Authentication-Event Processing for Enhanced SOC Investigations

Seth Hastings
The University of Tulsa
seth-hastings@utulsa.edu

Tyler Moore
The University of Tulsa
tyler-moore@utulsa.edu

***Abstract*—Security Operations Centers (SOCs) receive thousands of security alerts each day, and analysts are responsible for evaluating each alert and initiating corrective action when necessary. Many of these alerts require consulting user authentication logs, which are notoriously messy and designed for machine use rather than human interpretability. We apply a novel methodology for processing raw logs into interpretable user authentication events in a university SOC dashboard tool. We review steps for data processing and describe views designed for analysts. To illustrate its value, we utilized the dashboard on a 90-day sample of alert logs from a university SOC. We present two representative alerts from the sample as case studies to motivate and demonstrate the generalized workflows. We show that enhanced data from the dashboard could be utilized to completely investigate over 84% of alerts in the sample without additional context or tools, and a further 13% could be partially investigated.**

I. Introduction

The number of digitally connected systems has rapidly increased as the transition to cloud services continues. This has expanded attack surfaces and the number of users and systems under active monitoring, which has led to a surge in the number of alerts that organizations receive daily. The average Security Operation Center (SOC) team now experiences thousands of alerts per day on average, which leads to 67% being ignored due to alert fatigue and a high volume of false positives [1].

This paper presents a methodology and technology for assisting analysts operating in a SOC. In particular, we describe a dashboard tool that ingests raw authentication logs and presents aggregated information to help analysts evaluate alerts more effectively. The underlying process for converting raw logs to actionable authentication event data was described in work previously published at WOSOC [6].

The core methodology underlying the SOC dashboard described in this paper was developed with the intention of creating an accurate measure of user authentication for research purposes. Once developed, alternative use cases were quickly recognized, including use in a SOC to assist analysts. This assistance was envisioned as filtering, aggregating, and enhancing the most relevant content into a form that is more easily parsed by the human analyst, presented alongside the alert under investigation. Thus, we hope to improve the plight of SOC analysts by providing a method for quick assessment of an alert through an intuitive presentation of enhanced authentication logs and meaningful derivatives.

The remainder of the paper is organized as follows. Section II reviews the literature on SOC operations related to authentication logs and their derivatives. Section III describes the construction of the dashboard and describes the types of data and artifacts available in each panel. Section IV provides example investigations of two alerts using real-life data from our 90-day sample, which are then generalized as dashboard workflows. Section V explores the types of alerts present in our sample, identifies those for which the dashboard provides sufficient data for an investigation, and calculates alert coverage using our sample. Finally, we conclude in Section VI.

II. Related Work

A growing topic of research is the tools and techniques used to handle large volumes of alerts, which include AI tools for detection and alert prioritization [8]. Tilbury and Flowerday elaborate on the high number of alerts SOC's receive, noting that the reported amount is the result after automated security tools have been applied [14]. They note that the quantity of alerts that are generated is partially due to the metrics tool vendors are competing over, which prioritizes low false negatives over low false positives. This approach helps ensure fewer missed incidents or vulnerabilities, but often results in a deluge of more trivial alerts. As the metrics by which security tool companies compete do not incentivize lower overall alert volume, we focus on reducing the time required to resolve alerts as a way to improve analyst performance.

Improving analyst alert resolution time is not only an intuitive path for improvement, but also a commonly used metric for analyst workload. The SANS SOC Survey asked the question "how do you calculate per-analyst workload"; most respondents use ticket start and stop times, calculating the time to clear a ticket as an approximation for workload. Thus, an improvement in the time to clear an alert should be reflected in organizational performance metrics [3].

Workshop on Security Operation Center Operations and Construction (WOSOC) 2025
24 February 2025, San Diego, CA, USA
ISBN 79-8-9919276-0-4
https://dx.doi.org/10.14722/wosoc.2025.23016
www.ndss-symposium.org

Surveys of security practitioners have identified a lack of context provided in alerts as a barrier to efficient alert investigations [2]. The "Voice of The SOC" report from Tines seeks to capture the challenges facing those in the industry and gather recommendations for future development. Their 2023 publication examined the metrics used to measure job performance; they found that the mean time to investigate was the top response [15]. Similarly, survey respondents ranked everyday challenges they encounter, and the number one challenge was "too many logs", with "too many alerts" coming in at number three. The report also observed that "the greatest challenges security practitioners face on a regular basis include too much data and not enough information." We seek to answer this call through an implementation of the novel event-based methodology, which turns raw logs into palatable and actionable information, crystallized in the SOC dashboard for alert investigations.

