Change Point Detection for Monitoring SIP Networks

Çağatay Yıldız*, Murat Semerci*†, Taha Yusuf Ceritli*, Barış Kurt*, Ali Taylan Cemgil* and Bülent Sankur‡
*Department of Computer Engineering, Bogazici University, Istanbul, Turkey
Email:{cagatay.yildiz1, murat.semerci, yusuf.ceritli, baris.kurt, taylan.cemgil}@boun.edu.tr
†Department of Cyber Security, NETAS, Istanbul, Turkey
Email: msemerci@netas.com.tr
‡Department of Electrical & Electronics Engineering, Bogazici University, Istanbul, Turkey
Email: bulent.sankur@boun.edu.tr

***Abstract*—SIP (Session Initiation Protocol) is currently the most popular protocol that enables session control in computer communication networks. Concomitantly with its wide deployment, SIP networks have become targets of various attacks, such as DDoS attacks. This study focuses on intelligent systems for DDoS attack monitoring based on the observation of message flows and on anomaly detection methods. These methods, using statistical message flow models, analyze the network traffic and raise an alarm when a disruptive change occurs. We test the proposed models extensively with data sets generated from two sources, a network traffic generator and an attack simulator.**

I. INTRODUCTION

VoIP (Voice over IP) telephony and packet-switched networks are rapidly replacing circuit-switched networks due to their cost-effective management and superior features as a carrier in multimedia applications. Session Initiation Protocol (SIP) has become a very popular request-response type protocol to initialize, audit, modify and end VoIP sessions between the clients [1]. It provides all the signaling and service tools to register clients, to check their locations and their availability, to exchange information on their data transmission capabilities, and to start and to end communication sessions.

The popularity of SIP is due to its lightweight nature, simplicity and the ease of implementation. Many of the communication service providers presently operate with SIP servers. With the prevalence of 5G technologies, the usage and the coverage of VoIP technologies is expected to broaden and SIP is a strong candidate to be the most preferred signaling protocol. But the simplicity of text-based SIP and the accessibility of SIP servers via the Internet make both SIP servers and users open targets for threats such as cyber-attacks, fraud or service abuse. In the literature, there is a plethora of studies describing SIP security threats and their potential countermeasures [2]. These countermeasures typically rely on message traffic features such as the number of ongoing sessions or active clients, the intensity of SIP messages at the SIP server, the resource usage rates of the server and so forth [3].

In this study we focus only on detection of the Distributed Denial of Service (DDoS) attacks. DoS attacks can be regarded as the most commonly occurring cyber-attack [4]. DDoS type of attack is especially hard to combat with since multiple computers are exploited as attackers.

In this work, we propose two different change point models to detect DDoS attacks. Both of them are based on the assumption that the pattern and intensity of the message traffic at a SIP server would change significantly under an attack. When the features derived from the message traffic are considered as a time series, the attack detection problem is reduced to change point detection in time series. The first model, called Adaptive Distance-based Change Point Detection Estimator (DCPM), considers the variations in the distances between the successive feature vectors. The second model, called Bayesian Change Point (BCPM), assumes that an underlying mechanism generates the data and any observed traffic pattern(feature vector) with low probability of being generated from this mechanism indicates a possible attack.

The paper is organized as follows: Section II discusses the first proposed model, DCPM. Section III focuses on BCPM. The experimentation and its results are provided in Section IV. Finally, the conclusion and future research directions are discussed in Section V.

II. ADAPTIVE DISTANCE-BASED CHANGE POINT DETECTION ESTIMATOR

Feature vector instances, if generated from the same process, are expected intuitively to have high similarity, or small distances, while feature vectors from different generative processes tend to be less similar, or their pairwise distances will be larger. Based on this premise, a significant change in the distances between consecutive feature vectors in a time series can be interpreted as an indicator of a change in the data generating process. In the context of SIP networks, such an abrupt change can signal a DDoS attack. Thus, DCPM first computes the distances between sequential feature vectors. It is common to use distance functions in machine learning problems, such as classification (k-nearest neighbor, LMNN [5] etc.) or clustering (k-means, c-means, etc.). In this work, we use Mahalanobis and Euclidean Distances, whose details are presented below.

