Faculty of Mathematics, Physics and Informatics
Comenius University,
Mlynská dolina, 842 48 Bratislava

CS&P Workshop 2021

Attack trees with Time Constraints

Aliyu Tanko Ali,
Damas Gruska
{aliyu.ali, gruska}@fmph.uniba.sk

27 - 09 - 2021

A real-time attack on systems require a real-time measure to stop (or reduce the damage) the attack.

Plan of Talk

Attack Trees

Challenges

Time constraints

Attack trees to timed automata

Conclusion

Definition

Attack trees are conceptual diagrams showing how an asset, or target, might be attacked [B. Schneier].

Graphical representation of ontology that :

Definition

Attack trees are conceptual diagrams showing how an asset, or target, might be attacked [B. Schneier].

Graphical representation of ontology that :

- capture i.e., identifying root cause of threats (attacks)

Definition

Attack trees are conceptual diagrams showing how an asset, or target, might be attacked [B. Schneier].

Graphical representation of ontology that :

- capture i.e., identifying root cause of threats (attacks)
- classify i.e., shows the breakdown of threats and how they can be achieve

Definition

Attack trees are conceptual diagrams showing how an asset, or target, might be attacked [B. Schneier].

Graphical representation of ontology that :

- capture i.e., identifying root cause of threats (attacks)
- classify i.e., shows the breakdown of threats and how they can be achieve
- share i.e., study how to repel the attacks.

Tree structure

An attack tree is constructed in a bottom-up manner where values at the leaves of the attack will be propagated towards the root node of the tree.

- Root node represent the attack target

Tree structure

An attack tree is constructed in a bottom-up manner where values at the leaves of the attack will be propagated towards the root node of the tree.

- Root node represent the attack target
- internal nodes (child nodes) are conditions which must be satisfied to reach the root, and

Tree structure

An attack tree is constructed in a bottom-up manner where values at the leaves of the attack will be propagated towards the root node of the tree.

- Root node represent the attack target
- internal nodes (child nodes) are conditions which must be satisfied to reach the root, and
- leaf nodes are atomic actions that need to be carried out to initiate the attack
- nodes (except for leaves) are associated with gates refinement e.g *AND* and *OR* gates (can be easily extended to show sequential actions *SAND*,*SOR*)

Attack tree example 1

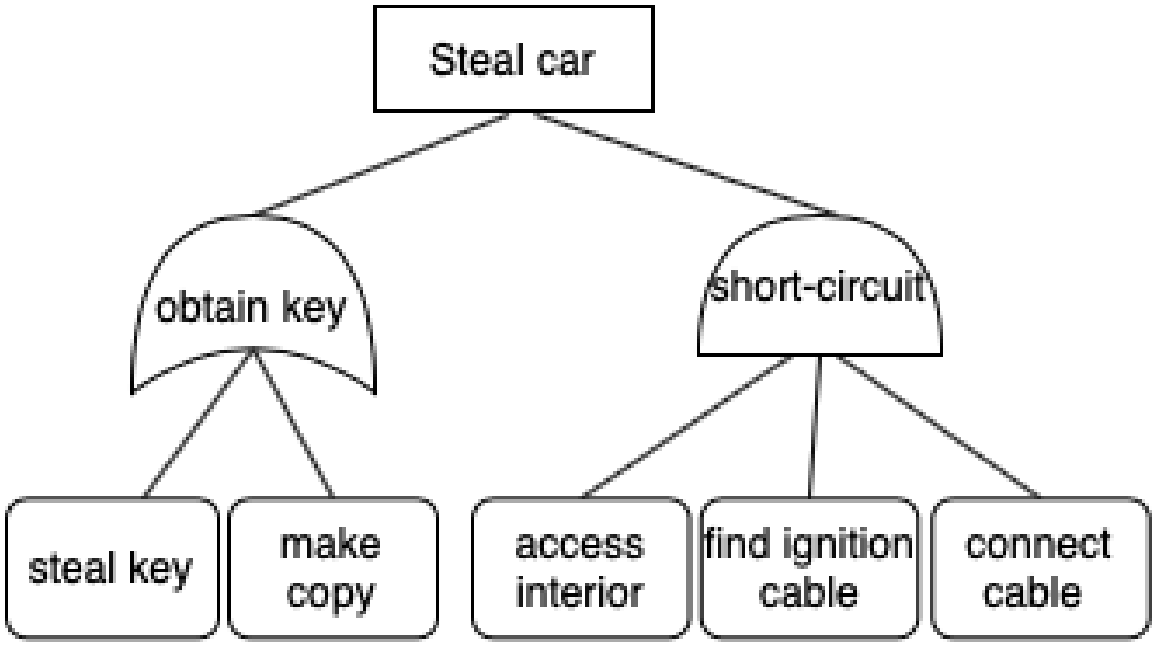


Figure: A simple example of attack tree

State of the art

From the literature, we already know that

- we can identify potential ways a system may be compromise [B. Schneier]

State of the art

From the literature, we already know that

- we can identify potential ways a system may be compromise [B. Schneier]
- attacks degree of success can be calculated [R. Kumar et. al.]

