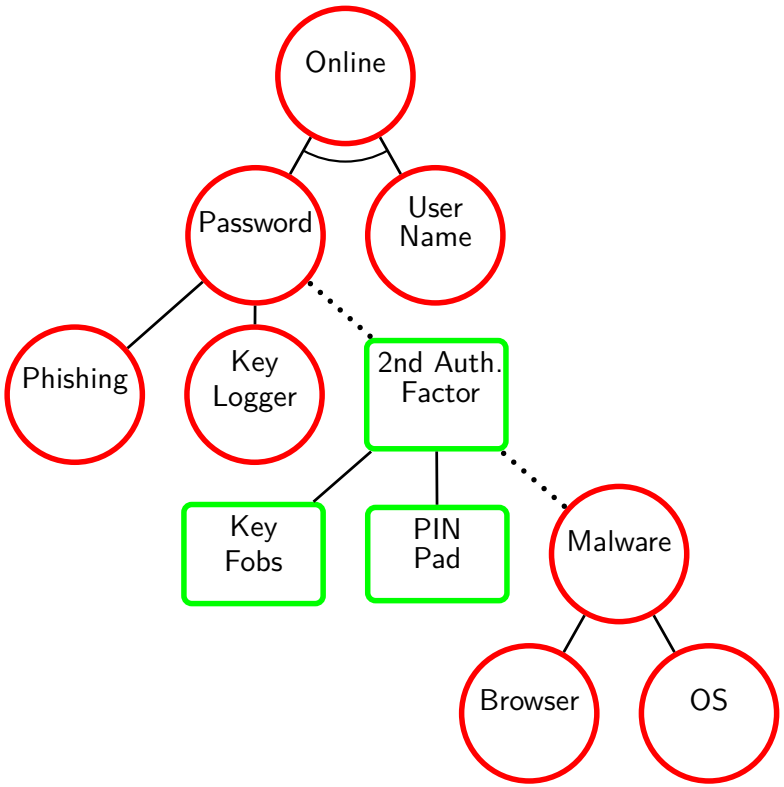
c^A: $x + y$



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

Computation of the Minimal Number of Specialists (Ind.)

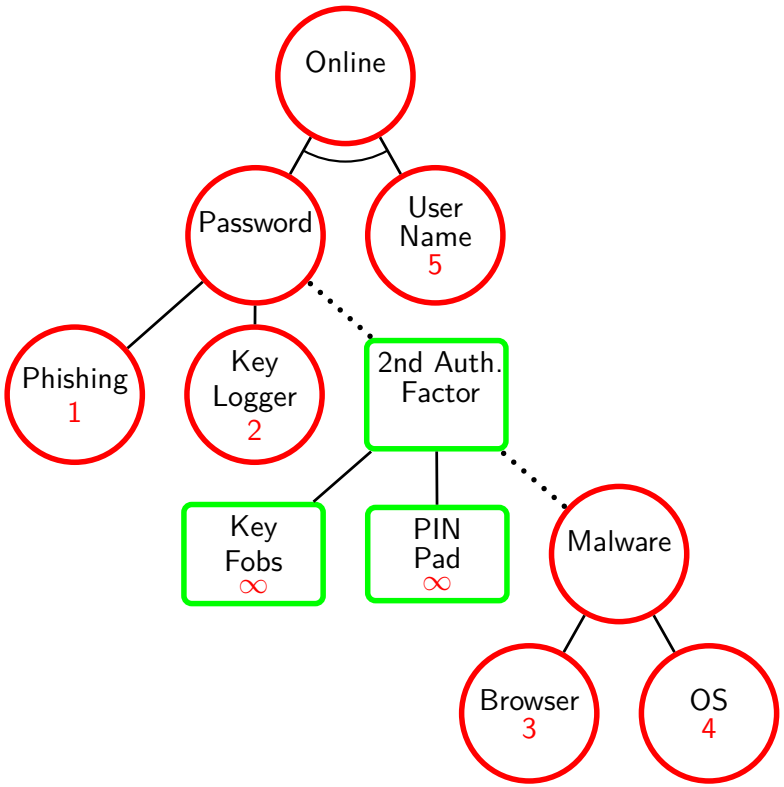
$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathsf{c}^A, \mathsf{c}^D) \mapsto (\mathbb{N} \cup \{\infty\}, \min, +, +, \min, +, \min)$$



- Attribute Domain

Computation of the Minimal Number of Specialists (Ind.)

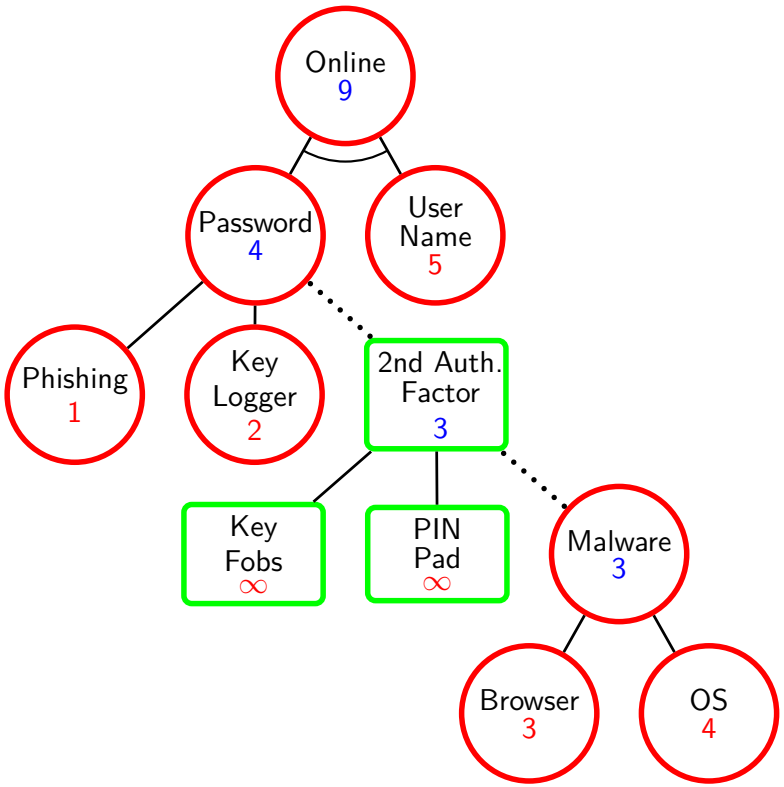
$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathrm{c}^A, \mathrm{c}^D) \mapsto (\mathbb{N} \cup \{\infty\}, \min, +, +, \min, +, \min)$$



