ACSAC 2020
December 7–11, 2020 • Online

Mimic the Whole Attack Chain: A First Look at Evasion against Provenance Graph based Detection

Zhenyuan Li[†] **(li_zhenyuan@qq.com)**, Runqing Yang[†] , Qi Alfred Chen[§] , Yan Chen[‡]
[†]Zhejiang University, [§]University of California, Irvine, [‡]Northwestern University

Introduction

Provenance graph based threat detection are widely studied as countermeasures against APT and other cyber threats for its powerful alert correlation capability. However, these detection approaches generally suffer from dependency explosion problem. And methods proposed to mitigate the problem pose underlying risks. In this poster, we first proposed a systemic mimicry attack approach against the underlying risks.

Then, we can generate adversary samples systematically. With these samples, we are able to provide a large dataset which can not only test the robustness of existing detection systems but also help with the design of new and more robust detection approaches.

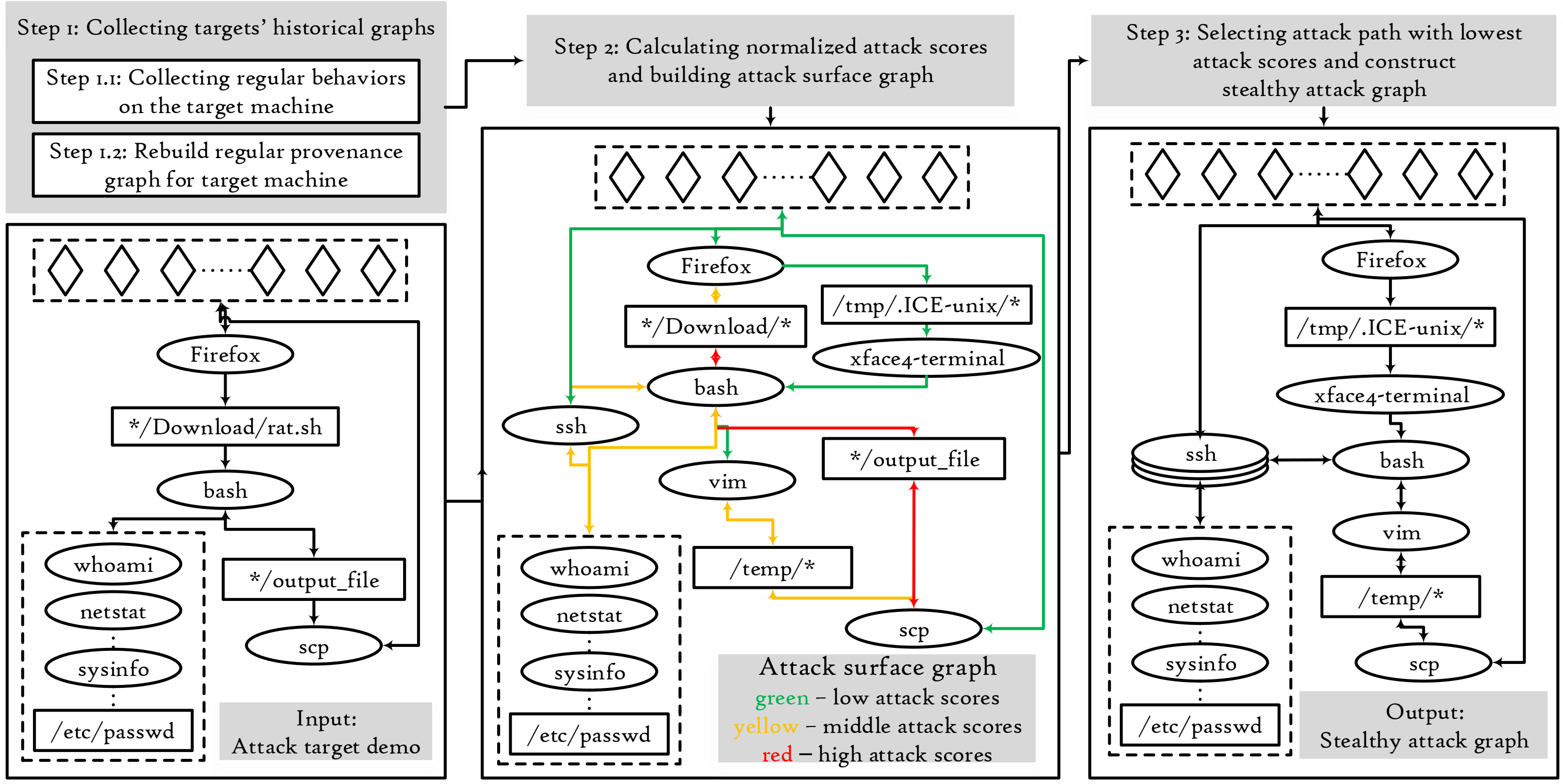
Background

1. System-level provenance graph based threat detection approaches prevalently suffer from dependence explosion problem.
   a) Backward/forward tracking b) Tag-based alert correlation
   c) Graph alignment based detection c) Anomaly detection
2. Mitigation approaches for dependence explosion problem bring the risk of mimicry attack.
   a) General data reduction b) Decay-based pruning c) Anomaly path ranking

Assumption

1. We only consider provenance graph based detection instead of other detection and combinations.
2. Attackers can collect enough regular behaviors data to cover target mimicry attack behaviors
3. Same behaviors share the similar provenance graphs on different machines.

Architecture and a Motivating Example


Step 1: Collecting targets' historical graphs
Step 1.1: Collecting regular behaviors on the target machine
Step 1.2: Rebuild regular provenance graph for target machine
Step 2: Calculating normalized attack scores and building attack surface graph
Step 3: Selecting attack path with lowest attack scores and construct stealthy attack graph