Existing research investigating authentication logs focus on challenges other than transformation into more useful derivatives. One study around authentication logs developed a statistical framework to identify suspicious login attempts and validated it using a sample of LinkedIn login data. A prototype implementation of their system trained on six months of real data and used only two features: IP address and UserAgent, and achieved a high ratio of true positive to false positive with a 0.96 AUC [5]. Another study leveraged authentication logs to construct a simple model to predict compromise using log-in time and consecutive failures. While these works do not transform the logs into derivatives, they do demonstrate how data that SOCs already collect can be used in novel ways.

III. Methodology

This section provides a brief review of the methodology and an overview of the configuration of the SOC event dashboard. We first detail the tools and packages used to develop the software then move to the raw data sources and derived data types. The dashboard design is then reviewed, and examples are presented of the metrics, charts, and reports made available.

A. Event Methodology

Figure 1 displays the basic data processing flow underpinning the authentication events that are the core unit of analysis for the derivatives available on the dashboard. For a more detailed description, please reference the original methodology paper [6]. The exact data processing used here differs in several ways as a result of the authentication data coming from the Graph API rather than the Entra ID portal and a change in intended use.

The API logs are more detailed than those from the portal and come as unified logs with all related attributes in a single row. In the portal, data is split into different file types by the interactivity label that Microsoft assigns and have separate authentication details files that provide a few necessary authentication related attributes. Thus, the largest changes to the source data are greater granularity and a lack

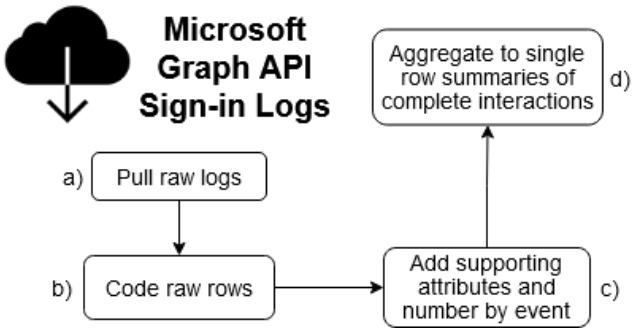

Fig. 1. Event Data Processing

of separation of source files; no manual unification of data was necessary. All attributes are available without additional processing.

Lastly, the largest change is not due to a difference in data, but a difference in use case. The purpose of the event methodology as originally constructed was to research user-interactive authentication events, requiring a dataset that accurately represented a user experience as the unit of analysis. Thus, when using this methodology for researching user experience related questions, non-interactive authentication events were discarded.

The SOC dashboard serves a very different purpose. Many alerts can be triggered by events that would not be labeled as interactive authentications, and non-interactive authentication data can be relevant to alert investigations. Therefore, data processing was modified to preserve these authentication events instead of discarding them. While a dashboard that focuses solely on user-interactive authentication events could still have utility, it would be unnecessarily limiting to maintain this constraint.

B. Dashboard Overview

The SOC dashboard, still in the 0.x phase under active development, was developed and tested using Qt Creator, a cross-platform integrated development environment (IDE), written in C++.[13] The libraries utilized were primarily native QT libraries, with an open source json parser from nlohmann[11]. Threading was implemented using the QThread class, and API calls are performed using the curl c++ library[4].

Broadly described, the dashboard takes in raw sign-in logs and security alerts from the Microsoft Graph API, transforms the raw data series into single row authentication events that encapsulate a complete interaction as shown in Figure 1. These authentication events serve as the building blocks for the enhanced reporting the dashboard offers, and are the atomic units from which other derived datasets are produced. An "Event" includes the core details about an authentication interaction including the elapsed time, authentication details, error classification, and frequency of errors. Additional derived attributes are included and will be explained as relevant in the text. For more details please reference [6].

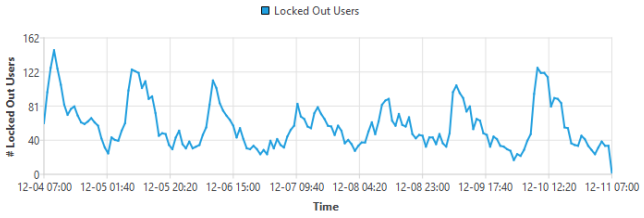

Fig. 2. Example locked-out graph

C. Dashboard Design

The dashboard is designed as a single window application with tabs or "panels" for each form of data used, and several views aggregating the data, presenting it in visualizations such as charts and graphs. Table I lists the relevant panels, their purpose, data types and selected examples.

1) User Logs: User logs are event-derived data tables summarizing key event attributes for specific users over a given time period. In the research setting, periodic log aggregation has been used to transform longitudinal event logs into panel data, offering opportunities to perform various kinds of analysis [7]. They include information including the time spent authenticating, number of failures, type and frequency of error encounters, time spent unauthenticated after failure, forms of second-factor authentication used, the number of applications being authenticated to, and more.

