A Stochastic Framework for Quantitative Analysis of Attack-Defense Trees

R. Jhawar K. Lounis S. Mauw

CSC/SnT
University of Luxembourg
Luxembourg

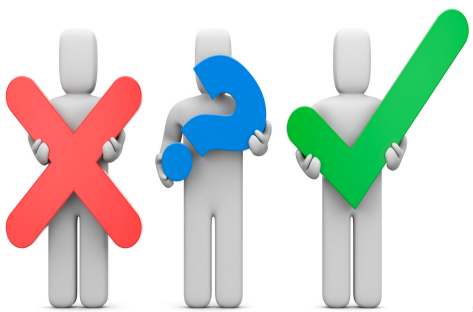
Security and Trust of Software Systems, 2016
ADT2P & TREsPASS Project

Plan

Cyber attacks nowadays

Cyber attacks are becoming more and more: Complex, Organized, Distributed and Sophisticated.

Their impact therefore is sometimes weighty, in some cases not tolerable.



Graphical security models

To fend of cyber attacks negative impact, research efforts have come with the development and design of security models:

Graphical security models

Attack trees: A tree-based model for cyber attacks representation. Introduced by Schneier in 1999.

Attack graphs: A directed graph-based model for cyber attacks representation.

Attack countermeasures trees: A tree-based model to graphically represent attacks and defenses in the same layout.

Attack-defense trees: Extend the attack tree model with **refinable** countermeasures. Introduced by Kordy *et al.* in 2010.

Quantitative analysis of security models

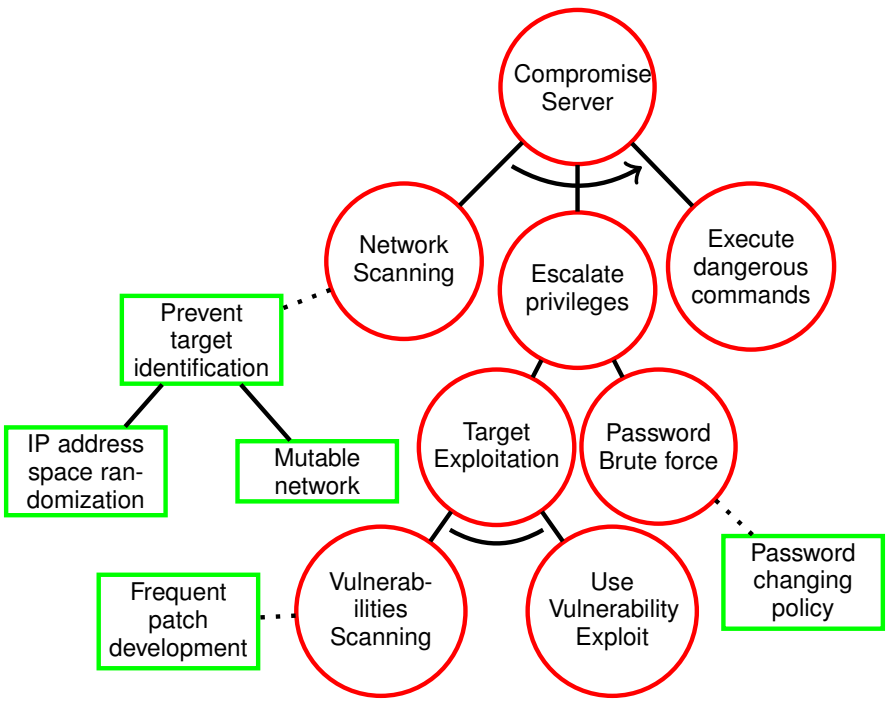
- How? Quantitative analysis is performed either by the use of analytical approach relying on Baysian Networks, Petri Nets, Markov chains or simulations such as Discrete simulation, Monte Carlo simulation.
  - By? Computing metrics or attributes like : Probability of an attack or a scenario in a given time, cost of the attacks, efficiency of countermeasures, mean time to breach a system, the most probable scenario, ...
- Why? Perform quantitative analysis which will help to reduce the risk and the negative impact of cyber attacks.

ADTrees

- What is it? Graphical methodology.
- Used for? Security scenario representation.
- Ancestor? Attack Trees.
- Interpretation: Can be seen as game between two players (proponent vs opponent).
- Semantics: Multisets, De Morgan lattice, Equational, Propositional, Series-Parallel graphs.
- Practice: Used in industry.

ADTrees

Graphically:



ADTrees

Definition 1

ADTrees are defined by means of an abstract syntax called ADTerms, typed-terms over the signature $\Sigma = (\mathbb{S}, \mathbb{F})$, where :

- $\mathbb{S} = \{p, o\}$ is the set of types of players.
- $\mathbb{F} = \{(\vee_k^p)_{k\in\mathbb{N}}, (\wedge_k^p)_{k\in\mathbb{N}}, (\overrightarrow{\wedge}_k^p)_{k\in\mathbb{N}}, (\vee_k^o)_{k\in\mathbb{N}}, (\wedge_k^o)_{k\in\mathbb{N}}, (\overrightarrow{\wedge}_k^o)_{k\in\mathbb{N}}, c^p, c^o\} \cup \mathbb{B}^p \cup \mathbb{B}^o$ is a set of function symbols.

