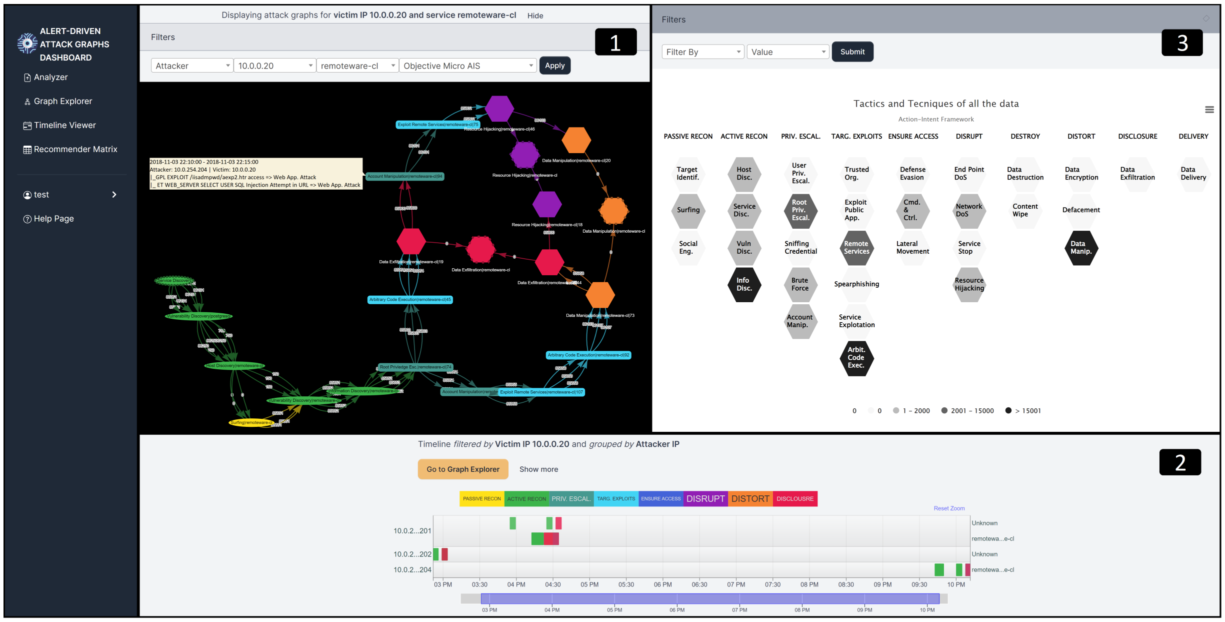
Critical Path Exploration Dashboard for Alert-driven Attack Graphs

Azqa Nadeem[*]
Delft University of Technology

Sònia Leal Díaz[†]
La Salle Ramon Llull University

Sicco Verwer[‡]
Delft University of Technology



2018-11-03 22:10:00 - 2018-11-03 22:15:00
Attacker: 10.0.254.204 | Victim: 10.0.0.20
|_GPL EXPLOIT /iisadmpwd/aexp2.htr access => Web App. Attack
|_ ET WEB_SERVER SELECT USER SQL Injection Attempt in URL => Web App. Attack

Figure 1: The critical path exploration dashboard for alert-driven attack graphs. Attacker strategies extracted from the uploaded intrusion alerts are visualized as a: (1) *Graph explorer* showing a unified view of all strategies, (2) *Timeline viewer* allowing analysts to analyze individual attacker actions, and (3) *Recommender matrix* highlighting the alerts that require urgent attention of an analyst.

1 Introduction

Analysts working in Security Operations Centers (SOC) are responsible for managing, triaging, and analyzing large volumes of intrusion alerts for the forensic analysis of security incidents. The process of reverse engineering attacker objectives and strategies from intrusion alerts is often manual and time-intensive. Extensive research has been conducted to reduce analyst workload by introducing visual analytics tools in their workflows that visualize large datasets, identify unique patterns, and provide actionable intelligence [2–4].

Recently, 'alert-driven attack graphs' (AGs) have been proposed as a novel paradigm of attack graphs that reverse engineer attacker strategies from the actions observed through intrusion alerts [3]. Nadeem et al. have developed a system called SAGE to generate these alert-driven AGs. SAGE is an interpretable, unsupervised, sequential learning pipeline that compresses thousands of intrusion alerts into AGs without a priori expert knowledge about existing vulnerabilities and network topology. It discovers temporal and probabilistic patterns in alert datasets for modeling their context (shown by so-called *state identifiers*) so as to differentiate between similar attacker strategies that lead to distinct outcomes. Individualized AGs are generated for every objective that is exploited on the victim host(s). Security analysts can triage and visualize AGs of interest for obtaining threat intelligence regarding potentially scripted attack attempts, attacker behavior dynamics, and fingerprintable paths.