State of the art

From the literature, we already know that

- we can identify potential ways a system may be compromise [B. Schneier]
- attacks degree of success can be calculated [R. Kumar et. al.]
- study and identify the possible attack paths that can be easily exploited [S. Haque et. al.]

State of the art

From the literature, we already know that

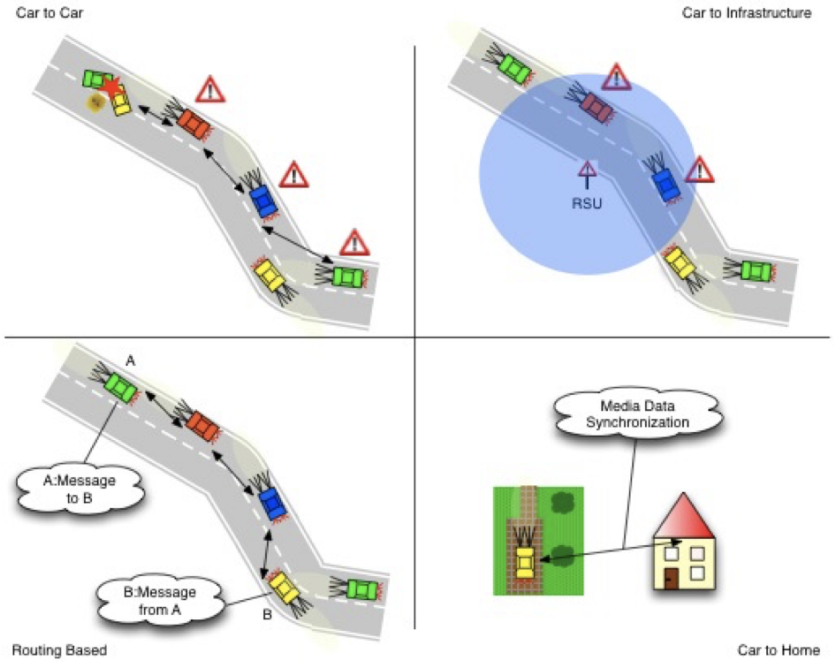
- we can identify potential ways a system may be compromise [B. Schneier]
- attacks degree of success can be calculated [R. Kumar et. al.]
- study and identify the possible attack paths that can be easily exploited [S. Haque et. al.]
- attack-defence and countermeasures to prevent attack from happening [Kordy, B et. al.]

Dynamic threat environment

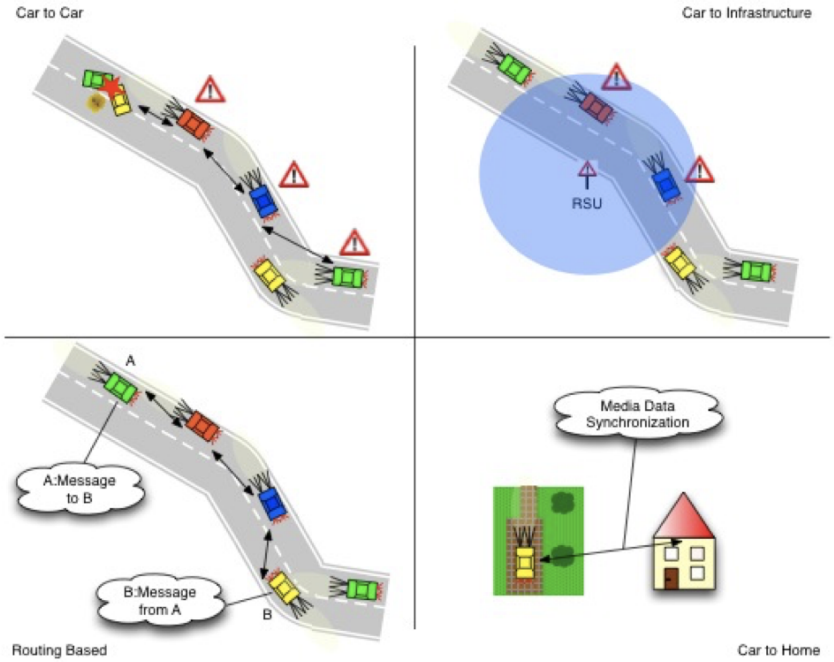
Dynamic threat environment

Most assets are no longer stand-along i.e., interacts with other set of systems and objects from the physical environment, e.g CPS.