Reporting artifacts generated from this enhanced data aggregation include graphs of population success and error rates, and the number of users currently locked out of their accounts.[1] Figure 2 provides an example.

2) Application Logs: Application logs are derived from event logs and present aggregations over a particular application or set of applications. These logs provide several strategic opportunities, including visibility into the unique applications to which users attempted to authenticate. The typical organization introduces over 300 new services every month; in some industries, such as telecommunications and insurance, this number increases to 1,000 new services per month [10]. The application-specific log summary allows organizations to quickly identify lapsed or unused authentication targets that can be decommissioned or otherwise protected to reduce their attack surfaces. In a 30-day sample of authentication data, 22% of the targeted applications had zero successful authentications, and can be quickly identified using the Application Health panel of the dashboard as discussed in section III-C5.

3) Device Logs: Device logs share the same basic design as application and user logs, aggregating authentication data specific to devices over specified time periods. Their primary use case is investigating device-specific alerts, such as indications of unwanted software or unusual network activity. Charts and graphs of a device's recent authentication history including number of users, average session time, authentication failures, and forms of second factor authentication enable an analyst to quickly search for behavior indicating compromise.

4) Alert Logs: The final type of data used in the SOC dashboard is the alert logs, which analysts review when clearing alerts in web portals such as Microsoft Defender.

5) Reports and Health Summaries: In addition to panels where the above data types can be generated as needed, the dashboard includes "health" and "report" panels for the application, user, and device logs. A health panel includes several tables and charts, constructed from the prior log types, to enable a snapshot assessment of the state of authentication for users, devices, or applications, as described in Table I. See Figure 3 for an example application chart.[2]

[1]We say a user is locked out when they have two or more consecutive failed authentication events with over 12 hours unauthenticated

[2]Charts and tables are generated using a default time range that can be adjusted as needed.

TABLE I
DASHBOARD PANELS

Panel	Purpose	Data Types	Example
1. Table Views	Viewing raw data	Raw sign-in logs, coded logs, event logs, user/application/device logs	Tables described in text
2. User Health	At a glance view of global user performance	Graphs: time to authenticate, locked-out users, success rate; Charts: error counts by type, portion of events containing errors	Figure 2
3. Application Health	At a glance view of global application performance	Tables: lowest success rate, highest hacking errors, highest configuration errors; Charts: top 10 application event counts, top 10 applications total time authenticating	Figure 3
4. Device Health	At a glance view of global device statistics	Graphs: unique devices per day, devices with hacking errors	Not fully implemented
5. Report Views	Reports for users, applications, and devices	Equivalent to the "Health" panels, but specific to a subset of users/applications/devices	NA
6. Alert Summary	Summary of selected alert with supporting data	Graphs: user success rate, user errors, application history; Charts: MFA type history, login time history, location history; Tables: event logs, raw logs	Figure 5

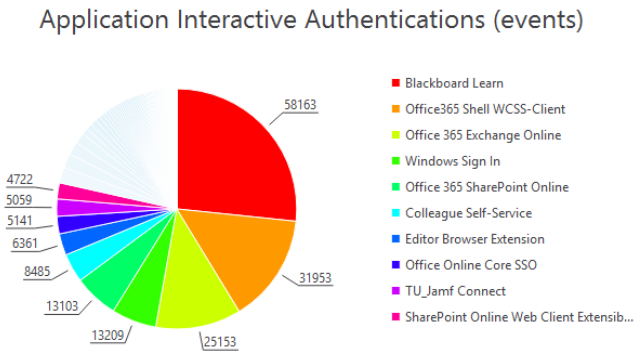

Fig. 3. Application Health Example Chart

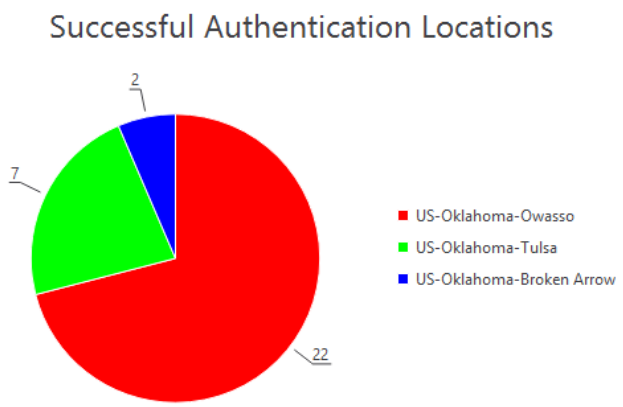

Fig. 4. Authentication Location Pie Chart

The report panels are similar to the health panels, but default to longer timelines and allow the analyst to select a subset 1–n of users, applications, or devices on which to generate the charts and graphs. These panels are designed to quickly provide a historical view of a given asset that can be shared or archived for investigation or audit purposes. The primary difference between the report and health panels is that the health panels are global, while the report panels are specific to one or more users, applications, or devices.