While SAGE does the heavy lifting in terms of discovering and displaying attacker strategies, it is infeasible and time-consuming for security analysts to visualize each AG separately for finding global patterns. While the average complexity of SAGE AGs is lower than alternative modeling approaches, the AGs of relatively common objectives (e.g., data exfiltration on HTTP) can be significantly more complex with hundreds of vertices. The AGs are also not interactive, i.e., attack paths cannot be filtered by time or attack stage of interest. Preliminary interviews with security analysts revealed that such large graphs with no interaction capabilities are unhelpful in an operational setting. In addition, SAGE also does not prioritize critical attack paths[1] that might need the urgent attention of security analysts.

In this work, we develop a web-based visual analytics dashboard for alert-driven attack graphs with querying and prioritization capabilities for critical attack paths. The dashboard reduces analyst workload by i) discovering and extracting attacker strategies from intrusion alerts (done by the SAGE module), ii) consolidating the discovered attacker strategies in a dashboard with filtering capabilities, and iii) prioritizing critical events that might require an analyst's urgent attention. Figure 1 shows the three visualizations of the proposed dashboard: (1) *Graph explorer* presents a consolidated view of all attacker strategies discovered by SAGE, and interactively visualizes them in order to find relationships between attacker objectives. (2) *Timeline viewer* enables the analyst to investigate temporal correlations between attacker actions, and compare the tactics/techniques used by the attackers at different time stamps. (3) *Recommender matrix* shows a condensed version of the MITRE ATT&CK stages that assists the analyst in prioritizing critical events based on their prevalence and urgency in the alert dataset.

The proposed dashboard addresses the limitations of SAGE AGs in the following ways: The unified view provided by the graph explorer and timeline viewer helps analysts discover temporal re-

[*] e-mail: azqa.nadeem@tudelft.nl
[†] e-mail: sonialeal01011@gmail.com
[‡] e-mail: s.e.verwer@tudelft.nl

[1] A critical path is a series of actions that lead to critical event(s).

lationships between different objectives. The complexity of the global graph is managed via several filters. We opt for two different views for investigating attack paths (graph explorer) and attacker actions (timeline viewer) to reduce the cognitive load on analysts. The recommender matrix highlights critically urgent attack stages. Based on the unique circumstances of a SOC, analysts can adjust the threshold for what is considered *urgent*. In addition, analysts can filter the urgency of critical events based on specific attacker hosts, victim hosts, and targeted services.

2 Approach

The proposed dashboard is a Django application, implemented as a wrapper around SAGE, responsible for visualizing the attacker strategies extracted by SAGE. The dashboard has four pages (three visualizations and an *analyzer*), which are listed in the left pane for navigation, see Figure 1. Analysts can upload intrusion alert files in JSON format in the analyzer. This triggers the execution of the SAGE module, which processes the alerts, applies unsupervised sequence learning, extracts temporal and probabilistic patterns from the alerts, and produces state sequences that reflect attacker strategies (see [2] for details). The discovered attacker actions and their temporal relationships are then stored in a MySQL database that is later queried to populate the three visualizations, i.e., graph explorer, timeline viewer, and recommender matrix.

Graph explorer (GE). A global, unified view of all attacker strategies is shown in the graph explorer. Analysts can use filters to query for specific attack paths based on the attacker host, victim host, targeted service, and attack stage of interest. This view enables analysts to draw conclusions about the most frequently used pathways towards an objective, and the temporal inter-dependency between them. The GE is implemented using the `vis.js Network` chart that enables efficient interactions with large graphs. The nodes and edges in the graph represent the same information as the SAGE AGs, i.e., the nodes correspond to the attacker actions (showing the attack stage, targeted service, and state identifier), while the edges show the temporal relation between the nodes. A *node tool-tip* shows the time-ordered list of alert signatures caused by a specific attacker action for each attacker/victim pair.

Timeline viewer (TV). The timeline viewer enables analysts to focus on specific attacker actions that occurred during a user-selected time-window to determine, for instance, if a victim is targeted by numerous attackers at the same time. The timeline viewer is implemented using the `D3 timelines-chart`, which supports several *swimlanes*. Each swimlane corresponds to the various actions taken by an attacker on a victim using a specific service. The segments in a swimlane correspond to the nodes in the GE (i.e., attacker actions). The segment color corresponds to the action's attack stage. The swimlanes are either grouped by attacker or victim IP and can be filtered by time. Once the interesting actions are narrowed down, clicking on *'Go to Graph Explorer'* redirects to the GE and shows the corresponding attack paths.