- Attribute Domain
- Basic assignment

Computation of the Minimal Number of Specialists (Ind.)

$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathrm{c}^A, \mathrm{c}^D) \mapsto (\mathbb{N} \cup \{\infty\}, \min, +, +, \min, +, \min)$$



- Attribute Domain
- Basic assignment
- Bottom-up computation

Attribute Domain Summary

Take home message

Given an ADTree, an attribute domain, a basic assignment we can use the bottom up algorithm to provide a value for the root node of the ADTree.

uni.lu UNIVERSITÉ DU LUXEMBOURG

Components for an Analysis

Component	Functional Example	ADTree Example
Question	What is the functional value?	What is the minimal number of specialists...?
Model	$f(x, y) = x \circ y$	ADTree and Bottom-up algorithm
Interpretation	$\circ = +$	Attribute domain
Input values	$x = 2, y = 3$	Basic assignment

Components for an Analysis

Component	Functional Example	ADTree Example
Question	What is the functional value?	What is the minimal number of specialists...?
Model	$f(x, y) = x \circ y$	ADTree and Bottom-up algorithm
Interpretation	$\circ = +$	Attribute domain
Input values	$x = 2, y = 3$	Basic assignment

Outline

Questions vs. Attribute Domains: Introduction

- A question is an informal way to describe a quantity.
- An attribute domain is a formal way to describe a quantity.
- Necessary to link questions and attribute domains.
- Specify format of questions to
  - avoid underspecification of attribute domain.
  - automate selection of attribute domain.
  - classify empirically occurring questions.

Exemplary Questions

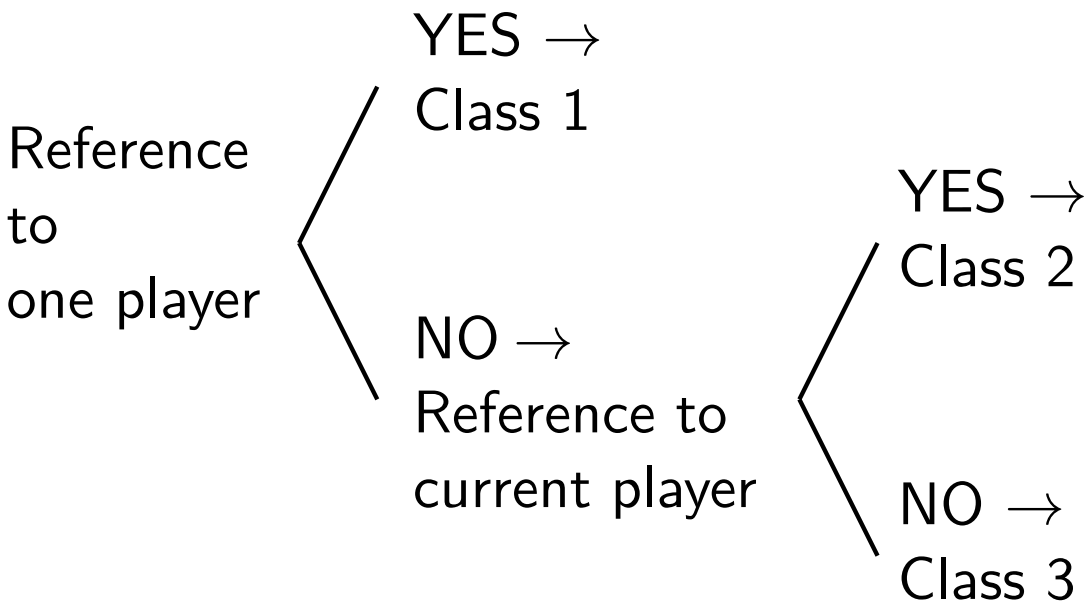
Class 1:
– What is the minimal number of specialists needed to attack the system assuming that the specialists work independently?
– What is the minimal time of the attack assuming that all actions are executed one after another?

Class 2:
– Who is the winner of the scenario?
– How likely is the system to be breached?

Class 3:
– How much data traffic will occur?
– What are the environmental costs?

uni.lu UNIVERSITÉ DU LUXEMBOURG

A Classification of Questions into Three Classes



Exemplary Questions

Class 1:

– What is the minimal number of specialists needed to attack the system assuming that the specialists work independently?

– What is the minimal time of the attack assuming that all actions are executed one after another?

Class 2:

– Who is the winner of the scenario?

– How likely is the system to be breached?