A. Mahalanobis and Euclidean Distances

Consider two d-dimensional vectors in our data set, V, $\mathbf{x}_i, \mathbf{x}_j \in \Re^d$. The squared Euclidean distance between these vectors is as follows:

$$D_E(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{x}_i - \mathbf{x}_j) \tag{1}$$

The squared Euclidean distance assumes that the feature components are uncorrelated and that they have the same scale. If feature components had different scale, the ones with larger scale would dominate the distance measure and lead to wrong conclusion in a pattern recognition or learning problem.

The Mahalanobis distance, given in Equation 2, is defined over symmetric positive semi-definite (PSD) matrices, ($\mathbf{M} \in \mathbf{S}_+$) $d \times d$, and the choice of $\mathbf{M}$ can be made to account for the correlations between features and the differences between scales. The sample co-variance matrix, $\mathbf{\Sigma}$, gives rise to a special case of Mahalanobis matrix, which assumes the data is generated from a multivariate Gaussian distribution. Under this choice it can be shown that $\mathbf{M}$ maps the data to an uncorrelated and unit variance Gaussian distribution. In case where the features follow a standard Gaussian with uncorrelated variables, then we have: $\mathbf{M} = \mathbf{\Sigma} = \mathbf{I}$.

$$D_\mathbf{M}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \tag{2}$$

Any symmetric positive semi-definite matrix can be factorized as $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$ such that $\mathbf{L}$ is a $d' \times d$ projection matrix and $d' \leq d$. Thus, the relation below can be obtained.

$$\begin{aligned} D_\mathbf{M}(\mathbf{x}_i, \mathbf{x}_j) &= (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{L}^\top \mathbf{L} (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{L}(\mathbf{x}_i - \mathbf{x}_j))^\top \mathbf{L} (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{L}\mathbf{x}_i - \mathbf{L}\mathbf{x}_j)^\top (\mathbf{L}\mathbf{x}_i - \mathbf{L}\mathbf{x}_j) \\ &= \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 = D_E(\mathbf{z}_i, \mathbf{z}_j) \\ &= D_\mathbf{L}(\mathbf{x}_i, \mathbf{x}_j) \end{aligned} \tag{3}$$

where $\mathbf{z}_i = \mathbf{L}\mathbf{x}_i$ is the projected vector. Equation 3 shows that the Mahalanobis distance in the feature space is equivalent to the Euclidean distance in a projected space.

B. Distance-Based Change Point Model

The distance-based change detection is achieved by the inspection of the running sum of distances between the current feature vector and the its immediate predecessors in a time-frame of pre-determined size (k). The running distance sum is compared with a preset threshold value, ϵ_{th}, and an alarm is raised if the sum exceeds ϵ_{th}. The main novelty of this study is to learn a Mahalanobis matrix under a loss function that adapts itself to the inlier variations and trends in the traffic intensity in order to more reliably detect attacks.

The running distance sum can be defined as a function of the symmetric positive definite matrix, $\mathbf{M} \in \mathbf{S}_{++}$, as follows:

$$f(\mathbf{M} | \mathbf{x}_t : \mathbf{x}_{t-k-1}) = \sum_{j=t-k-1}^{t-1} (\mathbf{x}_t - \mathbf{x}_j)^\top \mathbf{M} (\mathbf{x}_t - \mathbf{x}_j) \tag{4}$$

```
Initialize $\mathbf{M}_0$ (by default $\mathbf{I}$)
Initialize $k$ (by default 5), $\lambda$ (by default 1), $\beta$ (by default 1) and $\epsilon_{th}$
repeat
    if $f(\mathbf{M}_{t-1} | \mathbf{x}_t : \mathbf{x}_{t-k-1}) > \epsilon_{th}$ then
        Raise the alarm
    end if
    Evaluate $\mathbf{M}^*$
    Set $\mathbf{M}_{t-1} = \mathbf{M}^*$
until The traffic ends
```

Fig. 1. Adaptive online distance-based change point detection algorithm

If the running distance sum computed using the current Mahalanobis matrix is above the threshold, $f(\mathbf{M}_{t-1} | \mathbf{x}_t : \mathbf{x}_{t-k-1}) > \epsilon_{th}$, then an alarm is raised. The Mahalanobis matrix is updated periodically under the loss function given below:

$$\min_{\mathbf{M} \in \mathbf{S}_{++}} \quad f(\mathbf{M} | \mathbf{x}_t : \mathbf{x}_{t-k-1}) + \lambda D_{ld}(\mathbf{M}, \mathbf{M}_{t-1}) + \beta D_{ld}(\mathbf{M}, \mathbf{I}) \tag{5}$$