Definition 2

ADTrees are closed-terms over the signature $\Sigma = (\mathbb{S}, \mathbb{F})$, and generated by the following grammar, where $b^s \in \mathbb{B}$ and $s \in \mathbb{S}$:

$$t :\equiv b^s \mid \vee^s(t, \ldots, t) \mid \wedge^s(t, \ldots, t) \mid \overrightarrow{\wedge}^s(t, ..., t) \mid c^s(t, t)$$

ADTrees

Examples of ADTerms :

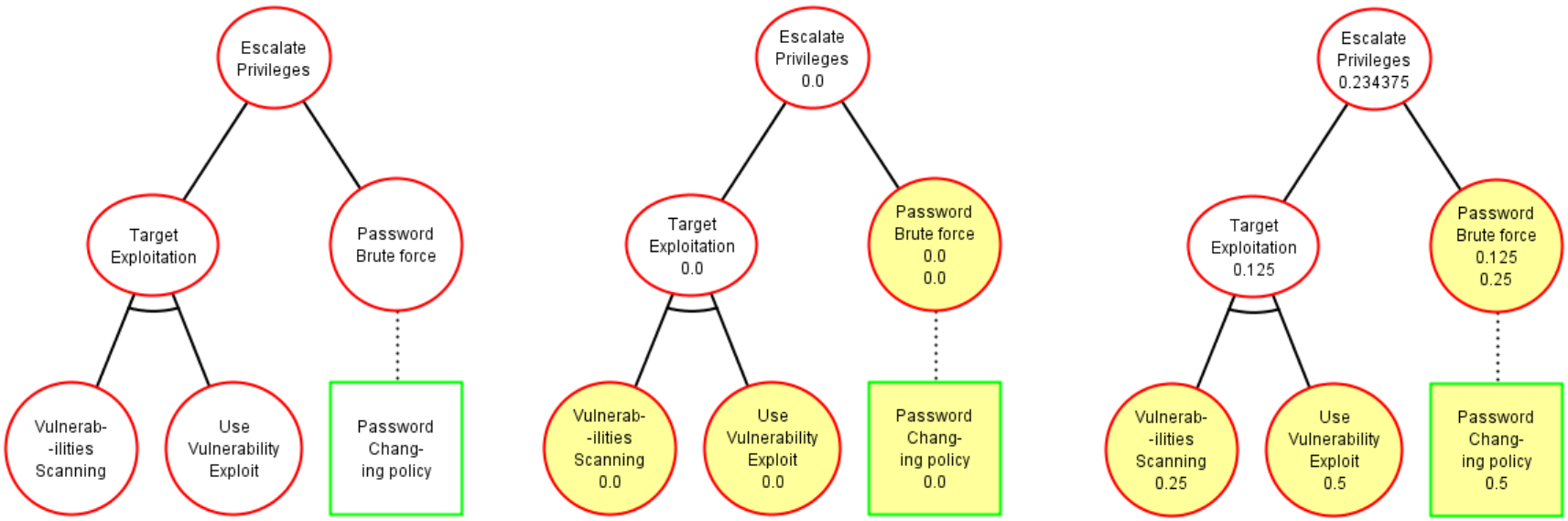
- $t_0 = b^p$ (Basic event)
- $t_1 = \vee^p(b_0^p, t_0)$ (Disjunction refinement)
- $t_2 = \wedge^p(t_1, b_1^p)$ (Conjunction refinement)
- $t_3 = \overrightarrow{\wedge}^p(t_2, b_2^p, b_3^p)$ (Sequential Conjunction refinement)
- $t_4 = c^p(t_3, b_0^o)$ (Counter-defense)

ADTree Quantitative Evaluation

- The quantitative evaluation of an ADTree consists in assessing a set of attributes like:Probability, cost, or time.
- It is performed through the standard bottom-up procedure.

ADTree Quantitative Evaluation

Standard Bottom-up procedure (ADTool):



ADTree and need for a new semantics

However

1. The bottom-up procedure works only for independent events.
2. So far, there is only one approach [KPS14] [1] for quantitative analysis of ADTree with dependent actions.
3. Only discrete analysis can be done.

1. B. Kordy, M. Pouly, and P. Schweitzer. A probabilistic framework for security scenarios with dependent actions. In International Conference on Integrated Formal Methods, 256-271, 2014

ADTree and need for a new semantics

We need to develop a new semantics for ADTree. The new semantics should allow dependent events to occur, and provide modeling capabilities for defense in a more realistic way. It should also provide a continuous analysis method for ADTree evaluation.