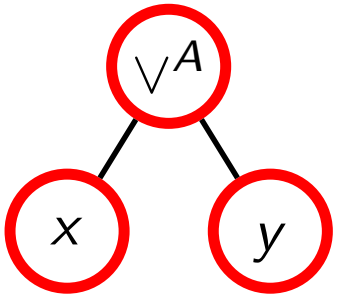
Class 3:

– How much data traffic will occur?

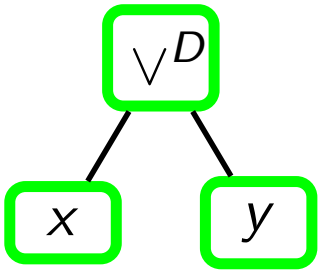
– What are the environmental costs?

uni.lu UNIVERSITÉ DU LUXEMBOURG

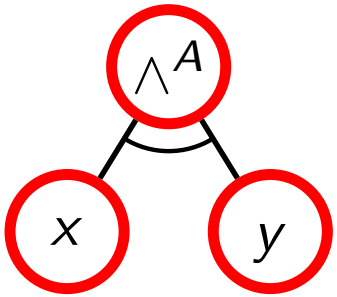
Minimal Number of Specialists (Independent)



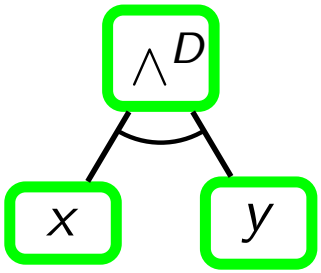
$\vee^A$: minimum



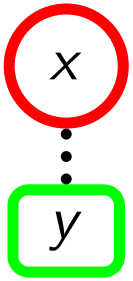
$\vee^D$: addition



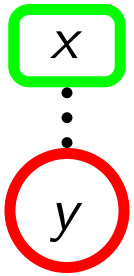
$\wedge^A$: addition



$\wedge^D$: minimum

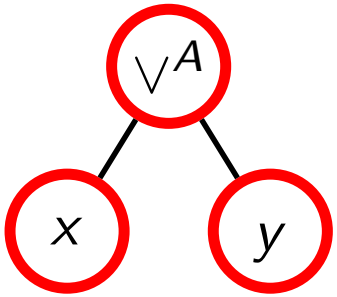


c^A: addition

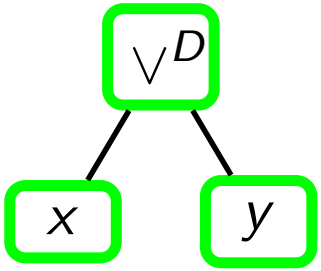


c^D: minimum

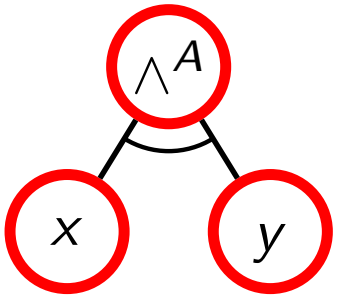
Minimal Number of Specialists (Parallel)



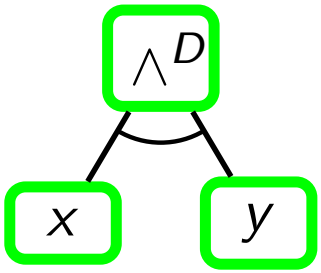
$\vee^A$: minimum



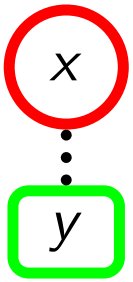
$\vee^D$: maximum



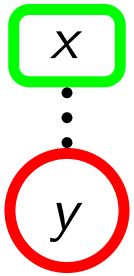
$\wedge^A$: maximum



$\wedge^D$: minimum

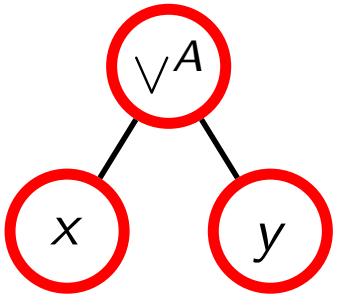


c^A: maximum

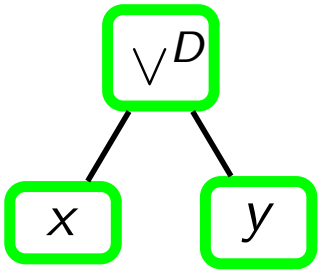


c^D: minimum

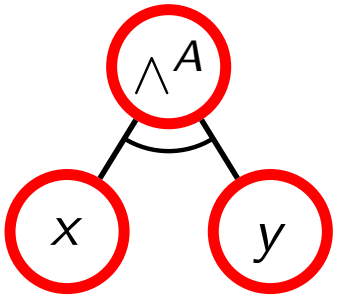
Class 1 Generalization



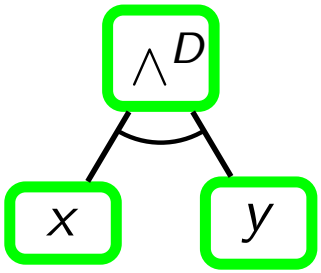
$\vee^A$: at least one



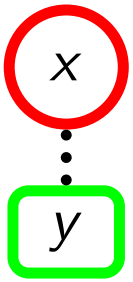
$\vee^D$: all



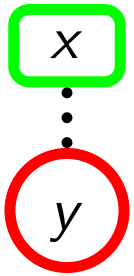
$\wedge^A$: all



$\wedge^D$: at least one



c^A: all



c^D: at least one

$\longrightarrow$ Two different operators suffice

Deducing the Structure of a Question in Class 1

We specify

- the value domain,
- “at least one”,
- “all”,
- whether “at least one” holds for attacker or defender.