6) Alert Summaries: The alert summary panel leverages available data to consolidate the most relevant information for an alert into a single space, enabling quick and comprehensive investigations. It includes basic details of the alert, tables of event and raw logs, and various charts and graphs displaying success rates, error rates, application usage, authentication times, and more, depending on the alert type. An example of this can be seen in Figure 5 below.

IV. Implementation and Usage

This section presents example investigations complete with the graphs, charts, and data utilized. Investigations are then generalized into two core workflows that can address the majority of alerts.

A. Case 1: Unfamiliar Location

The severity of the "Unfamiliar Location" alerts range from low to high, as captured by Table IV. We chose a high-severity instance as an example of an alert that would receive investigation priority. The "Unfamiliar Location" category is selected from the filter options and a high severity example is selected by double-click. The alert summary is populated similarly to Figure 5, and we began evaluating the inciting event. We define an inciting event as:

> The derived single-row event summary representing 1-n rows of data that includes the single raw authentication log the alert is associated with.

The event attributes were examined, revealing a disparity between the locations of the inciting event and the surrounding events—specifically, an authentication attempt from "AU", Australia, when the user was based in the central United States. Observing this discrepancy, we proceeded with a more comprehensive evaluation of the location relative to the user's history and consulted the location chart. Figure 4 displays the distribution of locations for successful authentications, showing that the user had never successfully authenticated from the location of the sign-in attempt that triggered the alert.[3] Observing a significant deviation from the user's successful authentication history, we concluded that the alert was a true positive. The entire investigation was conducted within a single tab of the dashboard tool, without requiring additional inputs to generate the referenced context.

1) Case 2: Password Spray Attack: OWASP defines password spray attacks as "a type of brute-force attack in which the attacker uses one password (e.g. Secure@123) against many different accounts to avoid account lockouts that would normally occur when brute-forcing a single account with many passwords" [12]. Microsoft assigned high severity ratings to both instances of the password spray alert in the sample. The two alerts, triggered by authentication attempts less than 10 minutes apart, likely indicate the same password spray attack.

The investigation of the alerts begins by clicking on either password spray alert to generate a summary. The inciting event was reviewed, revealing a failed authentication due to a configuration error, which ruled out erroneous user input. The event summary also indicated that a password was used in the authentication attempt, warranting further evaluation. While the event partially matched the characteristics of a password spray attack, it did not align with the expected behavior; such an attack would not typically result in a configuration error, but rather show predominantly incorrect password attempts.

Next, because this type of attack targets multiple users, the global success rate and error rate graphs were consulted, as shown in Figure 6. The graph displays the global success rate, with a red line marking the time of the alert. Observing no drop in success rates or increases in password-related errors at the time of the alert (not shown), we concluded that the alert was a false positive. Additional due diligence can be performed by referencing the application-specific history, which includes a

[3]The same chart is available for failed authentications.