ADTree and need for a new semantics

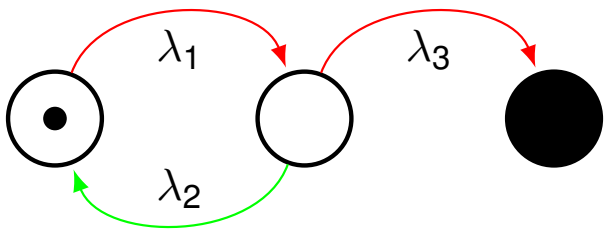
- We proposed to use : Continuous Time Markov Chain or CTMC as a new semantics for ADTree.
- We model attacks/defense execution using exponential distribution (good for delayed impact defenses).
- Using the analytical approach of CTMCs, we can evaluate several attributes, and perform a continuous analysis by the use of Cumulative Distribution Function.

Continuous Time Markov Chains

Definition 1

A Continuous Time Markov chain is a tuple (S, G, π), where:

- S is a finite disjoint set of states.
- $G\colon\ S \times S \longrightarrow \mathbb{R}$ is the infinitesimal generator matrix which gives the rate of transition between two states $s \in S$ and $s' \in S$.
- $\pi\colon S \longrightarrow [0, 1]$ is the initial probability distribution on S.



Continuous Time Markov Chains

Definition 2

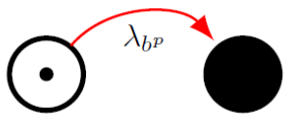
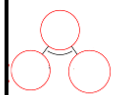
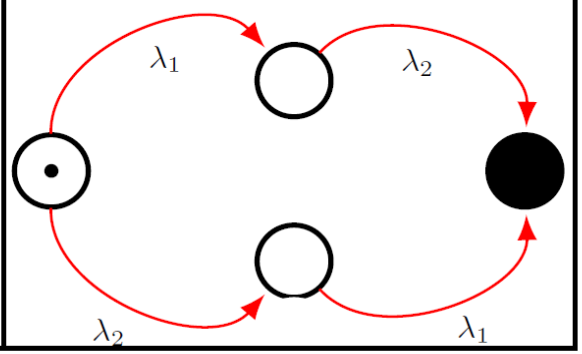
An explicit continuous time Markov chain M is a tuple (S, S_0, S_*, G), where:

- S is a finite disjoint set of states.
- $S_0 \subset S$ is a finite set of initial states.
- $S_* \subset S$ is a finite set of final states.
- $G : S \times S \longrightarrow \mathbb{R}$ is the infinitesimal generator matrix which gives the rate of transition between two states $s \in S$ and $s' \in S$.

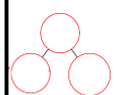
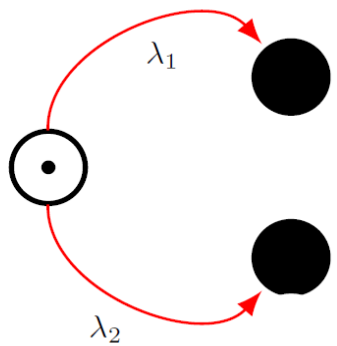
ADTree to CTMC

- We have formally defined the semantics of ADTrees in terms of CTMC for each component : Basic events, conjunction refinement, Disjunction refinement, Sequential conjunction refinement, and Countermeasure.

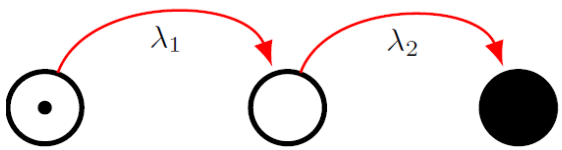
ADTree to CTMC

Element	Markov chain	Graphical
Basic event 	$\mathbb{M} = (\{s_0, s_*\}, \{s_0\}, \{s_*\}, G^{b^s})$ $G^{b^s} = \begin{bmatrix} -\lambda_{b^s} & \lambda_{b^s} \\ 0 & 0 \end{bmatrix}$	
Conjunction Refinement 	$\mathbb{M} = (\prod_{i=1}^{k} S^{b_i}, \prod_{i=1}^{k} S_0^{b_i}, \prod_{i=1}^{k} S_*^{b_i}, G^{\wedge^s_{k\in\mathbb{N}}})$ $G^{\wedge^s_{k\in\mathbb{N}}}(s_i, s_j) = \begin{cases} -\sum_{i\neq j} G^{\wedge^s_{k\in\mathbb{N}}}(s_i, s_j) & if \quad i = j \\ 0, & if \quad i \neq j \quad \wedge \quad \lvert s_i \Delta s_j \rvert > 2 \\ 0, & if \quad s_i \in \Omega \quad \wedge \quad \forall j \\ G^{idf}(S^{idf} \cap s_i, S^{idf} \cap s_j) & otherwise \end{cases}$	