$\longrightarrow$ If the question has exactly four parts, the attribute domain is completely determined.

Example of a Question in Class 1

1. What is the minimal
2. number of specialists needed
3. to attack the system
4. assuming that the specialists work independently?

Part 2 - Notion: determines the value domain
Part 1 - Modality: determines the operator for “at least one”
Part 4 - Execution style: determines the operator for “all”
Part 3 - Owner of the question: determines assignment of “at least one” and “all” to attacker and defender.

uni.lu UNIVERSITÉ DU LUXEMBOURG

Part 2 of a Question in Class 1

- attack potential,
- attack time,
- consequence,
- costs,
- detectability,
- difficulty level,
- elapsed time,
- expertise,
- impact,
- insider required,
- mitigation success,
- outcome,
- penalty,
- profit,
- response time,
- resources,
- severity,
- skill level,
- special equipment needed,
- special skill needed,
- survivability.

Parts 1,3 and 4 of a Question in Class 1

Notion (Part 2)	Modality (Part 1)	Owner (Part 3)	Exec.style (Part 4)	Structure $(D, \mathrm{op}_1, \mathrm{op}_2)$
duration	min	A	sequential	$(\mathbb{R}, \min, +)$
duration	max	D	sequential	$(\mathbb{R}, +, \max)$
duration	comb	A	parallel	$(\mathbb{R}, \max, \max)$
...	...	...	...	...

$$(D, \mathrm{op}_1, \mathrm{op}_2) \longrightarrow (D, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_2, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_1)$$

uni.lu UNIVERSITÉ DU LUXEMBOURG

Questions for ADTrees in Class 1 Summary

Message to take home

A question for ADTrees in Class 1 needs to contain four components to construct the attribute domain: notion, modality, owner and execution style.

Exemplary Questions

Class 1:
– What is the minimal number of specialists needed to attack the system assuming that the specialists work independently?
– What is the minimal time of the attack assuming that all actions are executed one after another?

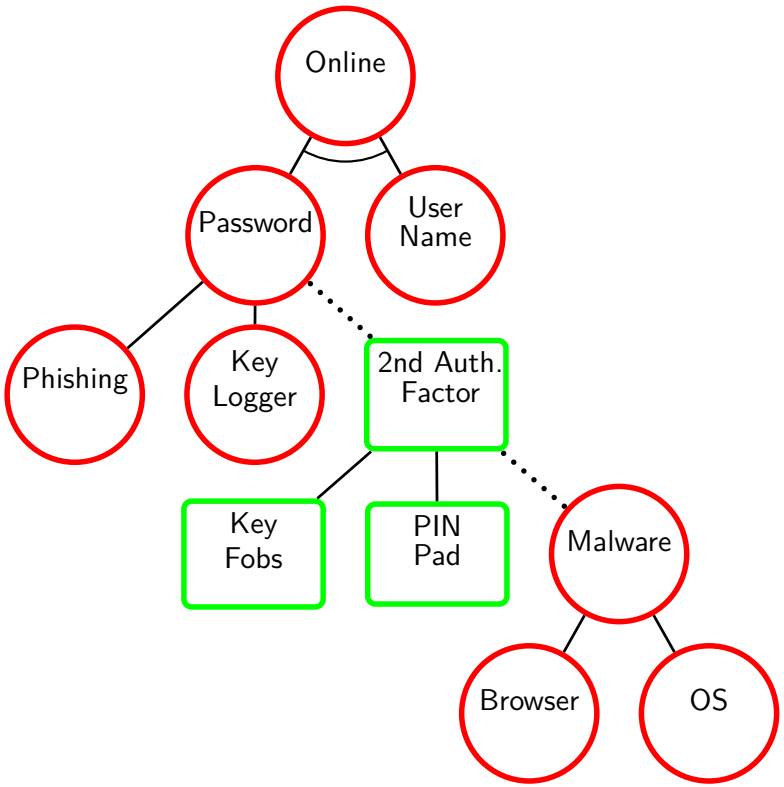
Class 2:
– Who is the winner of the scenario?
– How likely is the system to be breached?

Class 3:
– How much data traffic will occur?
– What are the environmental costs?

uni.lu UNIVERSITÉ DU LUXEMBOURG

Example: Computation of the Winning Player

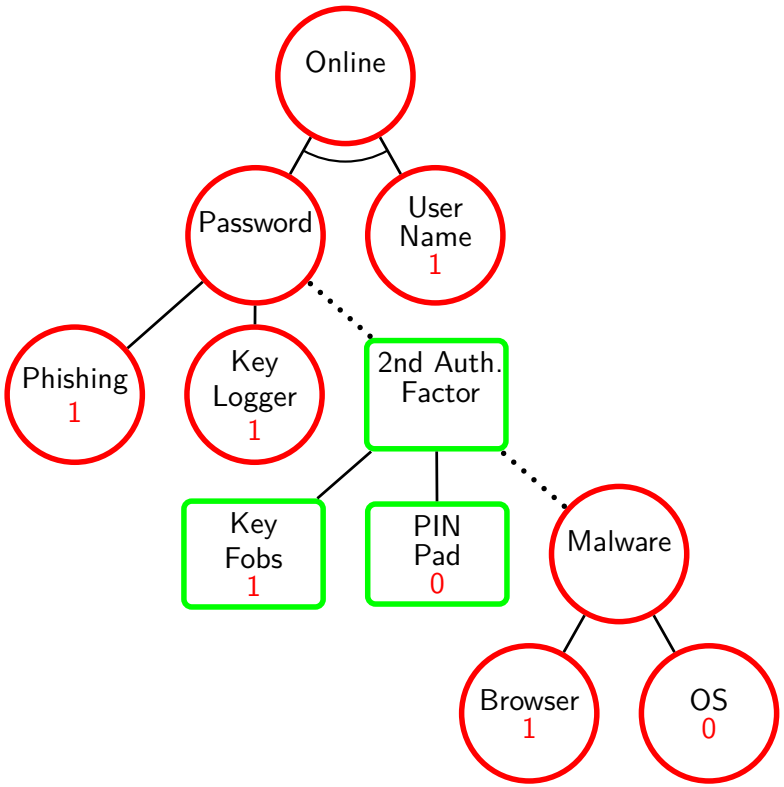
$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathrm{c}^A, \mathrm{c}^D) \mapsto (\{0,1\}, \vee, \wedge, \vee, \wedge, \star, \star)$$