In Equation 5, the second and the third terms, $\lambda D_{ld}(\mathbf{M}, \mathbf{M}_{t-1})$ and $\beta D_{ld}(\mathbf{M}, \mathbf{I})$, respectively, are logarithmic determinant divergence [6] regularization functions. The former imposes the updated matrix to be as similar as possible to its predecessor while the latter imposes it to be as close as possible to the identity matrix to prevent it from converging to an irrelevant matrix. Thus, their weights provide a trade-off between taking into consideration and discarding (aging) of recent measurement data through the preceding Mahalanobis matrix. The parameters to be set are, t time index, k nearest neighbor (time-frame size) and the regularization cost weights, λ and β. When the algorithm starts, $\mathbf{M}_0$ is initialized to the identity matrix, $\mathbf{M}_0 = \mathbf{I}$. LogDet function is a pseudo-metric that measures the distance between two matrices. Since it is a convex function with respect to its first parameter, we are guaranteed to find the optimal positive definite matrix.

$$D_{ld}(\mathbf{M}, \mathbf{M}_{t-1}) = tr(\mathbf{M}\mathbf{M}_{t-1}^{-1}) - \log\det(\mathbf{M}\mathbf{M}_{t-1}^{-1}) - d \tag{6}$$

The optimal Mahalanobis matrix ($\mathbf{M}^*$) can be found by taking the derivative of Equation 5 and setting it to zero.

$$\begin{aligned} \mathbf{M}^* \;=\; & \Bigg(\frac{\lambda}{\lambda+\beta} \mathbf{M}_{t-1}^{-1} + \frac{\beta}{\lambda+\beta} \mathbf{I} + \\ & \frac{1}{\lambda+\beta} \sum_{j=t-k-1}^{t-1} (\mathbf{x}_t - \mathbf{x}_j)(\mathbf{x}_t - \mathbf{x}_j)^\top \Bigg)^{-1} \end{aligned} \tag{7}$$

This Mahalanobis matrix update is repeated at each time index and the change detection algorithm is given in Figure 1.

The performance of the algorithm heavily depends on the traffic volume and the features of the system (the dimension of the feature vector, the size of the time-frame etc). Experiment evidence has shown that the threshold can be set empirically as follows:

$$\epsilon_{th} = k \left(\frac{d}{2} \right)^2 \tag{8}$$

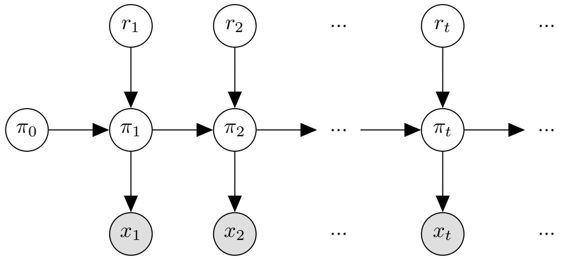

Fig. 2. Graphical model of Bayesian change point model

III. Bayesian Multiple Change Point Model

Bayesian change point model (BCPM) is a variant of switching state space models [7]. In BCPM observations are denoted by x_t and latent variables are represented by π_t and r_t. Here, π_t is a Markov chain conditioned on r_t and $r_t \in \{0, 1\}$ indicates the regime. Furthermore, observations x_t are conditioned on latent states π_t. The graphical model of BCPM is illustrated in Figure 2.

