

# Time Series Anomaly Detection in Medical Break-the-Glass

Qais Tasali  
qtasali@ksu.edu  
Department of Computer Science  
Kansas State University  
Manhattan, Kansas

Nikesh Gyawali  
gnikesh@ksu.edu  
Department of Computer Science  
Kansas State University  
Manhattan, Kansas

Eugene Y. Vasserman  
eyv@ksu.edu  
Department of Computer Science  
Kansas State University  
Manhattan, Kansas

## ABSTRACT

The time-critical nature of medical emergencies, the requirements for system availability, and for real-time communication all make it exceedingly challenging to consistently enforce least-privilege access during medical emergencies (Break the Glass situations). Strict access control has to be suspended (must fail-open) when an emergency is declared, and only after the emergency has passed can a post-hoc audit be performed to determine the reasons (legitimacy) for overriding access control – standard operating procedure for healthcare facilities. Unfortunately, this does not proactively protect against misuse, but provides for identification and punishment of culprits. It is therefore essentially impossible to limit clinicians access to bare minimum permissions to perform life-saving activities during emergency access, especially in distributed medical systems. In this work we investigate the effectiveness of anomaly detection to ease the human burden of post-hoc audits in the medical Break-the-Glass (BTG) context. We use two different prediction models to perform real-time and post-BTG statistical analysis on time-series session log data for flagging anomalous user sessions and actions. Our approach combines a real-time fast analysis engine working on a partial feature set, as well as a post-hoc, slower analysis tool which works with the complete times series data of everything which occurred during the entire time of the emergency.

## ACM Reference Format:

Qais Tasali, Nikesh Gyawali, and Eugene Y. Vasserman. 2020. Time Series Anomaly Detection in Medical Break-the-Glass. In *Hot Topics in the Science of Security Symposium (HotSoS '20)*, April 7–8, 2020, Lawrence, KS, USA. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3384217.3386397>

## 1 INTRODUCTION

In medical cyber-physical systems (mCPS), fail-closed access control is not always the safest approach. During an emergency situation, medical systems' *availability* must be prioritized over other security properties, leading to a non-traditional access control model. As a result, limiting clinicians access to bare minimum permissions that are essential to perform live-saving activities without compromising patient safety or confidentiality even during unforeseen situations is an unresolved and ongoing challenge. Previous work and industry practices have suggested a virtual version of the "Break the Glass" (BTG) concept, an analogy to breaking a physical barrier

to access a protected resource such as a fire extinguisher during a fire, or crash cards for a medical emergency. BTG is used to override access controls and allow for unrestricted access to resources, e.g. Electronic Health Records (EHRs) [6, 8, 9]. The Health Insurance Portability and Accountability Act (HIPAA) [5, 10] in the United States also requires that availability of medical resources be prioritized over patient's privacy (also known as "fail-open" requirement) to ensure medical systems do not deny life-saving treatment to patients in unforeseen situations such as medical emergencies.

In a collaborative and somewhat chaotic environment such as a medical facility's network, it is challenging to monitor and detect the anomalous behavior of users in real time because of the dynamic nature of the environment and the complexity of users' behavior [3]. For that and many other reasons, HIPAA also requires a post-hoc audit to determine the legitimacy of the access control override and any resource access or modification performed during the event. For auditing purposes, such information must be clearly logged and communicated to the relevant parties. Unfortunately, this does not proactively protect against misuse, but only helps identify and punish a culprit after the fact.

In an attempt to solve the somewhat contradictory set of access control requirements, Tasali et al. [9] attempt to combine normal and emergency behavior into one "meta-model", which retrofits BTG functionality into existing access control policies. Their approach is based on the classic Attribute Based Access Control (ABAC) [4] model, and BTG is specified as a finite state machine in terms of system operating states (*normal*, *controlled BTG*, and *uncontrolled BTG*). To support Break-the-Glass, the system allows overriding "deny" decisions automatically (instead of on a one-by-one basis) during a declared emergency (BTG) session. Transitions between operating states are governed by the fulfillment status of obligations by policy decision point (PDP), which determines whether a change from one state to another is needed. For example, if a clinician overrides an access control decision by initiating a BTG session, and all returned BTG obligations are met, the system changes its state from normal to controlled BTG. Inability to fulfill obligations results in uncontrolled BTG.