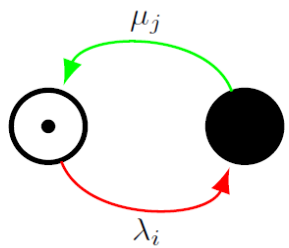
ADTree to CTMC

Element	Markov chain	Graphical
Disjunctive Refinement 	$\mathbb{M} = (S^{\vee^s_{k\in\mathbb{N}}}, S_0^{\vee^s_{k\in\mathbb{N}}}, S_*^{\vee^s_{k\in\mathbb{N}}}, G^{\vee^s_{k\in\mathbb{N}}})$ $S^{\vee^s_{k\in\mathbb{N}}} = S_0^{\vee^s_{k\in\mathbb{N}}} \cup S_*^{\vee^s_{k\in\mathbb{N}}} \cup S_{mid}^{\vee^s_{k\in\mathbb{N}}}$ $S_0^{\vee^s_{k\in\mathbb{N}}} = \prod_{i=1}^{k} S_0^{b_i}$ $S_*^{\vee^s_{k\in\mathbb{N}}} = \bigcup_{i=1}^{n} S_*^{b_i} \times \prod_{j\neq i} S_0^{b_j}$ $S_{mid}^{\vee^s_{k\in\mathbb{N}}} = \bigcup_{i=1}^{n} S_{mid}^{b_i} \times \prod_{j\neq i} S_0^{b_j}$ $G^{\vee^s_{k\in\mathbb{N}}}(s_i, s_j) = \begin{cases} -\sum_{i\neq j} G^{\vee^s_{k\in\mathbb{N}}}(s_i, s_j) & if \quad i = j \\ 0, & if \quad i \neq j \quad \wedge \quad \lvert s_i \Delta s_j \rvert > 2 \\ 0, & if \quad s_i \in \Omega \quad \wedge \quad \forall j \\ G^{idf}(S^{idf} \cap s_i, S^{idf} \cap s_j) & otherwise \end{cases}$	

ADTree to CTMC

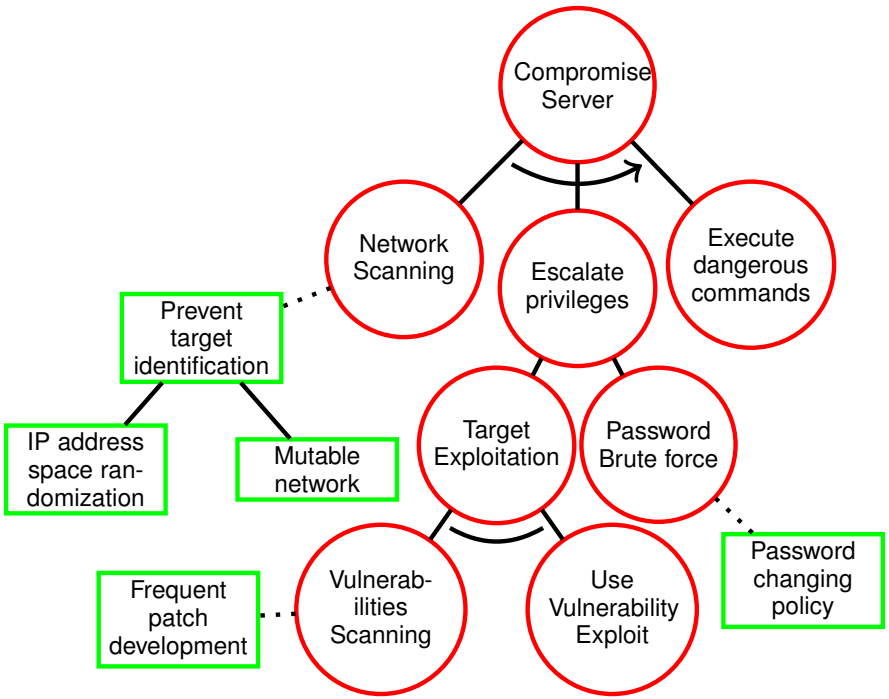
Element	Markov chain	Graphical
Sequential Conjunction Refinement 	$\mathbb{M} = (S^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}, S_0^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}, S_*^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}, G^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}})$ $- S^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} = S_0^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} \bigcup S_*^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} \bigcup S_{mid}^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}$ $- S_0^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} = \prod_{i=1}^{k} S_0^{b_i}$ $k\in\mathbb{N}$ $- S_*^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} = \prod_{i=1}^{k} S_*^{b_i}$ $- S_{mid}^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}} = \bigcup_{i=1}^{k-1} S_*^{b_i} \times S_0^{b_{i+1}} \bigcup S_{mid}^{b_i} \times S_0^{b_{i+1}} \bigcup S_*^{b_i} \times S_{mid}^{b_{i+1}}$ $G^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}(s_i, s_j) = \begin{cases} -\sum_{i\neq j} G^{\overrightarrow{\wedge^s}_{k\in\mathbb{N}}}(s_i, s_j), & \text{if } i = j \\ 0, & \text{if } i \neq j \wedge \lvert s_i \Delta s_j \rvert > 2 \\ 0, & \text{if } s_i \in \Omega \wedge \forall j \\ G^{idf}(S^{idf} \cap s_i, S^{idf} \cap s_j), & \text{otherwise} \end{cases}$	