The observation at time t, x_t, is assumed to be a random variable generated from a multinomial distribution with an unknown parameter π_t. At change points, or when $r_t = 1$, π_t is reset to a new value; otherwise, it keeps its value unchanged. Considering all together, the generative model of BCPM can be expressed as follows:

$$\begin{aligned}
\pi_0 &\sim \mathcal{D}ir(\pi_0; a) \\
r_t &\sim \mathcal{BE}(r_t; p) \\
\pi_t | r_t, \pi_{t-1} &\sim [r_t = 0]\delta(\pi_t - \pi_{t-1}) + [r_t = 1]\mathcal{D}ir(\pi_t; a) \\
x_t | \pi_t &\sim \mathcal{M}(x_t; \pi_t)
\end{aligned}$$

Here, $\mathcal{D}ir$, $\mathcal{BE}$ ve $\mathcal{M}$ represent Dirichlet, Bernoulli and Multinomial distributions, respectively, and δ denotes the Dirac delta function. With K dimensional observations, these distributions are expressed as follows:

$$\begin{aligned}
\mathcal{D}ir(\pi; a) &= \frac{\Gamma\left(\sum_{i=1}^{K} a_i\right)}{\prod_{i=1}^{K} \Gamma(a_i)} \prod_{i=1}^{K} \pi_i^{a_i - 1} \\
\mathcal{BE}(r; p) &= p^r (1-p)^{(1-r)} \\
\mathcal{M}(x; \pi) &= \frac{\Gamma\left(\left(\sum_{i=1}^{K} x_i\right) + 1\right)}{\prod_{i=1}^{K} \Gamma(x_i + 1)} \prod_{i=1}^{K} \pi_i^{x_i}
\end{aligned}$$

There are two parameters in the above model, namely, a and p, to be set. The former one is sometimes referred as *reset* parameter. In the case of a change π_t is set to a new value, which is a random variable drawn form a Dirichlet distribution with parameter a. The other parameter p denotes the prior probability of change.

In our BCPM, deciding whether a change has occurred at time t is tantamount to calculating the posterior probability of r_t. As we aim to do so online, the task is reduced to filtering, or evaluating $p(r_t | x_{1:t})$. This can be achieved by integrating out π_t:

$$\begin{aligned}
p(r_t | x_{1:t}) &\propto p(r_t, x_{1:t}) \\
&= \int_{\pi_t} p(r_t, \pi_t, x_{1:t}) \qquad (9)
\end{aligned}$$

State space models in which π_t is continuous and the cardinality of r_t is greater than 2 are intractable as their complexity grows exponentially with t [8]. Fortunately change point models do not suffer from this as r_t can only take two different values and filtering can be done via forward-backward algorithm. Let forward messages be defined as follows:

$$\begin{aligned}
\alpha_{t|t-1}(r_t, \pi_t) &= p(r_t, \pi_t, x_{1:t-1}) \\
\alpha_{t|t}(r_t, \pi_t) &= p(r_t, \pi_t, x_{1:t})
\end{aligned}$$

As shown in equation 9, $\alpha_{t|t}(r_t, \pi_t)$ is proportional to the target distribution. Below is the way $\alpha_{t|t}(r_t, \pi_t)$ is expressed recursively.

$$\begin{aligned}
\alpha_{t|t}(r_t, \pi_t) &= p(r_t, \pi_t, x_{1:t}) \\
&= \sum_{r_{t-1}} \int_{\pi_{t-1}} d\pi_{t-1} p(r_{t-1:t}, \pi_{t-1:t}, x_{1:t}) \\
&= \sum_{r_{t-1}} \int_{\pi_{t-1}} d\pi_{t-1} p(r_{t-1:t}, \pi_{t-1:t}, x_{1:t-1}) \\
&\quad \times p(x_t | r_t, \pi_t) \\
&= \sum_{r_{t-1}} \int_{\pi_{t-1}} \Big(d\pi_{t-1} \alpha_{t-1|t-1}(r_{t-1}, \pi_{t-1}) \\
&\quad \times p(r_t, \pi_t | r_{t-1}, \pi_{t-1}) \Big) p(x_t | r_t, \pi_t) \qquad (10)
\end{aligned}$$