In this work, we explore strategies to assist with BTG audits using statistical analysis and anomaly detection to limit the extent of uncertainty of the system state following an emergency access session in [9], and allow for recovery to a known safe state when an emergency ends. (Therefore, in this work we only focus on the uncontrolled BTG state of the system because the system in uncontrolled BTG can only return to normal state via an audit.) We use semi-supervised learning over test cases derived from subsets of complete combinations of all possible interactions within the system in a finite period of time. Since this represents the complete range of behavior of the system, as the number of length of training

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).  
HotSoS '20, April 7–8, 2020, Lawrence, KS, USA  
© 2020 Copyright held by the owner/author(s).  
ACM ISBN 978-1-4503-7561-0/20/04.  
<https://doi.org/10.1145/3384217.3386397>

subsets increases, our model approaches complete coverage of every foreseeable system event sequence.

## 2 APPROACH

Anomaly detection is a long-standing research area in machine learning, used for diverse application domains such as financial fraud detection, computer network intrusion, and fault detection in safety-critical systems [2]. Alizadeh et al. [1] look for behavioral anomalies during BTG by comparing constructed behavior profiles with expected behaviors using logs collected from a medical facility. A final score assigned to a user's profile determines the extent of deviation between that user's actual and expected behavior. Sadly, the study stops short of real-time log analysis.

We adopt the Gaussian Mixture Model (GMM) due to its expressiveness and wide acceptance in statistical modeling [7, 12]. We calculate likelihood for a feature  $x$  given  $\mu$  and  $\sigma^2$ . And if this value is less than a threshold  $\epsilon$ , we flag the session as an anomaly.

### 2.1 Problem Formulation

We define an event  $E_t$ , an uncontrolled BTG session, as a time-stamped observation starting at time  $t$ , and is described by a set of feature-value pairs [11] such as  $\langle \text{"A"}, \text{count} \rangle$ , representing number of times resource "A" was accessed, or  $\langle \text{"Uncontrolled BTG"}, \text{time} \rangle$ , representing total duration of uncontrolled BTG session for the event  $E_t$ . (Note that the feature-value set is extensible.) An event sequence  $S$  is a time-ordered finite sequence of events  $E_{t_1}, E_{t_2}, \dots, E_{t_n}$  which includes all  $n$  events in time interval  $t_1 \leq t \leq t_n$ . This event is associated with user, a *domain object*  $D$ .

If  $X$  is the unknown event, we need a *prediction model*  $\mathcal{P}$  that predicts if  $X$  is anomalous or not, using all features in events  $E_t$ , and  $i$  and  $j$  bounds all events in a given time range. Here  $\mathcal{P}$  is a function that maps an event sequence to  $\{+, -\}$ , where  $+$  and  $-$  denote  $X$  as anomalous or not respectively:  $\mathcal{P} : E_{t_i}, E_{t_{i+1}}, \dots, E_{t_j} \rightarrow \{+, -\}$ .

The model  $Q$  excludes features that cannot be evaluated in real time, such as the duration of the BTG session. One may consider  $Q$  as a way to flag anomalies as close to real-time as possible, at the cost of a higher false negative rate. One may consider  $Q$  as a way to flag anomalies as close to real-time as possible, at the cost of a higher false negative rate. The full list of combined decision for the two models is given in Figure 1.

$\mathcal{P}$  complements the analysis of  $Q$ . In fact,  $\mathcal{P}$  is identical to  $Q$  except that  $Q$  uses a subset of features from  $E_t$  in a given time range. This is why (in Figure 1)  $Q$  flagging an event as anomalous and  $\mathcal{P}$  considering the same event as non-anomalous is not expected.