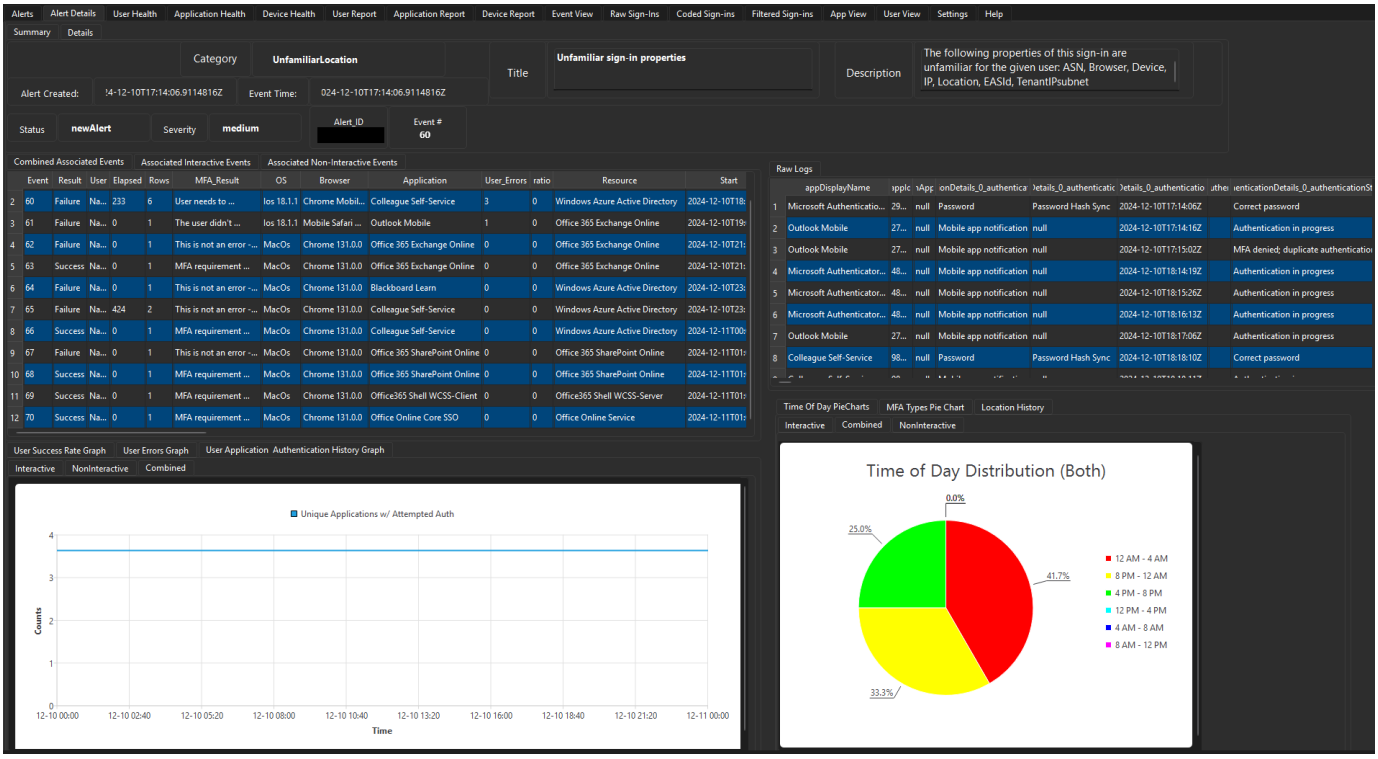

Fig. 5. Dashboard Prototype Example

success rate graph providing greater granularity to ensure the attack was not localized and undetectable at the global level.

B. Workflows

We now present generalized workflows inspired by the two examples. Figure 7 shows the generalized workflow of the ”unfamiliar location” example alert. This workflow is suitable for alerts triggered by a single action and not indicative of a pattern or multiple actions. The second workflow, generalized from Example 2, is designed for alerts involving or implying the relevance of multiple actions leading to a trigger.

1) Workflow 1: Upon loading the alert summary, the analyst first compares the inciting event to those immediately proceeding and following it.

The analyst evaluates the primary attributes of the inciting event, such as the applications being authenticated to, the types of second factor being used, and more. The exact attributes of interest vary depending on the alert under investigation. The analyst then determines whether the attributes are consistent with the user's observed authentication history. If they match, the alert is typically identified as a false positive. If the inciting event encompasses multiple rows of raw data, the analyst consults the ”Raw” tab which is pre-populated with the original data rows associated with the event. The raw data available here has a greater number of attributes than the equivalent raw data normally consulted in the authentication log portal, giving the analyst enhanced granularity when greater scrutiny is required. This can help the analyst confirm that each individual action surrounding the alert is legitimate. If the inciting event does not represent multiple rows of raw data, analysts may conclude the alert is a false positive.

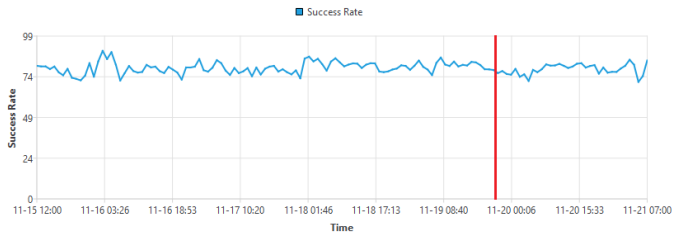

Fig. 6. Global Success Rate around Alert

If the attributes of the inciting event do not match those in the user's surrounding event history, the analyst evaluates the inciting event in relation to the user's broader history. The exact charts and graphs consulted will vary with the specific alert being investigated. When an alert is triggered by a suspicious failure, the graphs of error types and success rates over time can be consulted. This enables the analyst to evaluate whether the type of error(s) encountered were novel in that user's history or if they had previous encounters. Changes in the error composition of failed events can be indicative of malicious or fraudulent activity. Changes in the user's success rate, number of applications being authenticated to, or number of devices used can be similar indicators.