- Attribute Domain, where $a \star b = a \wedge \neg b$

Example: Computation of the Winning Player

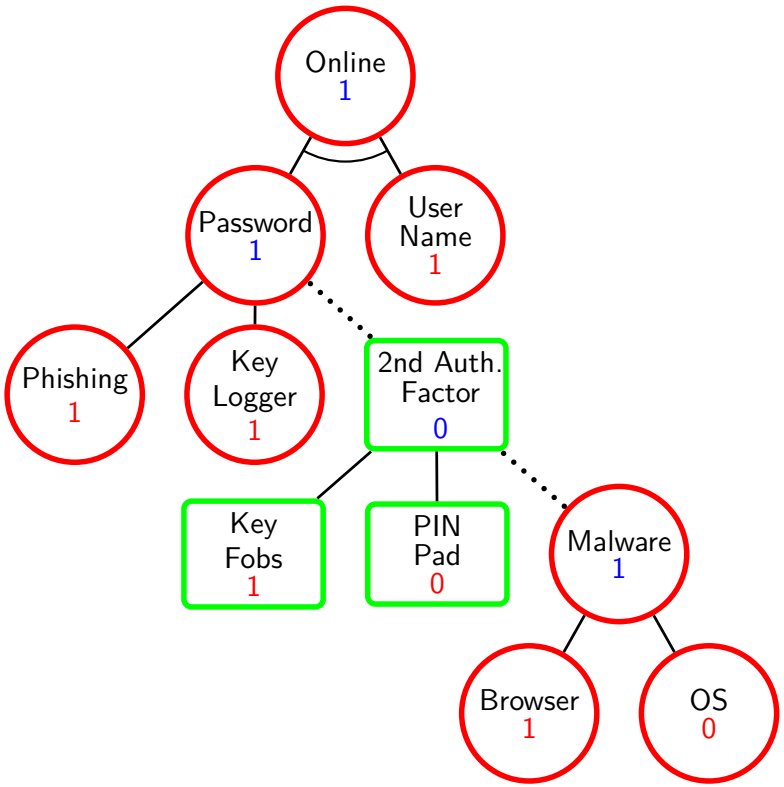
$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathrm{c}^A, \mathrm{c}^D) \mapsto (\{0,1\}, \vee, \wedge, \vee, \wedge, \star, \star)$$



- Attribute Domain, where $a \star b = a \wedge \neg b$
- Basic assignment: 1's and 0's

Example: Computation of the Winning Player

$$A = (D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathrm{c}^A, \mathrm{c}^D) \mapsto (\{0,1\}, \vee, \wedge, \vee, \wedge, \star, \star)$$



- Attribute Domain, where $a \star b = a \wedge \neg b$
- Basic assignment: 1's and 0's
- Bottom-up computation

Class 2 Generalization

- Reference to the current player.
- The value for a subtree answers the question for the subtree.
- We need three operators.
- Values of a player can be transformed to values of the other player, using a negation function.
- Given the negation function the third operator can be expressed in terms for the second operator.
- Deduction of the parts of the question is similar to deduction in Class 1.

Exemplary Questions

Class 1:
– What is the minimal number of specialists needed to attack the system assuming that the specialists work independently?
– What is the minimal time of the attack assuming that all actions are executed one after another?

Class 2:
– Who is the winner of the scenario?
– How likely is the system to be breached?

Class 3:
– How much data traffic will occur?
– What are the environmental costs?

uni.lu UNIVERSITÉ DU LUXEMBOURG

Attributes in Class 3

- Questions neither have a reference to the current player nor the owner of the question.
- Both the attacker and the defender contribute equally to the result.
- Deduction of the parts of the question is similar to Class 1.

Class Summary

	Attribute Domain $(D, \vee^A, \wedge^A, \vee^D, \wedge^D, \mathsf{c}^A, \mathsf{c}^D)$
Class 1	$(D, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_2, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_1)$
Class 2	$(D, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_3, \mathrm{op}_3)$
Class 3	$(D, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_1, \mathrm{op}_2, \mathrm{op}_2, \mathrm{op}_2)$

uni.lu UNIVERSITÉ DU LUXEMBOURG

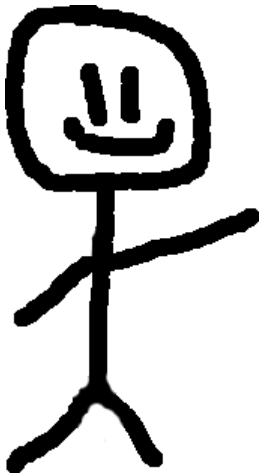
How to Apply our Work to Attack Trees

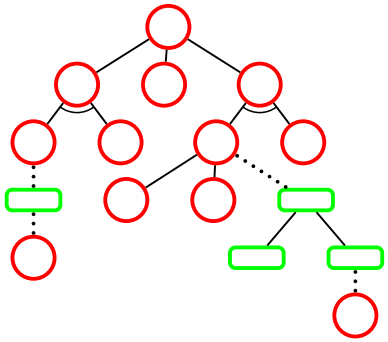
- Attribute domains equally relevant for attack trees.
- An attribute domain for attack trees is the three tuple $(D, \vee^A, \wedge^A)$.
- Simpler case, because the owner of a question is always the attacker.
- Properly specified questions for attack tree contain three of the four parts.
- Classes 1, 2 and 3 coincide for attack trees.