The degree of belief on the filtered density $p(r_t | x_{1:t})$ can be enhanced if more evidence is provided. However, because the inference task is done online, we are restricted to use fixed-lag smoothing. That is, if the lag is selected a constant value L, we calculate $p(r_t | x_{1:t+L})$. For this, backward message passing is needed. If messages are defined as

$$\begin{aligned}
\beta_{t|t+1}(r_t, \pi_t) &= p(x_{t+1:t+L} | r_t, \pi_t) \\
\beta_{t|t}(r_t, \pi_t) &= p(x_{t:t+L} | r_t, \pi_t)
\end{aligned}$$

backward recursion becomes

$$\begin{aligned}
\beta_{t|t}(r_t, \pi_t) &= p(x_{t:t+L} | r_t, \pi_t) \\
&= \sum_{r_{t+1}} \int_{\pi_{t+1}} d\pi_{t+1} p(x_{t:t+L}, \pi_{t+1}, r_{t+1} | r_t, \pi_t) \\
&= \sum_{r_{t+1}} \int_{\pi_{t+1}} d\pi_{t+1} p(x_{t+1:t+L}, \pi_{t+1}, r_{t+1} | r_t, \pi_t) \\
&\quad \times p(x_t | r_t, \pi_t) \\
&= \sum_{r_{t+1}} \int_{\pi_{t+1}} \Big(d\pi_{t+1} \beta_{t+1|t+1}(r_{t+1}, \pi_{t+1}) \\
&\quad \times p(r_{t+1}, \pi_{t+1} | \pi_t, r_t) \Big) p(x_t | r_t, \pi_t) \qquad (11)
\end{aligned}$$

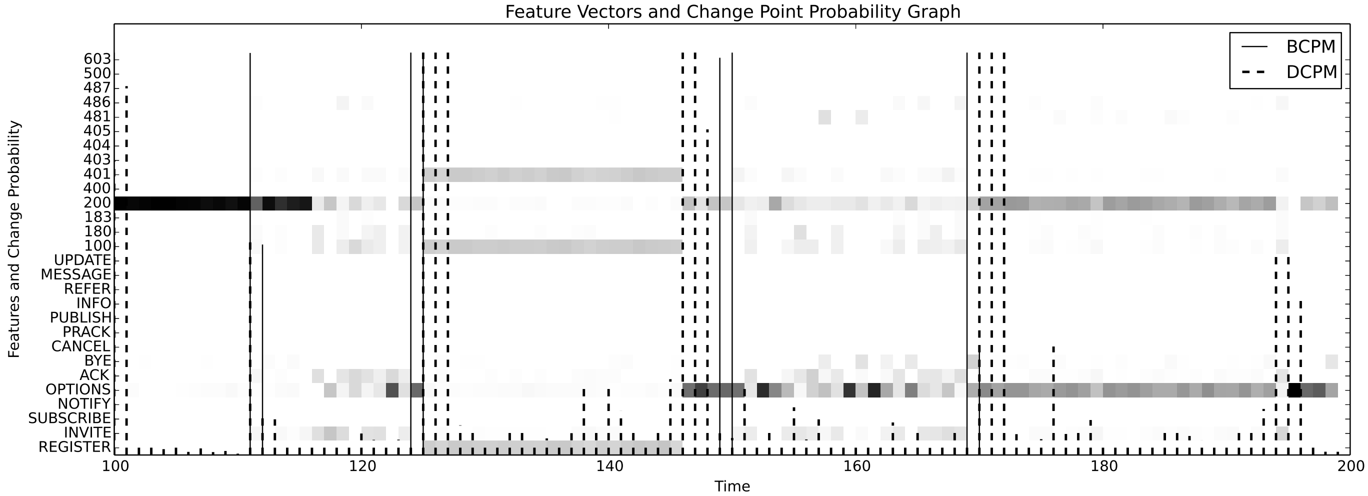

Fig. 3. Visualization of change points probabilities on the histogram of the data for $t \in [100, 200]$. The higher the bars, the higher the probability of change.

Then the smoothed density $p(r_t, \pi_t, x_{1:t+L})$ is expressed as the multiplication of forward and backward messages:

$$\begin{aligned} p(r_t, \pi_t, x_{1:t+L}) &= p(r_t, \pi_t, x_{1:t}) p(x_{t+1:t+L} | r_t, \pi_t) \\ &= \alpha_{t|t}(r_t, \pi_t) \beta_{t|t+1}(r_t, \pi_t) \end{aligned}$$

IV. Experiments

A. Experiment Design

The performance of change point detection models, detailed in Sections II and III, is analyzed with synthetic data. An Asterisk-based PBX software, named *Trixbox*, serves as a SIP server [9]. To mimic the traffic on a SIP server, we have built a probabilistic SIP network simulation system, which initiates calls between a number of users in real-time [10]. Lastly NOVA V-Spy, a vulnerability scanning tool, is used to simulate DDoS attacks targeting the server [11].