2) Workflow 2: In this workflow, the alert being investigated implies risk based on multiple actions by a single user or

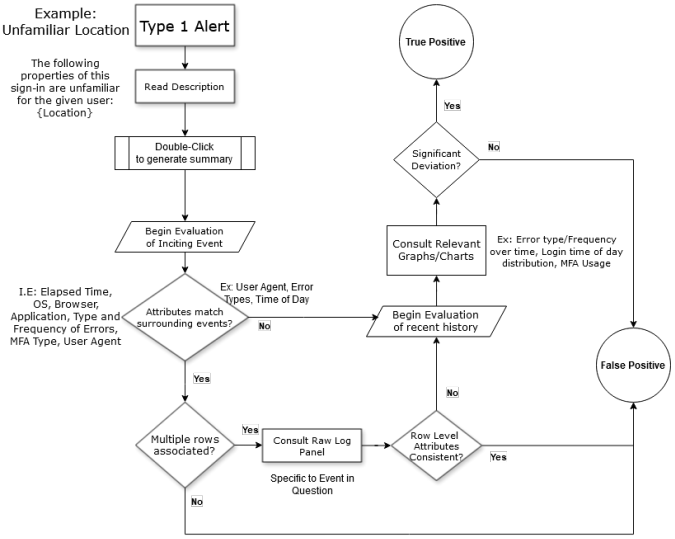

Fig. 7. Workflow 1

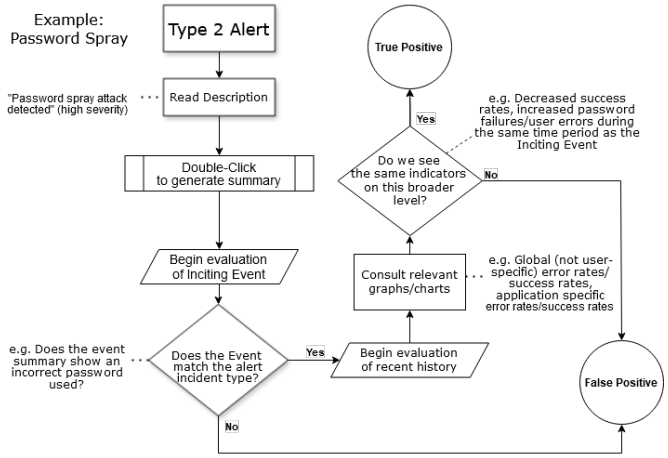

Fig. 8. Type 2 alert workflow

interactions from multiple users. For a single user, this can involve repeated authentication attempts or authentications coming from different locations within a timeframe too short for the user to have traveled the distance between them. For an alert with a multi-user trigger, this could be something like a password spray attack.

The investigation begins by double-clicking the alert on the main display and reviewing the inciting event on the generated alert summary page. In this workflow, the evaluation focuses less on specific raw data attributes and more on a holistic view of the interaction. For the example in Figure 8, which involves a password spray attack, the event is first checked for incorrect password usage. If the inciting event does not match the expected behavior for the alert, a false positive determination may be made.

If the event closely matches the alerted behavior, the analyst proceeds to the next step: evaluating recent history. At this stage, the analyst may expand the scope beyond a single user and consults graphs and charts showing metrics relevant to the alert category; in the previous example, the success rates and error rates across the global set of users. If global metrics unchanged in the time around the alert, a more specific set of charts and graphs can be generated for the application authenticated to in the inciting event. If the charts and graphs show the effects that the inciting incident suggested, the analyst may make a true positive determination. If the expected behavior for the alert under investigation is limited to the inciting event and does not persist within that same user, as in a brute force attack, or across the broader user-base, as in a password spray attack, the analyst may determine that the alert is a false-positive.

V. Event Data Coverage of Alerts

In this section we review the types of alerts present in the Microsoft Defender alert data queried from the Graph API and describe the relevant raw and derived data that the dashboard utilizes for enhanced alert review. Alerts are grouped by the Microsoft assigned, organizationally managed alert category.

TABLE II
Alert Categories

Category	#Sub-Types	Severity Range	#Alerts
1. AnomalousToken	1	Medium	10
2. AnonymousLogin	1	Low to Medium	159
3. CredentialAccess	1	High	9
4. DefenseEvasion	1	Informational	12
5. Discovery	1	Low to Medium	50
6. Execution	1	Informational	7
7. ImpossibleTravel	1	Low to Medium	549
8. InitialAccess	1	Low	9
9. LateralMovement	1	High	2
10. LdapSearchRecon.	1	Medium	4
11. Malware	3	Info. to Medium	74
12. MCASALERT	6	Informational	222
13. PassTheTicket	1	Medium	74
14. PasswordSpray	1	High	2
15. Priveledge Escalation	2	Medium	3
16. Ransomware	2	Medium to High	13
17. SuspiciousActivity	1	Medium	17
18. ThreatManagement	5	Info. to High	466
19. UnfamiliarLocation	1	Low to High	1436
20. UnwantedSoftware	1	Low to High	10
Total Alert Count: 3067			