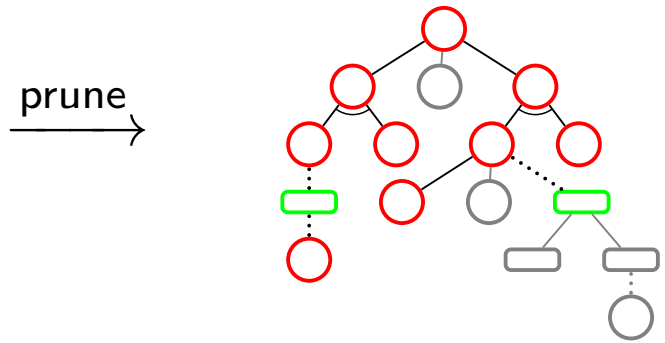
uni.lu UNIVERSITÉ DU LUXEMBOURG

Outline

Pruning: Introduction





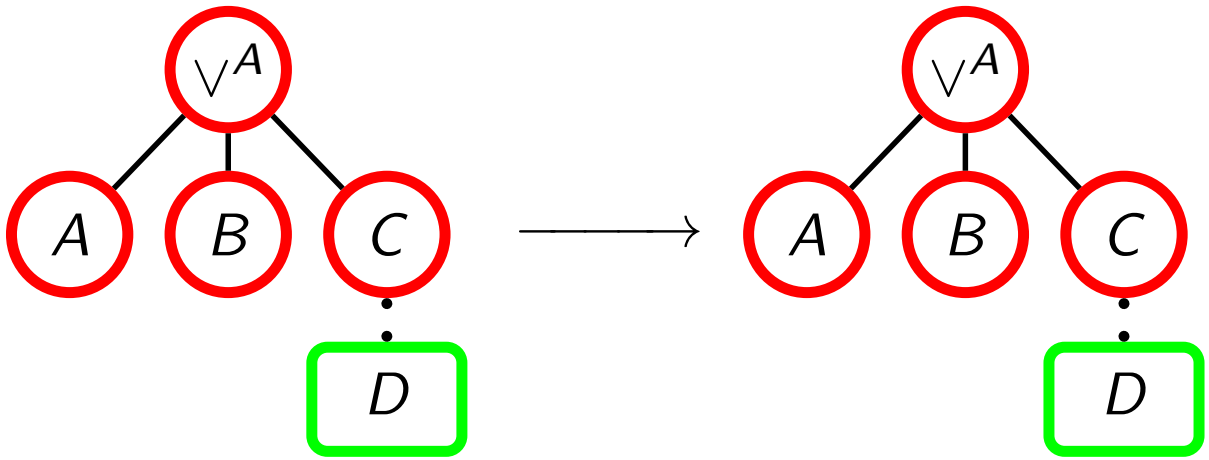


The Issue of the Assignment for the Non-Owner

Average cost of the attacker with serial execution, when all defenses are in place.

Basic assignment: $A = 1$; $B = 2$; $C = 3$; $D = \infty$

$\vee^A = avg$; $c^A = +$



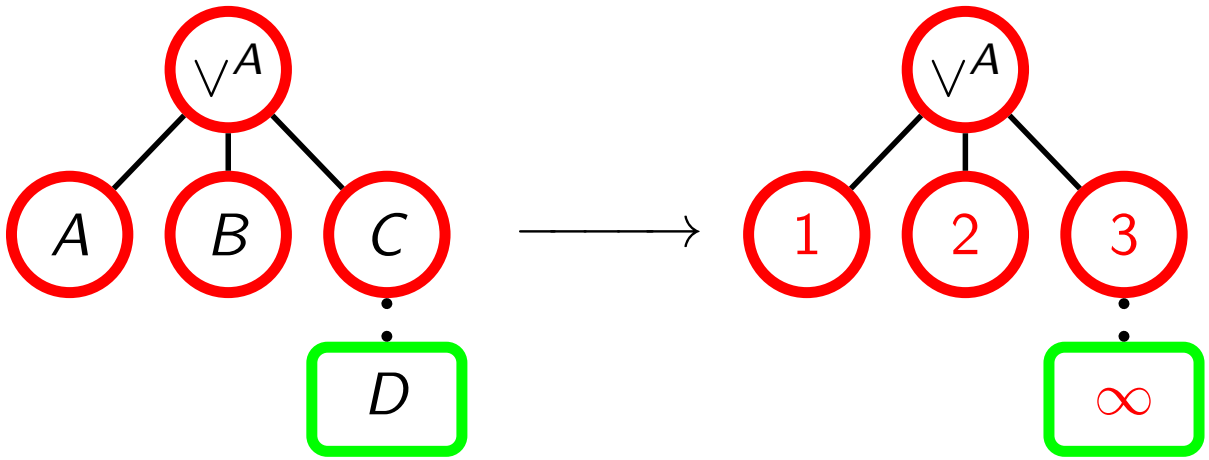
Average / Expected costs?

The Issue of the Assignment for the Non-Owner

Average cost of the attacker with serial execution, when all defenses are in place.