Firefox
*/Download/rat.sh
bash
whoami
netstat
sysinfo
/etc/passwd
*/output_file
scp
Input:
Attack target demo
Firefox
/Download/
/tmp/.ICE-unix/*
xface4-terminal
bash
ssh
vim
*/output_file
/temp/*
scp
whoami
netstat
sysinfo
/etc/passwd
Attack surface graph
green – low attack scores
yellow – middle attack scores
red – high attack scores
Firefox
/tmp/.ICE-unix/*
xface4-terminal
ssh
bash
vim
/temp/*
scp
whoami
netstat
sysinfo
/etc/passwd
Output:
Stealthy attack graph

Normalized Attack Scores

We can encode several critical properties, including stealthiness, feasibility, etc., as distances between nodes in an attack surface graph. Then we can find ideal attack chains by finding the shortest path in the attack surface graph.

The distance in the attack surface graph can be calculated with a normalized attack score. As the following equation shows:

$$\mathrm{NS}(s \rightarrow o) = \log\left(\frac{\mathrm{Freq}(s \rightarrow *)}{\mathrm{Freq}(s \rightarrow o)}\right) - \mathrm{Decay} + \mathrm{Feas}(s)$$

Where $\log\left(\frac{\mathrm{Freq}(s \rightarrow *)}{\mathrm{Freq}(s \rightarrow o)}\right)$ represents the normalized proportion of events from subject s to object o in all events start from subject s. And Decay and Feas(s) denote the incentives for extending the attack chain and the difficulties posed for extending the attack chain, respectively. The last two items can be ignored in the preliminary experiments for convenience.

Discussion and Future Work

As the pre-extracted attack surface graph shows, the original attack chain involves too many abnormal paths. And we can find much more stealthy attack chains as the output stealthy graph shows.

We plan to conduct further research and experiments on the feasibility of implementing real-world attacks and the effectiveness of bypassing existing detection systems.

We hope that we can generate more adversarial attack samples and use them to test the effectiveness and robustness of existing detection systems. Moreover, the generated samples can alleviate the problem of lack of samples from detection systems and thus help build better detection systems.