Alert categories with 1 or less alerts are omitted

A. Review of Alert Types

Table II lists the primary categories of alerts present in a 90 day sample of security alerts from a single alert source, Microsoft Defender, at the author's university. The table provides the abbreviated name of the alert category, the number of subtypes of alert in that category, the range of severity associated, and the number of alerts in that category for the sample. Subtypes represent minor differences within a category, which we omit from this table as they represent no substantive difference in associated data. Note that the category names may differ between organizations, and that a category like "Execution" may represent different alerts in another organizations context. Alerts range in severity from

TABLE III
DASHBOARD DATA PRESENCE FOR ALERT TYPES

Data Type	Alert Category																	
	Anomalous Token	Anonymous Login	Credential Access	Defense Evasion	Discovery	Execution	Impossible Travel	Initial Access	Lateral Movement	LDAP Alerts	Malware	MCAS Alerts	Pass The Ticket	Password Spray	Ransomware	Suspicious Activity	Threat Management	UnfamiliarLocation
Raw Authentication Logs	x	x					x							x				x
Derived - Row	x	x					x							x				x
Derived - Event							x							x				x
User Specific Charts, Graphs	x	x	x	x										x				x
Application/Device Specific Charts, Graphs		x	x	x	x	x	x	x	x	x	x	x	x			x	x	x
Investigation Sufficiency	C	C	P	P	P	P	C	P	N	N	N	P	N	C	P	N	N	C

C = Complete, P = Partial, N = None

Alert categories limited to those with more than one instance in the sample

"Informational", a notification that does not require investigation, to "Low", "Medium", and "High" severity, which can include indicators of account compromise or active presence of malware.

A quick examination of Table II reveals a wide variety of alerts with varying severity and frequency. With a total alert count of 3,067, this averages out to 34.1 per day. Noting that many of these may be "informational", which do not require investigation, we tally the number of alerts for each category of each severity. The total number of alerts requiring investigation is 2,569, including 80 high-severity, 366 medium-severity, and 2,123 low-severity alerts. On a daily basis, this averages to $\leq$ 1 high, 4.1 medium, and 23.6 low severity alerts, for a total of 28.5 per day.

B. Dashboard Investigation Completeness

Table III presents a matrix indicating each type of data utilized by the dashboard (non-exhaustive) that is associated with each alert category. Each row is a different data type and each column is an alert category, with the final row "Investigation Relevant" indicating that the head of the SOC confirmed the available data can be used to completely or partially investigate the alert as specified. A "C" in this row indicates complete investigation coverage, an "N" indicates none, and a "P" indicates partial coverage that would need to be supplemented with another tool for a thorough investigation. The raw data category covers the raw attributes present in the sign-in logs queried from the Graph API. The derived categories include the attributes added through coding and aggregation, and the rest of the data types are aggregations of those derived attributes. After identifying the data of interest for each type of alert, we design an alert summary page that can be used for relevant categories.

C. Alert Coverage

Table IV presents a breakdown of the number of alerts at each severity level, noting which can be investigated by the dashboard. The alerts are summed to calculate the dashboard coverage by category, totaled by level of investigation, and the percentage calculated. In our 90-day sample of 3,067 alerts, we found that 2,156 (70%) could be fully investigated using dashboard data, while 13% could be partially investigated. Excluding 'informational' severity alerts, which do not require investigation, 84% of alerts can be fully investigated using the dashboard, and 13% can be partially investigated.

TABLE IV
DASHBOARD ALERT SEVERITY COVERAGE

Category	High	Medium	Low	Info.	Total
Anomalous Token	0	10	0	0	10
Anonymous Login	0	10	149	0	159
Impossible Travel	0	14	535	0	549
Password Spray	2	0	0	0	2
Unfamiliar Location	40	272	1124	0	1436
Complete Investigation	42	306	1808	0	2156
Partial Investigation	35	43	273	64	415
No Investigation	3	16	48	438	505
Total	80	365	2129	502	3076
Complete Investigation %	53%	84%	85%	0%	70%
Partial Investigation %	44%	12%	13%	13%	13%
No Investigation %	4%	4%	2%	87%	16%

VI. CONCLUDING REMARKS

We have presented a dashboard tool for alert investigations in Security Operations Centers that builds on efforts to aggregate raw authentication logs into interpretable events [6]. We analyzed a 90-day sample of real alerts to evaluate the relevance of authentication logs and their derivatives to SOC analyst investigations. Of 3067 alerts captured in the 90-day sample, 84% of alert investigations could be completed in the dashboard as a standalone tool; a further 13% could be partially investigated.