$Q$	$\mathcal{P}$	Combined Decision
-	-	-
-	+	+
+	+	+
+	-	N/A

**Figure 1: Combined decisions for the models  $Q$  and  $\mathcal{P}$  where  $+$  and  $-$  denote whether an anomaly was detected.**

### 2.2 Model Fitting

To fit the model, we take all the events within a given time range. For all features of an event, we assume it follows a normal distribution  $\mathcal{N}(\mu, \sigma^2)$ , with the probability density function given by

$$p(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

For a user  $D_1$ , if  $x_1$  is the first feature of  $E_{t_1}$ , we calculate

$$\mu_1 = 1/n \sum_{i=1}^n x_1^i \text{ and } \sigma_1^2 = 1/n \sum_{i=1}^n (x_1^i - \mu_1)^2$$

where  $n$  is the number of events  $E_t$  in a given time range. We do this for all features  $x_2, x_3, \dots$  and for all users  $D_1, D_2, \dots, D_m$ . So for all features of  $E_t$ , we calculate  $\mu$  and  $\sigma^2$ , and our model is now ready to predict if a new uncontrolled BTG session will be anomalous or not based on the likelihood of features for the session.

### 2.3 Model evaluation

When an uncontrolled BTG session starts for a user  $D_1$ , we track the features  $\langle \text{feature}, \text{value} \rangle$  that can be evaluated in real time by model  $Q$ . We use a threshold time period  $w$  to denote grace period after a session begins before calculating the likelihood for each feature  $p(x; \mu, \sigma^2)$ . It is recalculated every time the value for the feature changes. Each feature has a minimum threshold  $\epsilon$  assigned to it – some features are more sensitive than others. Whenever the likelihood for any of the features exceeds this threshold, the model predicts the ongoing session as anomalous.

We can determine whether the uncontrolled BTG session is anomalous by calculating the likelihood of all features  $p(x) =$

$$p(x_1; \mu_1, \sigma_1^2) * p(x_2; \mu_2, \sigma_2^2) * \dots * p(x_n; \mu_n, \sigma_n^2) = \prod_{i=1}^n p(x_i; \mu_i, \sigma_i^2)$$

where  $n$  is the total events  $E_t$  in a given time range. We compare this likelihood with an experimentally-derived threshold  $\epsilon$ . If it is greater than the threshold, the session is flagged as anomalous, and tagged for a latter manual audit. Information regarding non-anomalous sessions is added to  $S$  (user profile) so that the model learns the user's behaviour over the time (see Section 2.1).

## REFERENCES

- [1] M. Alizadeh, S. Peters, S. Etalle, and N. Zannone. 2018. Behavior analysis in the medical sector: Theory and practice. In *ACM Symposium on Applied Computing*.
- [2] V. Chandola, A. Banerjee, and V. Kumar. 2009. Anomaly detection: A survey. *Comput. Surveys* 41, 3 (2009).
- [3] Y. Chen and B. Malin. 2011. Detection of Anomalous Insiders in Collaborative Environments via Relational Analysis of Access Logs. In *CoDASPY*.
- [4] V.C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone. 2014. Guide to Attribute Based Access Control (ABAC) Definition and Considerations. NIST Special Publication 800-162.
- [5] Legal Information Institute. 2013. 45 CFR 164.312 - Technical safeguards.
- [6] D. Povey. 1999. Optimistic security: A new access control paradigm. In *NSPW*.
- [7] D.A. Reynolds. 2009. Gaussian Mixture Models. *Encyclopedia of Biometrics* (2009).
- [8] Q. Tasali, C. Chowdhury, and E.Y. Vasserman. 2017. A Flexible Authorization Architecture for Systems of Interoperable Medical Devices. In *SACMAT*.
- [9] Q. Tasali, C. Sublett, and E.Y. Vasserman. 2020. Controlled BTG: Toward Flexible Emergency Override in Interoperable Medical Systems. *EAI SESA* (2020).
- [10] U.S. HHS Office for Civil Rights. 2013. HIPAA Administrative Simplification.
- [11] G.M. Weiss and H. Hirsh. 1998. Learning to predict rare events in categorical time-series data. In *International Conference on Machine Learning*.
- [12] K. Yamanishi, J.-I. Takeuchi, G. Williams, and P. Milne. 2004. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. *Data Mining and Knowledge Discovery* 8, 3 (2004).