Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Thesis
M.Sc.

Distributed Self-Learning Detection of Advanced Persistent Threats

Motivation

Advanced Persistent Threats (APT) are stealthy attacks that try to remain undetected as long as possible. Their goal is to thoroughly explore a network to either exfiltrate intellectual property or to disturb control processes of, e.g., industrial sites. APTs differ from “ordinary” attacks in their high sophistication and are more difficult to detect. A very strict method of securing a system against these types of attacks is restrictive whitelisting of network traffic. We define exactly which hosts, applications and services are allowed. Then, it is possible to specify expected network traffic by describing, e.g., which process communicates to whom, how often, at what points in time, etc. This behaviour can either be specified manually or learned by trainig traffic. At runtime the described behavior can be compared to the observed one to find anomalies. The goal of this work is to create an such an anomaly detection system which works upon a network topology and traffic specificatin and can then monitor a critical network during runtime.

In this setting, we assume that our network is partitioned into various functional domains using network virtualization technology like VPN, VLAN, etc. The compartmentalization of the network is beneficial for anomaly detection, as various distributed locations are created in the network that are suitable for network monitoring.

Your Task

Your work begins with a thorough familiarization with the necessary background (anomaly detection, modeling properties of networks, network virtualization, etc.) and related work in the field. You will then design (or use an existing) language to express properties of the target network, like, e.g., communication relations, traffic quantities, timing, etc. Furthermore you will design and implement anomaly detection components that can monitor the network and detect property changes.

For the evaluation of your solution, you are going to set up a demonstrator network together with services and clients. You will deploy your anomaly detection components throughout this network, e.g., on each gateway component. Afterwards you emulate malicious traffic reflecting ongoing attacks. Your anomaly detection system should be able to detect these attacks.

Requirements

Solid knowledge in computer networking is required for this thesis. Virtualization (Xen) and Python/Java programming skills are recommended.

Contact

Dr. Holger Kinkelin, Stefan Liebald and Marcel von Maltitz
{kinkelin, liebald, vonmaltitz}@net.in.tum.de

http://go.tum.de/376400