Learning Factor Graphs for Preempting Multi-Stage Attacks in Cloud Infrastructure

Phuong Cao, Zbigniew Kalbarczyk, Ravishankar Iyer

University of Illinois at Urbana-Champaign

1. INTRODUCTION

This paper presents statistical techniques for learning factor graphs for preempting multi-stage attacks in cloud infrastructure. We discuss methods for: i) learning parameters of multi-variate factor functions that capture relations among the events representing behavior of both a user and an attacker, and ii) construction of factor graphs to reason about an attack state with the purpose of preemptively detecting malicious activities. Our work is driven by real attacks reported in the wild.

In the context of this analysis, we focus on multi-stage attacks that can be represented by five distinct stages: initial compromise, host hopping, escalation of privilege, maintaining presence, and delivery of payloads. User and attacker activities are represented by events, which are derived from the log files collected at runtime by security monitoring tools, such as intrusion detection systems, network flows, and system logs.

In Figure 1, we present events corresponding to a multi-stage attack. This attack started with an illegitimate login to a user account via stolen passwords and ssh keys. While in the system, the attacker logged into a vulnerable node (events e1 and e2), obtained the root permission using a local privileged escalation exploit targeting Redhat Enterprise Linux, and deployed a rootkit (named Venom) in a memory-mounted directory */dev/shm* (event e3). The rootkit had two components that aimed at maintaining unauthorized and persistent access to the compromised system: i) a user-land binary that is an encrypted backdoor that receives remote commands to launch attack payloads, and ii) a Linux Loadable Kernel Module (LKM) that provides a stealth port knocking service for the user-land backdoor (events e4, e5, and e6). Port knocking is a stealthy technique to open a port only upon receiving a specific sequence of "knocking" packets. For example, the port 9090 on a compromised machine opens only after receiving a "knocking" sequence of three TCP packets such that: "TCP Source Port + Sequence Number = 1221". As a result, the compromised node can stay under the radar of regular network security scanning tools.

2. A FACTOR GRAPH MODEL

Modeling multi-stage attacks. Figure 1 shows a factor graph corresponding to the example multi-stage attack. The factor graph relates observed events to infer the state of the attack. A vector of five binary variable, s=[s_1,..,s_5], represents the attack stages . Our goal is to minimize manual efforts (as shown in our previous study [1]) involved in defining factor functions.

Learning. Based on past attack data from system logs, network flows, and incident reports, a set of factor function parameters θ are automatically learned using statistical techniques such as minimizing of cross-entropy between two probability distributions (Figure 1, Eq. 1). Instead of learning parameters for the complex joint probability distribution, it is sufficient to learn for parameters for cliques of random variables connected by a factor function [2]. For a given clique of the variables $(e1, s1, s2)$, the goal is to learn an approximate function $g(e1, s1, s2)$ given priors, i.e., two functions $f(s1)$ and $h(s2)$. Note that priors can be either defined by an expert or derived from the data. An intuitive technique is to minimize the Kullback-Leibler divergence among $g(e1, s1, s2)$, $f(s1)$, and $h(s2)$, using an iterative proportional fitting algorithm until convergence, i.e., $\varepsilon = |g_k - g_{k-1}|$ is small such that $\varepsilon \leq 10^{-3}$. The procedure repeated for k in $1..N$ iterations:

$$g_k(\mathrm{s1}, \mathrm{s2}) = g_{k-1}(s1, s2)\frac{f(s1)}{\sum_{s2} g_{k-1}(\mathrm{s1}, \mathrm{s2})}$$

Inference. Using the learned factor functions, it is possible to infer the current attack stage from observed data (Figure 1, Eq. 2) by maximizing the state vector using standard techniques such as Gibbs sampling or belief propagation [2]. Thus, an ongoing attack can be preemptively detected and a corresponding mitigation strategy can be applied.

3. CONCLUSION

We have shown that by using past data, it is possible to learn parameters of cliques in a factor graph. The factor graph is used to infer an attack state vector, which is critical to assessing the progress of an ongoing attack and determining a corresponding mitigation strategy.

REFERENCES

[1] Cao, Phuong, et al. "Preemptive intrusion detection." *Proceedings of the 2014 Symposium and Bootcamp on the Science of Security*. ACM, 2014.

[2] Koller, Daphne, and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

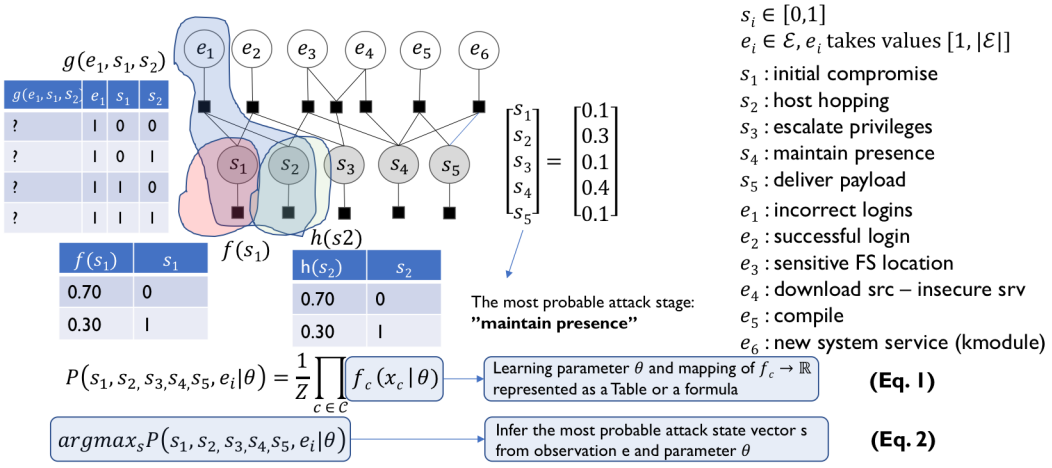


Figure 1. Factor graph model of an example multi-stage attack