Basic assignment: $A = 1$; $B = 2$; $C = 3$; $D = \infty$

$\vee^A = avg$; $c^A = +$



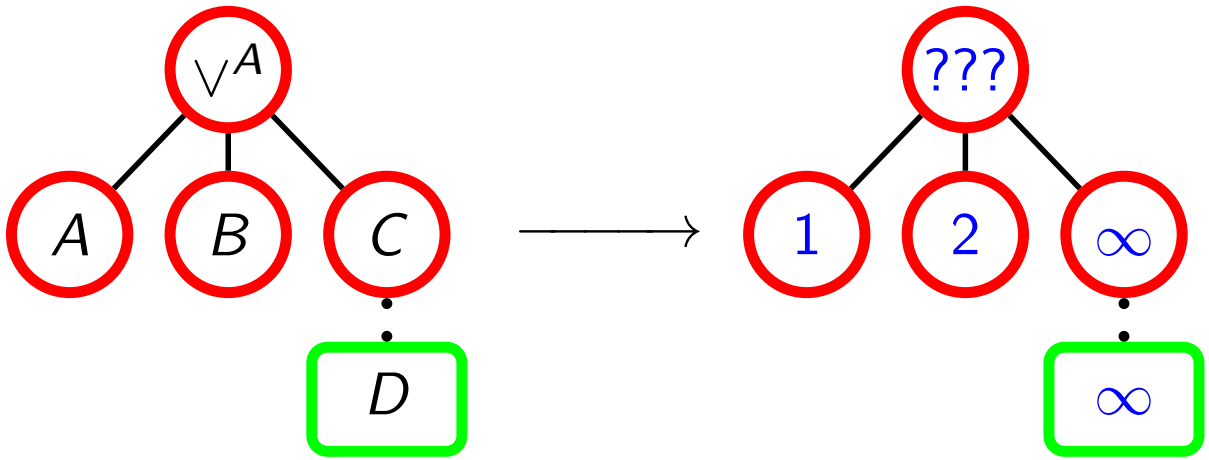
Average / Expected costs?

The Issue of the Assignment for the Non-Owner

Average cost of the attacker with serial execution, when all defenses are in place.

Basic assignment: $A = 1$; $B = 2$; $C = 3$; $D = \infty$

$\vee^A = avg$; $c^A = +$



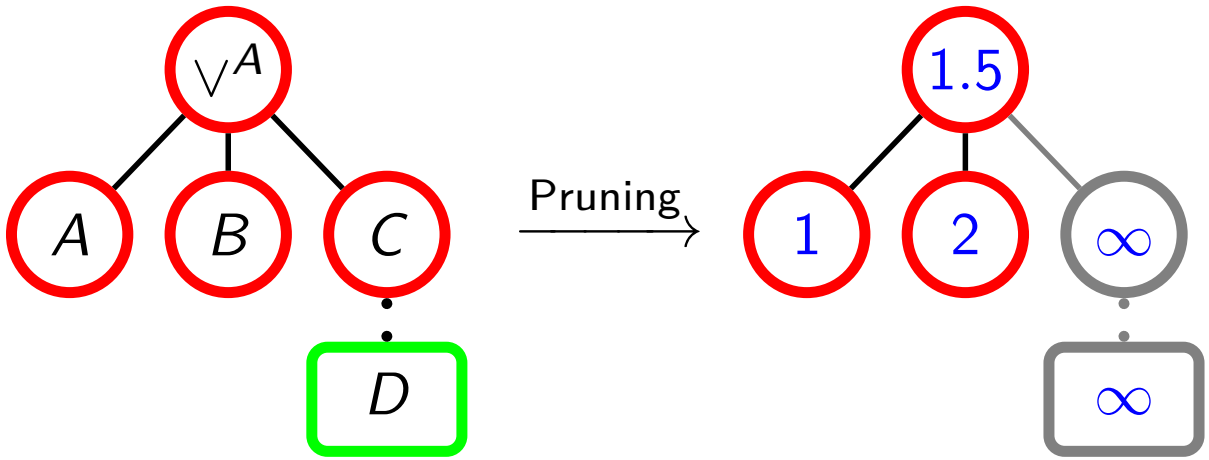
Average / Expected costs?

The Issue of the Assignment for the Non-Owner

Average cost of the attacker with serial execution, when all defenses are in place.

Basic assignment: $A = 1$; $B = 2$; $C = 3$; $D = \infty$

$\vee^A = avg$; $c^A = +$



Average / Expected costs?

Pruning

To overcome this inconsistency, we can

1. prune graphically,
2. prune mathematically, or
3. ensure that assigned element of non-owner is neutral element w.r.t. c^A and absorbing w.r.t. c^D.

Pruning Algorithm

Pruning

Starting from a leaf of the non-owner, we traverse the tree towards the root until we reach a node v such that

- v is a node of the owner and part of a proper[a] disjunctive refinement;
- v is a node of the non-owner and part of a proper conjunctive refinement;
- v is a node of the owner that counteracts a refined node of the non-owner.

The subtree rooted in node v is removed from the ADTree. The procedure is repeated, starting from all leaves of the non-owner.

[a] A refinement is called proper if it contains at least two refining nodes.

Pruning Theorems

Theorem (Semiring)

When $(D, \mathrm{op}_1, \mathrm{op}_2)$ *forms a semiring, pruning is unnecessary; The pruned and the unpruned tree yield the same results.*

Theorem (Pruning equivalence)

Graphical and mathematical pruning are equivalent.

uni.lu UNIVERSITÉ DU LUXEMBOURG

Pruning Summary

Message to take home

Pruning is removing unsuccessful subtrees of the owner of a question.