Data are collected by inspecting each SIP packet that arrives to or sent by the server. Counts of 14 SIP request and 14 SIP response packets are recorded periodically for each time unit (which is assumed to be a second in the experiments) and a 28 dimensional vector, made up of packet counts per unit interval, are fed to the models. The codes of packet types for which counts are recorded are as follows:

- Requests: Register, Invite, Subscribe, Notify, Options, Ack, Bye, Cancel, Prack, Publish, Info, Refer, Message, Update
- Responses: 100, 180, 183, 200, 400, 401, 403, 404, 405, 481, 486, 487, 500, 603

The experimental environment is controlled by two variables: the intensity of the traffic generated by the simulator and the intensity of DDoS attacks. We have set both of them as two-valued variables -corresponding to low or high intensity situations- so that we have 4 different setups in total (low traffic-low attack, low traffic-high attack, high traffic-low attack, high traffic-high attack). In each setup, five DDoS attacks (Invite, Register, Options, Cancel and Bye) are simulated. Each simulation is replicated twice - once when the rate of DDoS packets is constant and once when it is allowed to fluctuate. Notice that since the network traffic pattern changes substantially at the beginning and at the end of any attack, both instances are labeled as change points, and hence must be detected (see Figure 3).

B. Results

We measure the performance of models using precision (P) and recall (R) measures. Good performance is achieved when both P and R are close to 1. As the rates of false alarm and/or undetected change points increase, P and R, respectively, diverge more from 1. We also calculate the F-measure (F), or the harmonic mean of precision and recall. The time responsivity (T) shows the average delay the system incurs to detect an attack.

$$\begin{aligned} \text{Precision (P)} &= \frac{\text{\# true alarms}}{\text{\# alarms}} \\ \text{Recall (R)} &= \frac{\text{\# true alarms}}{\text{\# all changes}} \\ \text{F-Measure (F)} &= 2\frac{P \times R}{P + R} \\ \text{Responsivity (T)} &= \frac{\text{Total delay of true alarms}}{\text{\# true alarms}} \end{aligned}$$

Results are provided in Tables I and II. In addition to P, R and F, we measure the change point detection performance as IR and TR, corresponding, respectively, to the detection recall of attack onsets and attack offsets. Finally we give the responsivity encountered in the detection of change points. Table I exhibits the performance with the highest F measure among different parameter settings experimented, which are $k = \{3, 5, 7, 9\}$, $\lambda = \{1.0, 2.0, 4.0\}$ and $\beta = \{1.0, 2.0, 4.0\}$.

For DCPM, the precision is inversely proportional, as expected to the time-frame size results not shown in Table I. For fixed regularization weights, if the frame size is increased, the system becomes more stable and the number of all alarms

TABLE I
RESULTS OF THE DISTANCE-BASED CHANGE POINT MODEL

Traffic Intensity	Attack Intensity	P	R	F	IR	TR	T (sec)
Low	Low	0.88	0.75	0.81	1	0.5	0.2
Low	High	0.91	1	0.95	1	1	0.23
High	Low	0.85	0.85	0.85	1	0.7	0.53
High	High	1	1	1	1	1	0.3

TABLE II
RESULTS OF THE BAYESIAN CHANGE POINT MODEL

Traffic Intensity	Attack Intensity	P	R	F	IR	TR	T (sec)
Low	Low	0.81	0.65	0.72	1	0.3	0.2
Low	High	0.87	0.825	0.84	1	0.65	0.45
High	Low	0.94	0.8	0.86	1	0.6	0.35
High	High	0.93	0.9	0.91	1	0.8	0.43

decreases. This behavior may be due to the increase in the threshold value, so that changes with small magnitudes are tolerated. An alternative explanation is that the current feature vector, while deviating from its immediate predecessor, may not deviate from its older predecessors.

The regularization weights, (λ and β), provide a trade-off for aging. When λ is boosted, the system pays more attention to the recent trends in the traffic so that it becomes more tolerant to small changes in the recent past. If β is increased, then the system is more sensitive to any change in the traffic, and the false alarm rate increases proportionally. The upside of it is that its capacity to detect the offset of attacks is increased, which results in higher recall rates.