Attack trees analysis
problems

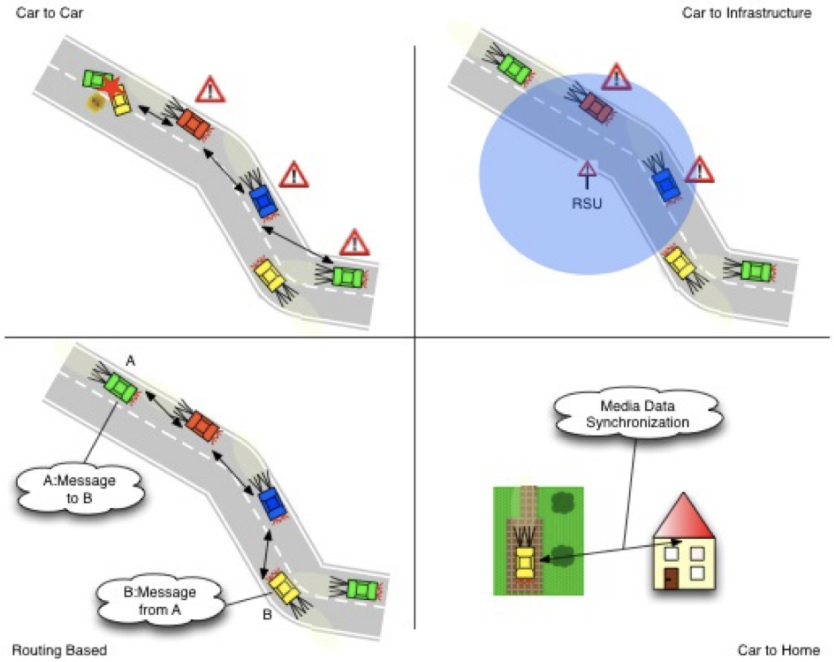


Attack trees analysis
problems



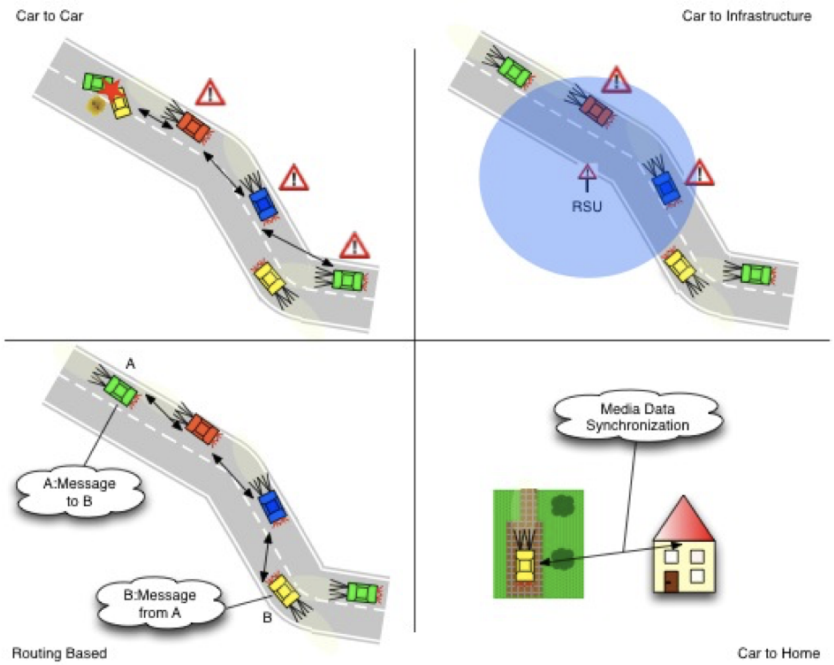
- AT are static i.e. can only capture threats at a specified instance

Attack trees analysis
problems



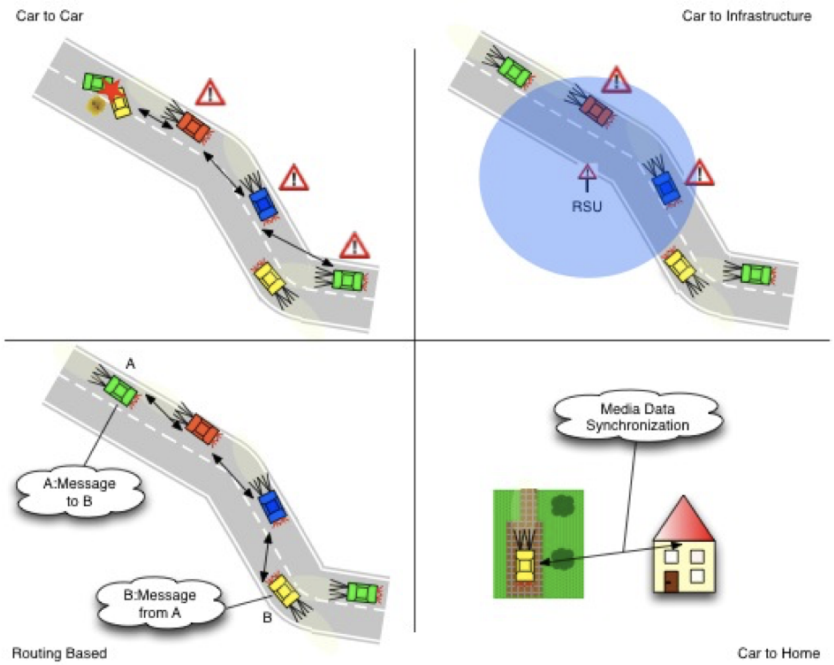
- AT are static i.e. can only capture threats at a specified instance
- Do not capture external threats i.e. threats from physical environment