ADTree to CTMC

Element	Markov chain	Graphical
Conter--measures 	$\mathbb{M} = (S^{c(M^s,M^{\bar{s}})}, S_0^{c(M^s,M^{\bar{s}})}, S_*^{c(M^s,M^{\bar{s}})}, G^{c(M^s,M^{\bar{s}})})$ – $S^{c(M^s,M^{\bar{s}})} = S_0^{c(M^s,M^{\bar{s}})} \cup S_{mid}^{c(M^s,M^{\bar{s}})} \cup S_*^{c(M^s,M^{\bar{s}})}$ – $S_0^{c(M^s,M^{\bar{s}})} = S_0^{M^s} \times (s_1^{M^{\bar{s}}}, s_2^{M^{\bar{s}}}, ..., s_{\lvert S_*^{M^{\bar{s}}}\rvert}^{M^{\bar{s}}})$ where $s_i^{M^{\bar{s}}} \in S_*^{M^{\bar{s}}}$ and $i \in \{1, ..., \lvert S_*^{M^{\bar{s}}}\rvert\}$ – $S_*^{c(M^s,M^{\bar{s}})} = S_*^s \times S_0^{M^{\bar{s}}}$ – $S_{mid}^{c(M^s,M^{\bar{s}})} = S_{mid}^{M^s} \times S_0^{M^{\bar{s}}} \cup S_{mid}^{M^{\bar{s}}}$ $G^{c(M^s,M^{\bar{s}})}(s_i, s_j) = \begin{cases} -\sum_{i \neq j} Q^{c(M^s,M^{\bar{s}})}(s_i, s_j), & if \quad i = j \\ \sum G^{M^s}(s_{i'}, s_{j'}) + \sum G^{M^{\bar{s}}}(s_{i''}, s_{j''}) & otherwise \end{cases}$	

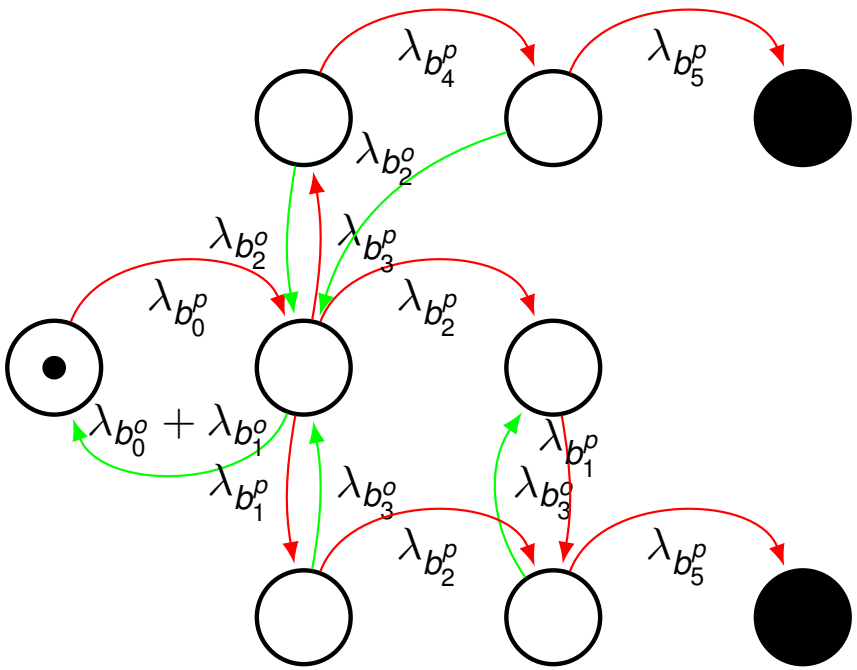
ADTree evaluation using CTMC

We took an example study:



ADTree evaluation using CTMC

We obtain a final CTMC representing the entire ADTree:

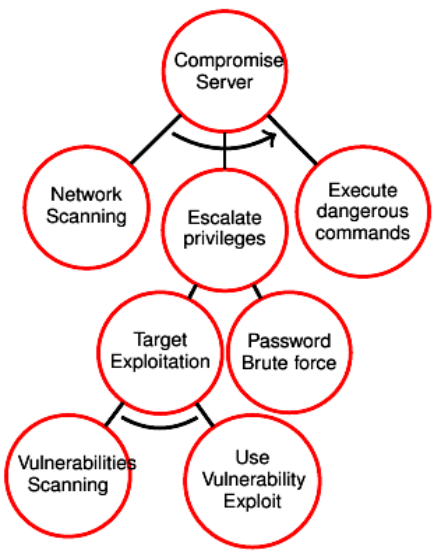


ADTree evaluation using CTMC

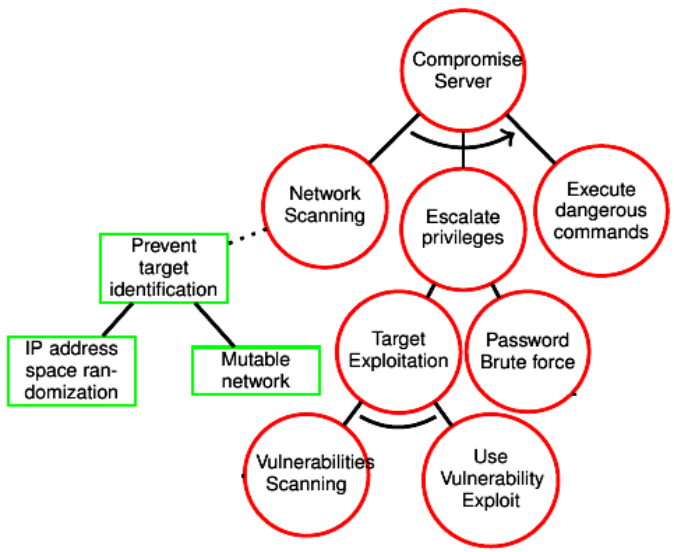
We have assessed three situational cases:

1. Attack tree (No defense is considered)
2. Adding countermeasure (Prevent target identification)
3. Adding the remaining countermeasures (Password policy, Frequent patches).

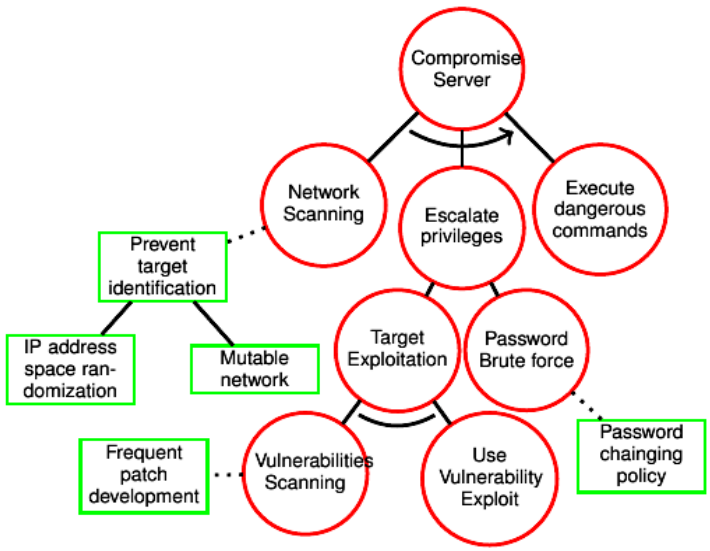
ADTree evaluation using CTMC



Case 1



Case 2



Case3

ADTree evaluation using CTMC

Analytical approach using CTMC is performed as follow:

$$G = \begin{bmatrix} -\lambda & \lambda \\ 0 & 0 \end{bmatrix}$$

infinitesimal generator

$$P = \begin{bmatrix} Q & R \\ 0 & Id \end{bmatrix}$$

mean probability transition matrix

$$N = (1 - Q)^{-1}$$

fundamental matrix

$$\mathrm{B} = N \times R$$

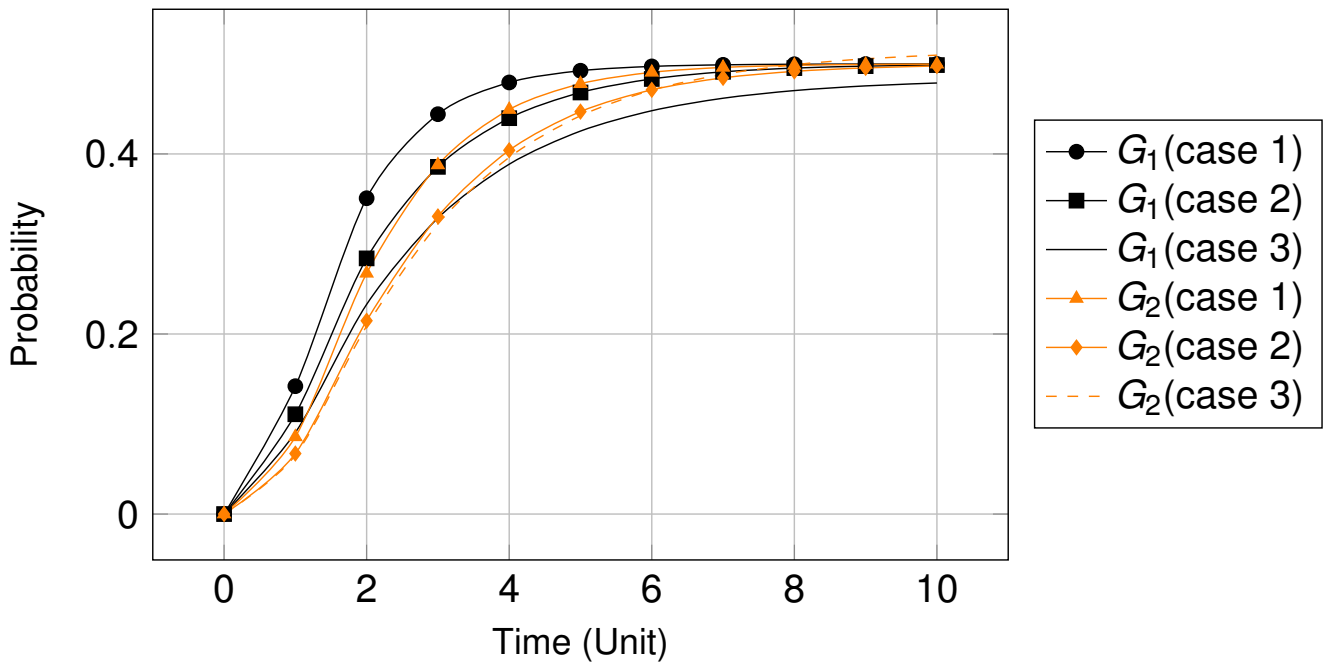
absorbency frequency matrix

$$P(t) = e^{G.t}$$

instantaneous probability matrix

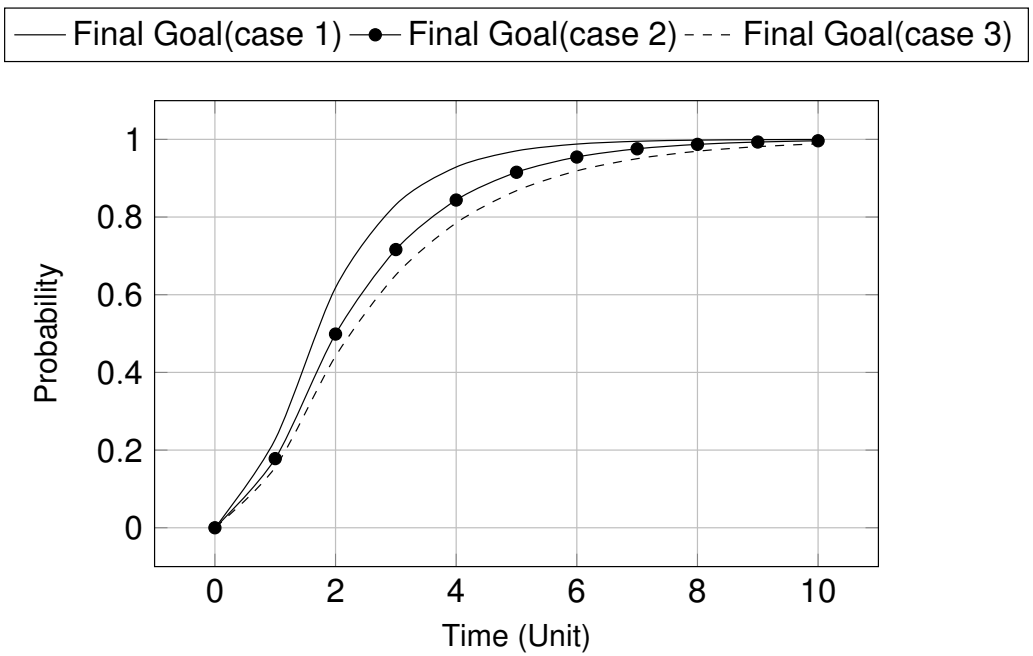
ADTree evaluation using CTMC

Probabilistic attributes: Probability of final states (black states) representing the final goals G_1 and G_2