Now that the dashboard has been developed, it creates multiple opportunities for future work investigating SOC operations. The dashboard will be deployed operationally inside the university SOC from January 2024. It has been instrumented to log all analyst interactions, the specific alerts

being investigated, the supplemental data types consulted, and timestamps for each action.

Once analysts have been adequately trained on the tool, we plan to conduct experiments comparing two groups: a treatment group using the dashboard as their primary investigation tool and a control group relying on standard tools. These experiments will evaluate the dashboard's impact on analysts' performance metrics and help empirically measure the importance of authentication logs in operations.

The dashboard provides more granular timing data than the logs available from the Entra ID portal. Specifically, the API data used by the dashboard includes separate timestamps for the first authentication factor (e.g., password or token) and the second factor (e.g., app notification or SMS code). Future research could leverage this data to study the impact of changes to MFA policies and procedures, focusing on the time cost associated with different second-factor methods or procedural modifications. This timing data can help quantify an additional cost of security enhancements by measuring the time spent authenticating. Furthermore, it can support research into differences in authentication performance both between and within users, examining factors like surveyed constructs or organizational groups such as student majors and faculty departments.

Future work could expand also this methodology and its variations to SOCs in larger organizations or apply it to historical datasets for broader analysis. Additional opportunities include using authentication logs to develop compromise detection models, as demonstrated in [5] and [9]. Furthermore, research could evaluate the effectiveness of specific events and their derived metrics in detecting compromises or improving security outcomes.

Acknowledgments

The authors would like to thank the University of Tulsa SOC, specifically mentioning Tyler Burroughs and Mario Fisher, for their continued support, their openness to research projects and their willingness to foster development of new tools.

References

[1] V. AI, "2023 state of threat detection," Vectra AI, Inc., Tech. Rep., 2023.

[2] B. A. Alahmadi, L. Axon, and I. Martinovic, "99% false positives: A qualitative study of SOC analysts perspectives on security alarms," 2022, p. 2783–2800. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/alahmadi

[3] C. Crowley, "SANS 2024 SOC survey: Facing top challenges in security operations," 2024.

[4] Daniel Stenberg, "libcurl." [Online]. Available: https://curl.se/

[5] D. Freeman, S. Jain, M. Duermuth, B. Biggio, and G. Giacinto, "Who Are You? A Statistical Approach to Measuring User Authenticity," in *Proceedings 2016 Network and Distributed System Security Symposium*. San Diego, CA: Internet Society, 2016. [Online]. Available: https://www.ndss-symposium.org/wp-content/uploads/2017/09/who-are-you-statistical-approach-measuring-user-authenticity.pdf

[6] S. Hastings, C. Bolger, P. Shumway, and T. Moore, "Transforming raw authentication logs into interpretable events," *Workshop on SOC Operations and Construction (WOSOC 2024)*, April 2024. [Online]. Available: https://dx.doi.org/10.14722/wosoc.2024.23003

[7] S. Hastings, T. Moore, B. Brummel, and S. Aurigemma, "The influence of security related stress and self-efficacy on actual security behaviors over time," in *16th IFIP WG 8.11/11.13 Dewald Roode Workshop on Information Systems Security Research*, 2024.

[8] F. Jalalvand, M. Baruwal Chhetri, S. Nepal, and C. Paris, "Alert prioritisation in security operations centres: A systematic survey on criteria and methods(pre-print)," *ACM Computing Surveys*, vol. 57, no. 2, p. 1–36, Feb. 2025.

[9] M. Liu, V. Sachidananda, H. Peng, R. Patil, S. Muneeswaran, and M. Gurusamy, "Log-off: A novel behavior based authentication compromise detection approach," in *2022 19th Annual International Conference on Privacy, Security & Trust (PST)*, 2022, pp. 1–10.

[10] P. A. Networks, "Unit 42 attack surface threat report," 2023.

[11] Niels Lohmann, "json." [Online]. Available: https://json.nlohmann.me/

[12] OWASP Foundation, "Password spraying attack," 2024, https://owasp.org/www-community/attacks/Password_Spraying_Attack, Last accessed on 2024-12-16.

[13] Qt Company Ltd., "Qt creator." [Online]. Available: https://www.qt.io/download-open-source

[14] J. Tilbury and S. Flowerday, "Automation bias and complacency in security operation centers," *Computers*, vol. 13, no. 7, p. 165, Jul. 2024.

[15] Tines, "Voice of the SOC 2023," Tines inc, Tech. Rep., 2023.