Attack trees analysis
problems



- AT are static i.e. can only capture threats at a specified instance
- Do not capture external threats i.e. threats from physical environment
- Cannot evolve as the threat environment of an asset changes over time

Attack trees analysis
problems



- AT are static i.e. can only capture threats at a specified instance
- Do not capture external threats i.e. threats from physical environment
- Cannot evolve as the threat environment of an asset changes over time

Proposed approach

- Introduced attack time to actions
- extended the gates refinement with a set of time constraints

Proposed approach

- Introduced attack time to actions
- extended the gates refinement with a set of time constraints

Let T_r be an attack tree, we defined an attack as as a mapping $\mathcal{A} : \mathcal{Q}_L \rightarrow (Act \times T) \cup \{Nil\}$ where

- $\mathcal{Q}_L$ is a set of leaf nodes
- Act is a set of attack actions
- T is a set of attack time

We say that $\mathcal{A}$ is a simple attack if $\mathcal{A}(l, t) \neq Nil$ only for one leaf node.

Time constraints

Let τ be a set of time constraints that is represented by a constant pair $< b, f >$, with b marking the start of an attack and f marking the (expected) end of attack on the parent node such that $b, f \in \mathbb{Q}$ and $b \leq f$.

A set of gates $\mathcal{G} : \{q_0\} \cup \mathcal{Q}_S \rightarrow \{AND, OR\} \times \tau$.

Time constraints

Let τ be a set of time constraints that is represented by a constant pair $< b, f >$, with b marking the start of an attack and f marking the (expected) end of attack on the parent node such that $b, f \in \mathbb{Q}$ and $b \leq f$.

A set of gates $\mathcal{G} : \{q_0\} \cup \mathcal{Q}_S \rightarrow \{AND, OR\} \times \tau$.

Given an attack tree T_r, a set of attack time T, and a set of attack actions Act. An attack over T_r can only succeed;

Time constraints

- for *AND* node - if for all simple attacks $\mathcal{A}_1 \ldots \mathcal{A}_n$ such that $t_1 \ldots t_n$ are less than or equal to the constant time interval on the node,
- for *OR* node - if there exists a simple attack such that its attack time is less than or equal to the constant time interval on the node.

Timed automata

Attack trees via (weighted) timed automata

- Let T_r be an attack tree and τ be a set of time constraints, we proposed how to formally define some properties of the tree via weighted timed automata.

  E.g how to check if an attack can succeed when the attack time exceeded the constant time interval

Timed automata

Attack trees via (weighted) timed automata

- Let T_r be an attack tree and τ be a set of time constraints, we proposed how to formally define some properties of the tree via weighted timed automata.

  E.g how to check if an attack can succeed when the attack time exceeded the constant time interval

- this properties can be checked and verified using *UPPAAL*.
- We translated the tree into a parallel composition of weighted timed automata (WTA).

UPPAAL translation

- The sets of nodes of the attack trees are translated to a set of locations in the weighted timed automata.
- For each leaf node in the attack tree, we have a WTA that represents a linear path from the leaf to the root node.
- Altogether there are as many WTAs as the leaf nodes in the attack tree. Each location that represents a leaf which has a clock that is activated when there is an attack in the corresponding leaf in the tree.
- An attack on a node in the tree represents an enabled transition in the WTA. Initially, clocks become active when events synchronized, and end with either a success or fail synchronization action.

UPPAAL analysis

- If a location that represent a the sub_goal of the attack tree has an *OR* gate, a single event (i.e. action) together with a clock is activated and a transition to the target location is enabled
- otherwise all the events (i.e., *AND* gate) needs to be activated, and the target location can only be reached if all elapsing time in the clocks is less than (predefined) *max* (that represent the constant time intervals).

Discussion and future work

- We discussed how attack trees can be extended with a set of attack time, and a set of time constraints at the gates refinement
- we show how this extensions can be modelled and analyse using WTA
- We plan to study how actions observations, which cannot be identified as vulnerabilities with attack trees can be prevented from aiding an attack.

Thank you!