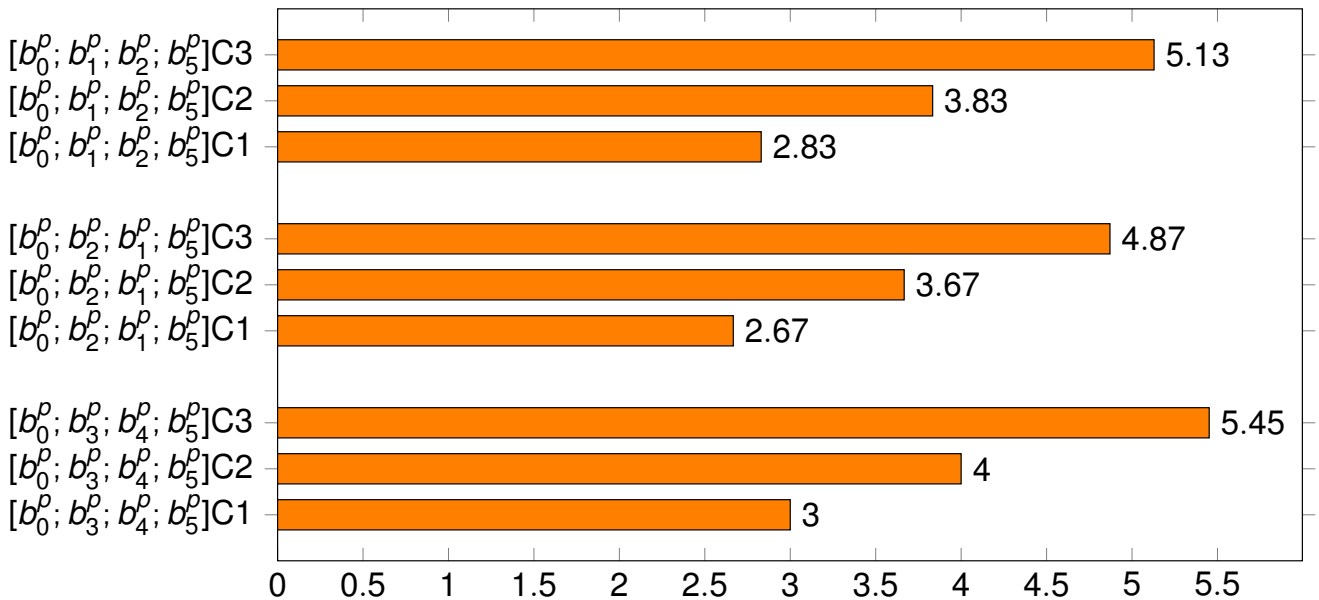
ADTree evaluation using CTMC

Probabilistic attributes: Probability of final states (black states) representing the final goal $G_1 + G_2$



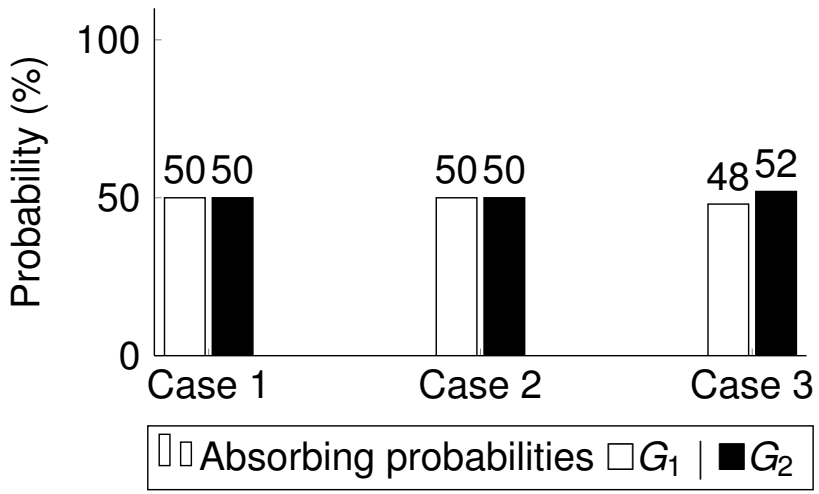
ADTree evaluation using CTMC

Probabilistic attributes: Expected number of steps for each scenario of G_1 and G_2



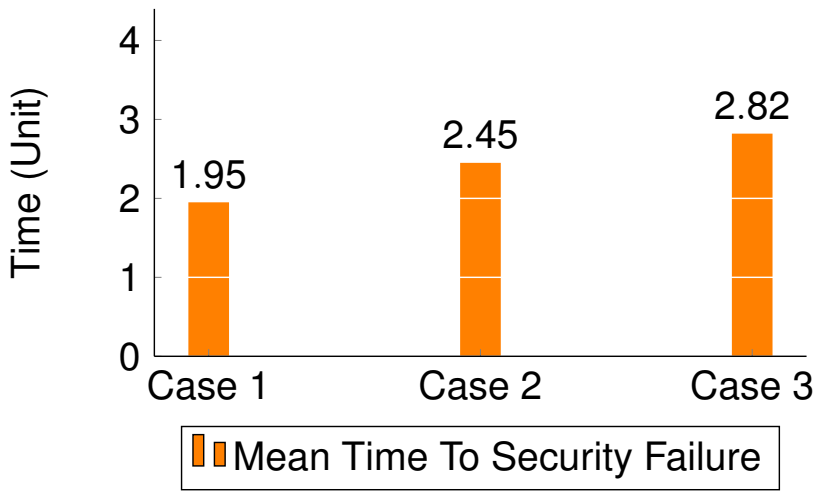
ADTree evaluation using CTMC

Probabilistic attributes: Absorbing probabilities for G_1 and G_2



ADTree evaluation using CTMC

Timed attributes: Mean time to security failure



Conclusions

- We proposed a new semantics for ADTrees in terms of CTMCs.
- We applied CTMC to perform quantitative analysis of ADTree with dependent actions.

Challenges and Future Work

Challenges :

- Not all attacks and/or countermeasures execution follow exponential-distribution.
- Estimating the rates for attacks/countermeasures has always been the main challenge for security assessment.

Future Work :

- Extend our framework in order to accurately model social attacks and complex behaviors laying on other distributions.
- Embed the framework within the ADTool software and make it more adaptable for real life security scenarios.

Thanks for your attention
Any questions ?