On Attack/Defense Trees

Patrick Schweitzer
SaToSS, Faculty of Sciences, Communication and Technology
University of Luxembourg

November 17th 2009
Outline

1. Intuition and overview of existing approaches to model attacks
2. Attack Trees
3. The new approach to include defenses
4. Future work
Intuition and overview

Intuition

Get money (illegally)

- Rob a bank
- Steal from ATM
- Hack into computer system

- Rob a store
- Enter with a gun
- Enter disguised

- Go to loan shark

- Enter at night
Intuition and overview

Intuition

Get money (illegally)

- Rob a bank
- Steal from ATM
- Hack into computer system

Get money from a bank

- Rob a store
- Enter with a gun

Enter disguised

Go to loan shark

Enter at night
Intuitions and overview

Guide to modeling attacks

Intuitive start: A mindmap (a special graph)

Problem: Complexity

Solution: Computer support (requires formalism)

Literature: Several approaches
Guide to modeling attacks

Intuitive start: A mindmap (a special graph)

Problem: Complexity

Solution: Computer support (requires formalism)

Literature: Several approaches
Intuition and overview

Guide to modeling attacks

Intuitive start: A mindmap (a special graph)

Problem: Complexity

Solution: Computer support (requires formalism)

Literature: Several approaches
Guide to modeling attacks

Intuitive start: A mindmap (a special graph)

Problem: Complexity

Solution: Computer support (requires formalism)

Literature: Several approaches
Intuition and overview

Different approaches to modeling attacks

- **Attack Trees**
  Essentially all information is contained in the leaves.

- **Attack Graphs or Attack Nets**
  Finite automata that fulfill security properties; separation of data and processes

- **Security Pattern Descriptions**
  Documents that describe in words the possible attacks on a system. They are very long exactly like this text which should never have been on the slide because nobody that listens to the talk reads that much text.

...
Intuition and overview

Different approaches to modeling attacks

- **Attack Trees**
  Essentially all information is contained in the leaves.

- **Attack Graphs or Attack Nets**
  Finite automata that fulfill security properties;
  separation of data and processes

- **Security Pattern Descriptions**
  Documents that describe in words the possible attacks on a system. They are very long exactly like this text which should never have been on the slide because nobody that listens to the talk reads that much text.

...
Attack Trees
Essentially all information is contained in the leaves.

Attack Graphs or Attack Nets
Finite automata that fulfill security properties; separation of data and processes

Security Pattern Descriptions
Documents that describe in words the possible attacks on a system. They are very long exactly like this text which should never have been on the slide because nobody that listens to the talk reads that much text.

...
1. Intuition and overview of existing approaches to model attacks

2. Attack Trees

3. The new approach to include defenses

4. Future work
Attack Trees

Attack Trees - the concept

Attack: How to get free food?
Attack Trees - the concept

Attack: How to get free food?
Attack Trees

Attack Trees - the concept

Attack: How to get free food?

Free food

Eat 'n' run

Pretend to work at restaurant
Attack Trees

Attack Trees - the concept

Attack: How to get free food?

- Free food
  - Eat 'n' run
  - Pretend to work at restaurant
  - Order meal
  - Sneak out
  - Ask Chef to prepare
  - Salami attack
Attack Trees

Attack Trees - the concept

Attack: How to get free food?

Free food

Eat 'n' run

Order meal

Sneak out

Pretend to work at restaurant

Ask Chef to prepare

Wait on customers

Steal part of their food

Salami attack

Sneak out

7/23
Attack Trees

Attack Trees - the concept

Attack: How to get free food?

Essentially a set of multisets, e.g.:

\{
  \{\{\text{Order meal, sneak out}\}\},
  \{\text{Ask Chef to prepare}\},
  \{\text{Wait on customers, steal part of their food, sneak out}\}\}

Properties of the existing model

Important properties of Attack Trees

- Uses **and** and **or** nodes
- Simple normal form: trees of depth 1
- Attributes can be attached to the leaves: then the attribute can be calculated for the root
- Projection only works for some attributes (Projection = Restriction of an attribute)
Important properties of Attack Trees

- Uses and and or nodes
- Simple normal form: trees of depth 1
  - Attributes can be attached to the leaves:
    then the attribute can be calculated for the root
- Projection only works for some attributes
  (Projection $=$ Restriction of an attribute)
Important properties of Attack Trees

- Uses and and or nodes
- Simple normal form: trees of depth 1
- Attributes can be attached to the leaves: then the attribute can be calculated for the root
- Projection only works for some attributes (Projection = Restriction of an attribute)
Important properties of Attack Trees

- Uses and and or nodes
- Simple normal form: trees of depth 1
- Attributes can be attached to the leaves: then the attribute can be calculated for the root
- **Projection** only works for some attributes
  (Projection = Restriction of an attribute)
Attack Trees

Including a defense in the framework

Free food

Eat 'n' run

Order meal

Sneak out

Pretend to work at restaurant

Ask Chef to prepare

Wait on customers

Steal part of their food

Salami attack

Sneak out
Attack Trees

Including a defense in the framework
Consider the Defense Tree ‘law enforcement’ instead of a policeman.