Recommender matrix (RM). The recommender matrix shows a condensed version of the MITRE ATT&CK framework, similar to the ATT&CK Navigator[2], where the different attack stages are highlighted based on the prevalence and urgency of critical events. Analysts can click on one of the highlighted attack stages to be redirected to the GE, which shows only the attack paths that led to this critical attack stage (colored in white for emphasis). The recommender matrix is implemented using the `Highcharts Honeycomb tile map`. Each hexagonal tile in the tile map shows an attack stage from the Action-Intent Framework (AIF) [1], while its color shows how urgently it needs to be assessed (darker → more urgent). The tiles from left to right increase in their severity, i.e, the left tiles typically show the start of an attack and the right tiles show adversary objectives. In addition, there are four categories of urgency, the thresholds for which can be configured by the analyst according to the acceptable risk levels of their SOC.

2.1 Exemplary use-case

We populate the dashboard using intrusion alerts collected from a distributed multi-stage team-based attack scenario collected during a collegiate penetration testing competition (CPTC) in 2018[3]. The dataset contains a total of 330,270 Suricata alerts generated by 6 student teams over 9 hours. We utilize the intrusion alerts of team 5 and 9 (resulting in 104,152 alerts) to compare the proposed dashboard against SAGE AGs. SAGE compresses the alerts into 63 AGs, while the unified dashboard allows to explore and reason about the discovered attacker strategies. The recommender matrix shows that alerts associated to *Data Manipulation* (DM), *Arbitrary Code Execution* (ACE) and *Information Discovery* (ID) require urgent attention of security analysts due to their prevalence and severity scores. ID is being highlighted because of the sheer frequency of raised alerts.

Data exfiltration attempts. A security analyst can view the most common strategies used by attackers to exfiltrate data over HTTP by using the *Data Exfiltration* and *http* filters in the GE. All of the filtered pathways do *Root Privileged Escalation* and *Data Manipulation* before exfiltration. By hovering over one of the privilege escalation nodes, the tool-tip enumerates alert signatures, such as "GPL EXPLOIT CodeRed v2 root.exe access", providing actionable intelligence that the CodeRED exploit was used to carry out this attack. Making sure that the network is not vulnerable to CodeRED can help mitigate similar privilege escalation attacks in the future.

Attacks after working hours. A security analyst may seek to investigate if any anomalous activity occurred at specific times, such as after working hours or on holidays. The TV can be used to investigate what transpired on a victim host. For example, the TV shows that the victim 10.0.0.22 was targeted by the attacker 10.0.254.204 until 9:44:42 PM. The analyst can view the strategies employed during this time by redirecting to the GE. The GE illustrates that *Arbitrary Code Execution* enabled *Resource Hijacking*, which further enabled *Data Manipulation* and *Data Exfiltration*. The analyst can use this information to determine which resources were compromised during these attacks.

3 Discussion and Future work

The proposed dashboard provides a consolidated view of the attacker strategies discovered by SAGE. The GE and TV help discover global attack patterns, while the RM highlights the most critical events to analyze. The filtering capabilities in each visualization help analysts drill down to more targeted attacks and provide greater flexibility than SAGE did alone. Additionally, since the dashboard is a web application, analysts can investigate cyber threats remotely, without having to run SAGE on individual machines.

We are currently designing a qualitative study with security practitioners to evaluate the different aspects of the dashboard in comparison to SAGE AGs. In addition, we are investigating ways to improve the scalability of the dashboard, and the actionability of the attack stages (TTPs) in the RM for effective mitigation.

References

[1] S. Moskal and S. J. Yang. Framework to describe intentions of a cyber attack action. *arXiv preprint arXiv:2002.07838*, 2020.

[2] A. Nadeem, S. Verwer, S. Moskal, and S. J. Yang. Alert-driven attack graph generation using s-pdfa. *IEEE TDSC*, 19(2):731–746, 2021.

[3] A. Nadeem, S. Verwer, and S. J. Yang. Sage: Intrusion alert-driven attack graph extractor. In *2021 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 36–41. IEEE, 2021.

[4] A. Sopan, M. Berninger, M. Mulakaluri, and R. Katakam. Building a machine learning model for the soc, by the input from the soc, and analyzing it for the soc. In *2018 IEEE VizSec*, pp. 1–8. IEEE, 2018.

[2] `https://mitre-attack.github.io/attack-navigator/`

[3] `https://mirror.rit.edu/cptc/`