Figure 3 illustrates the data and change point probabilities of both methods for a 100-second section of the simulation, starting at 100. As indicated by the plot, a register attack and an options attack are simulated during the intervals [125-145] and [170-...]. Note that in DCPM, the running distance sum is proportional to the threshold value, which is clipped to 1 if it is greater than 1. As desired, our models do not react to fluctuations in the network traffic, which is best observed around $t = 120$ and $t = 160$. Notice the false alarms at $t = 110$ and at the end of the run. Furthermore, we observe that DCPM detects all change points, although it usually gives a few consecutive warnings before it settles down M. BCPM on the other hand, raises alarms only when an abrupt change in the smoothed intensity occurs, which is why there is not one-to-one correspondence between the data and the alarms raised by BCPM. Notice that since the two approaches are different, their outputs are not identical.

IR scores in both tables indicate that onsets of all attacks are detected whereas not all offsets are detected. This is because the change in the packet rates at the offset is not as sharp as that at the onset, which is a property of V-Spy. We also see that the change in the model parameters, traffic or attack intensity do not yield a significant change in the delay. Considering the one second monitoring intervals and the fact that changes are detected within half a second on the average, we can conclude that our models have satisfactory reaction times.

V. CONCLUSION

In this study, we have proposed two different change point models to detect the attacks targeting SIP protocol. DCPM, as its name suggests, focuses on the distance changes between successive vectors with the rationale that a change in the data generating process will cause distance changes in the generated data. While no assumptions are made for the underlying data models, the parameters and the threshold should be judiciously set for good performance. The computational effort is small for reasonable feature dimensions, e.g., in several tens.

In BCPM, on the contrary, there is no threshold but only two model parameters to tune (p and a); however, the computations are much heavier. This is because this model attempts to detect changes in the data generating mechanism and raises an alarm if the odds of change in one of the generative model parameters, π_t, is very close to 1. On the other hand, as all latent variables are marginalized out, BCMP is more robust in noisy networks.

One way to improve the proposed models is by using data collected from alternative sources, such as resource usage, call states, log files, and to detect other types of threats (service abuses). The combination of two models to run simultaneously is also being investigated.

ACKNOWLEDGMENT

This study is a Bogazici University - NETAS collaboration and it is funded with TEYDEB project number 3140701, "Realization of Anomaly Detection and Prevention with Learning System Architectures, Quality Improvement, High Rate Service Availability and Rich Services in a VoIP Firewall Product”, by the Scientific and Technological Research Council Of Turkey (TUBITAK). NOVA V-Gate and V-Spy are trademark cyber-security products of NETAS.

REFERENCES

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, 2002.

[2] A. D. Keromytis, "A Comprehensive Survey of Voice Over IP Security Research", Communications Surveys Tutorials, IEEE, 14(2):514-537, 2012.

[3] S. Ehlert, D. Geneiatakis, T. Magedanz, "Survey of Network Security Systems to Counter SIP-based Denial-of-service Attacks", Journal of Computers & Security, 225-243, 2010.

[4] https://www.us-cert.gov/ncas/tips/ST04-015

[5] K. Q. Weinberger, L. K. Saul, “Distance Metric Learning for Large Margin Nearest Neighbor Classification", Journal of Machine Learning Research (JMLR), 10:207-244, 2009.

[6] J. V. Davis, B. Kulis, P. Jain, S. Sra, I. Dhillon, "Information-Theoretic Metric Learning”, 24th International Conference on Machine Learning, 209–216, 2007.

[7] P. Fearnhead, "Exact and Efficient Bayesian Inference for Multiple Changepoint Problems", Statistics and Computing, 203-213, 2006.

[8] D. Barber, A. T. Cemgil, "Graphical Models for Time-Series", IEEE Signal Processing Magazine, 27(6):18-28, 2010.

[9] http://www.fonality.com/trixbox

[10] B. Kurt, Ç. Yıldız, T. Y. Ceritli, M. Yamaç, M. Semerci, B. Sankur, A. T. Cemgil, "A Probabilistic SIP Network Simulation System", 24th IEEE Conference on Signal Processing and Communications Applications (Submitted), 2016.

[11] http://novacybersecurity.com/en/nova-vspy.html