Consider the Attack Tree ‘Mafia’ attached to law enforcement.

and so on...

New framework: Attack Tree - Defense Tree - Attack Tree - ...
Consider the Defense Tree 'law enforcement' instead of a policeman.

Consider the Attack Tree 'Mafia' attached to law enforcement.

and so on...

New framework: Attack Tree - Defense Tree - Attack Tree - ...
Consider the Defense Tree ‘law enforcement’ instead of a policeman.

Consider the Attack Tree ‘Mafia’ attached to law enforcement.

and so on...

New framework: Attack Tree - Defense Tree - Attack Tree - ...
The new approach to include defenses

1. Intuition and overview of existing approaches to model attacks

2. Attack Trees

3. The new approach to include defenses

4. Future work
The new approach to include defenses

The general idea: two functions describing the nodes

Structure: rooted tree \( T = (V, E, r, \tau, \phi) \)
(non-empty, finite, directed, connected, acyclic, rooted)
Type: \( \tau: V \rightarrow \{\bigcirc, \Box, \Diamond\} \)  Connector \( \phi: V \rightarrow \{\lor, \land, \neg, \lnot\} \)
The new approach to include defenses

The general idea: two functions describing the nodes

Structure: rooted tree $T = (V, E, r, \tau, \phi)$
(non-empty, finite, directed, connected, acyclic, rooted)
Type: $\tau: V \rightarrow \{\bigcirc, \square, \lozenge\}$
Connector $\phi: V \rightarrow \{\vee, \wedge, \neg, \neg\}$

\[\tau(v) \in \{\bigcirc, \square\} \implies \tau(w) \in \{\tau(v), \lozenge\}\] (1)
\[\tau(v) \in \{\bigcirc, \square\} \text{ and } |\text{Children}_v| > 1 \iff \phi(v) \in \{\vee, \wedge\}\] (2)
\[\tau(v) \in \{\bigcirc, \square\} \text{ and } |\text{Children}_v| \leq 1 \iff \phi(v) = \neg\] (3)
\[\tau(v) = \lozenge \implies \tau(w) \in \{f(v), \lozenge\}\] (4)
\[\tau(v) = \lozenge \implies |\text{Children}_v| = 1\] (5)
\[\tau(v) = \lozenge \iff \phi(v) = \neg\] (6)

$v, w \in V$ and $(v, w) \in E$
The new approach to include defenses

The additional properties
The new approach to include defenses

The additional properties

Property (1):
\[ \tau(v) \in \{ \bigcirc, \blacksquare \} \implies \tau(w) \in \{ \tau(v), \lozenge \} \]
The new approach to include defenses

The additional properties

Property (2): 
\(\tau(v) \in \{\bigcirc, \Box\} \text{ and } |\text{Children}_v| > 1\)
\(\iff\)
\(\phi(v) \in \{\lor, \land\}\)
The new approach to include defenses

The additional properties

Property (3):
\( \tau(v) \in \{\bigcirc, \Box\} \) and \( |\text{Children}_v| \leq 1 \)
\( \iff \)
\( \phi(v) = \neg \)
The new approach to include defenses

The additional properties

Property (4):
\( \tau(v) = \Diamond \quad \Rightarrow \quad \tau(w) \in \{ f(v), \Diamond \} \)
The new approach to include defenses

The additional properties

Property (5):
\[ \tau(v) = \Diamond \implies |\text{Children}_v| = 1 \]
The new approach to include defenses

The additional properties

Property (6):
\[ \tau(v) = \Diamond \iff \phi(v) = \neg \]
The new approach to include defenses

Semantics of the Adtrees

Semantics of the adtree:
Unique variable associated to leaf

\[
\llbracket v \rrbracket = \begin{cases} 

v & \text{if } v \in L(T), \\
\bigvee_{w \in \text{Children}_v} \llbracket w \rrbracket & \text{if } \phi(v) = \bigvee, \\
\bigwedge_{w \in \text{Children}_v} \llbracket w \rrbracket & \text{if } \phi(v) = \bigwedge, \\
\neg \llbracket w \rrbracket & \text{if } \phi(v) = \neg \text{ and } \text{Children}_v = \{w\}, \\
\neg \llbracket w \rrbracket & \text{if } \phi(v) = \neg \text{ and } \text{Children}_v = \{w\}. 
\end{cases}
\]
The new approach to include defenses