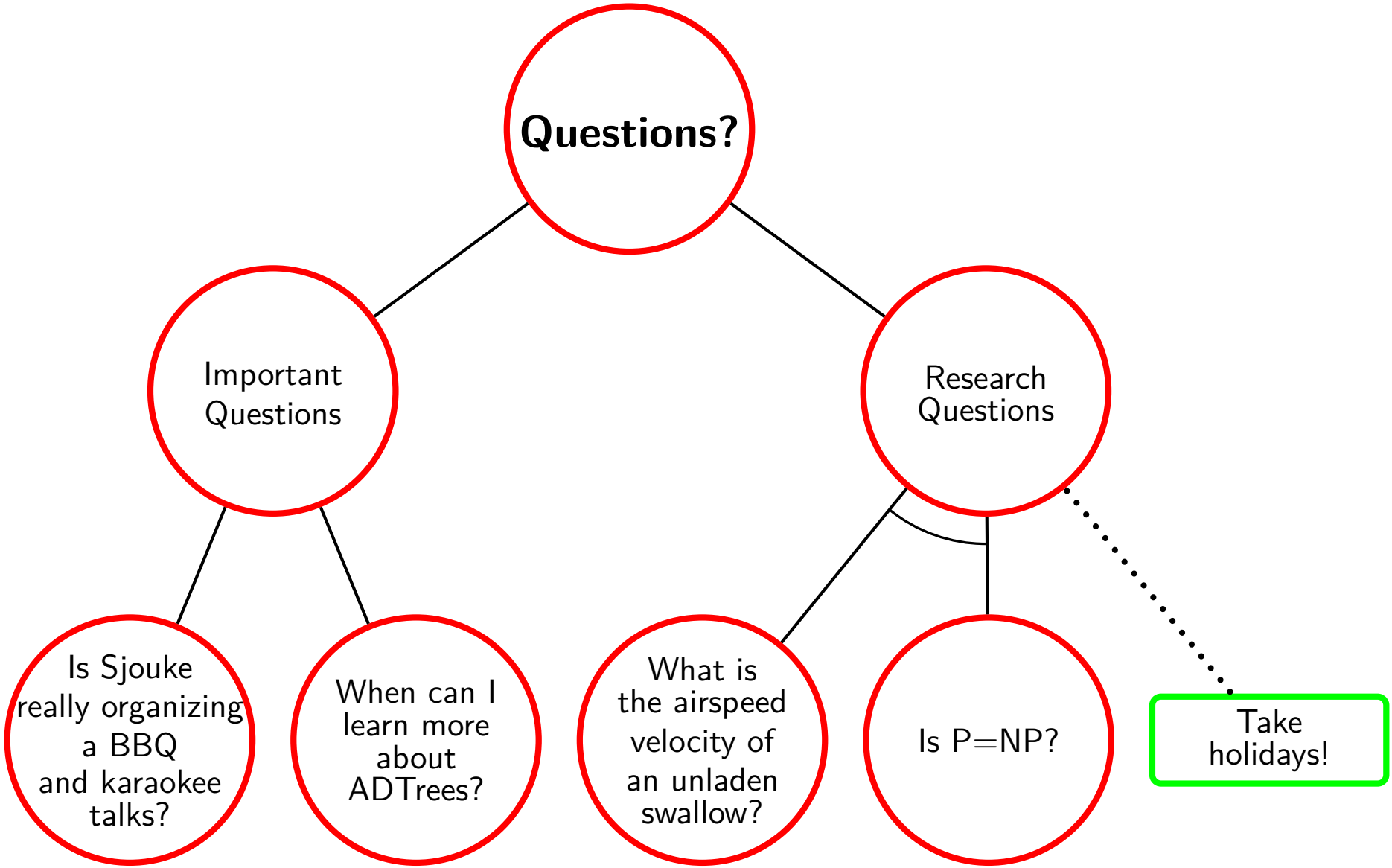
uni.lu
UNIVERSITÉ DU LUXEMBOURG

Summary

1. Attack–Defense Trees
2. Quantitative Analysis
3. Questions vs. Attribute Domains
   - Attributes with reference to one player (Class 1)
   - Attributes with reference to current player (Class 2)
   - Attributes with reference to third party (Class 3)
4. Pruning

uni.lu UNIVERSITÉ DU LUXEMBOURG

The End


Questions?
Important Questions
Research Questions
Is Sjouke really organizing a BBQ and karaokee talks?
When can I learn more about ADTrees?
What is the airspeed velocity of an unladen swallow?
Is P=NP?
Take holidays!

References


[FAST'10] Kordy, Mauw, Radomirović and Schweitzer
Foundations of Attack–Defense Trees. In Proceedings of FAST 2010, volume 6561 of LNCS. Springer 2011.


[GameSec'10] Kordy, Mauw, Melissen and Schweitzer
Attack-defense trees and two-player binary zero-sum extensive form games are equivalent. In Proceedings of GameSec 2010, volume 6442 of LNCS. Springer, 2010.


[SIIS'11] Kordy, Pouly and Schweitzer
Computational Aspects of Attack–Defense Trees. In Proceedings of SIIS 2011, volume 7053 of LNCS. Springer, 2011.


[JLC'12] Kordy, Mauw, Radomirović and Schweitzer
Attack–Defense Trees. Journal of Logic and Computation, 2012.


[IJSSE'12] Bagnato, Kordy, Meland and Schweitzer
Attribute Decoration of Attack–Defense Trees. International Journal of Secure Software Engineering, Special Issue on Security Modeling, 2012.