Logical formulas associated to trees

Propositional logic corresponding to the tree:

\[
((\neg(D_1 \land (D_2 \land D_3 \land (\neg A_1)))))) \lor
(A_2 \land (A_3 \land A_4 \land (\neg(D_4 \lor D_5)))) \lor
(A_5 \lor A_6)
\]
The new approach to include defenses

Trees in normal form

Normal form:
\[ A_1 \lor A_5 \lor A_6 \lor \neg D_1 \lor \neg D_2 \lor \neg D_3 \lor (A_2 \land A_3 \land A_4 \land \neg D_4 \land \neg D_5) \]
The new approach to include defenses

Exemplary transformation: Distributivity $\land$ to $\lor$

With $k \geq 1$ and $l \geq 2$
The new approach to include defenses

Full set of transformation rules

- Distributivity \((A \lor B) \land C \rightarrow (A \land C) \lor (B \land C)\)
- 1–level absorption \((A \land B) \lor A \rightarrow A\)
- 2–level absorption as above
- Double negation \(\neg \neg A \rightarrow A\)
- Empty refinement no formula
- Associativity \((\lor \text{ and } \land)\) \((A \lor B) \lor C \rightarrow A \lor B \lor C\)
- De Morgan \((\lor \text{ and } \land)\) \(\neg (A \lor B) \rightarrow \neg A \land \neg B\)
- Idempotency \((\lor \text{ and } \land)\) \(X \lor X \rightarrow X\)
The new approach to include defenses

Full set of transformation rules

• **Distributivity** \((A \lor B) \land C \rightarrow (A \land C) \lor (B \land C)\)
• 1–level absorption \((A \land B) \lor A \rightarrow A\)
• 2–level absorption as above
• Double negation \(\neg \neg A \rightarrow A\)
• Empty refinement no formula
• Associativity \((\lor \text{ and } \land)\) \((A \lor B) \lor C \rightarrow A \lor B \lor C\)
• De Morgan \((\lor \text{ and } \land)\) \(\neg (A \lor B) \rightarrow \neg A \land \neg B\)
• Idempotency \((\lor \text{ and } \land)\) \(X \lor X \rightarrow X\)
The new approach to include defenses

Full set of transformation rules

- **Distributivity** 
  \((A \lor B) \land C \rightarrow (A \land C) \lor (B \land C)\)

- **1–level absorption** 
  \((A \land B) \lor A \rightarrow A\)

- **2–level absorption** 
  as above

- **Double negation** 
  \(\neg \neg A \rightarrow A\)

- **Empty refinement** 
  no formula

- **Associativity (\(\lor\) and \(\land\))** 
  \((A \lor B) \lor C \rightarrow A \lor B \lor C\)

- **De Morgan (\(\lor\) and \(\land\))** 
  \(\neg(A \lor B) \rightarrow \neg A \land \neg B\)

- **Idempotency (\(\lor\) and \(\land\))** 
  \(X \lor X \rightarrow X\)
The new approach to include defenses

Currently working on

Proving the uniqueness of the normal forms

Requires:

• Strong termination (Patrick - almost finished)
  Applying rules indefinitely is not possible

• Local confluence (Barbara - finished)
  Order of applying the rules leads to same result
The new approach to include defenses

Currently working on

Proving the uniqueness of the normal forms

Requires:

- Strong termination (Patrick - almost finished)
  Applying rules indefinitely is not possible
- Local confluence (Barbara - finished)
  Order of applying the rules leads to same result
The new approach to include defenses

Currently working on

Proving the uniqueness of the normal forms
Requires:  ● Strong termination (Patrick - almost finished)
            Applying rules indefinitely is not possible
            ● Local confluence (Barbara - finished)
            Order of applying the rules leads to same result
The new approach to include defenses

Termination function

Termination function:
A function from the trees into a totally ordered set, s.t. the value before applying a transformation rule > the value after applying a transformation rule.
The new approach to include defenses

Termination function

Termination function:
A function from the trees into a totally ordered set, s.t. the value before applying a transformation rule $>$ the value after applying a transformation rule.

Whiteboard
Future work

1. Intuition and overview of existing approaches to model attacks
2. Attack Trees
3. The new approach to include defenses
4. Future work
Future work

Work on generalizing the framework

- Introduce attributes to the leaves
- Allow directed acyclic graphs
- Consider temporal order of children
- Check out the two existing software packages
- ...
Summary

1. Intuition and overview of existing approaches to model attacks
2. Attack Trees
3. The new approach to include defenses
4. Future work