# Human-in-the-loop Cyber Intrusion Detection Using Active Learning

Yeongwoo Kim*, György Dán* and Quanyan Zhu†
* Division of Network and Systems Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
† Department of Electrical and Computer Engineering, New York University, New York, USA
Email: *{yeongwoo, gyuri}@kth.se, †qz494@nyu.edu

***Abstract*—Timely detection of cyber attacks is essential for minimizing attack impact, but it requires accurate real-time situational awareness (SA). In practice, SA is hampered by frequent false alerts from anomaly-based intrusion detection systems (IDS), causing alarm fatigue. Investigating alerts by humans can enhance SA, but it is resource-intensive and it is often unclear which alerts to prioritize. In this paper, we propose a framework for optimizing human-in-the-loop attack detection, consisting of three key components: 1) dynamic alert prioritization, which ranks alerts based on previous alerts and investigations, 2) human alert investigation, referring to the manual analysis of alerts, and 3) sequential hypothesis testing, a method that confirms a hypothesis based on incoming alerts, with pruned hidden Markov models (HMMs). We formulate the problem as that of active learning in an HMM, and we propose two alert prioritization policies, namely Max Ratio and Max KL. The proposed policies aim to select the most informative alerts based on historical data and prior investigations, thereby minimizing the detection time. Simulation results show that our proposed policies reduce the time to detection by up to $79\%$ compared to a static baseline policy, while maintaining a target mean time between false detections (MTBFD).**

***Index Terms*—Situational awareness, intrusion detection, active learning, hidden Markov model, multihypothesis testing**

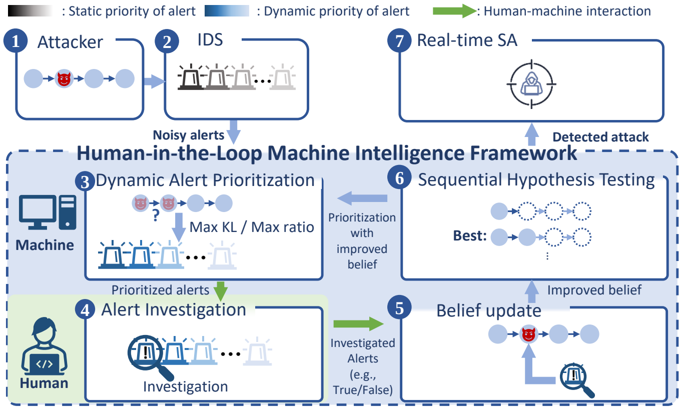
: Static priority of alert
: Dynamic priority of alert
: Human-machine interaction
1 Attacker
2 IDS
7 Real-time SA
Noisy alerts
Detected attack
Human-in-the-Loop Machine Intelligence Framework
3 Dynamic Alert Prioritization
?
Max KL / Max ratio
Machine
Prioritization with improved belief
6 Sequential Hypothesis Testing
Best:
Prioritized alerts
Improved belief
4 Alert Investigation
Human
Investigation
Investigated Alerts (e.g., True/False)
5 Belief update

Fig. 1. Illustration of the proposed human-in-the-loop framework for optimizing attack detection. The machine uses one of two dynamic alert prioritization policies for choosing alerts to investigate, *Max Ratio* or *Max KL*, aiming to minimize the expected time to detection. The human investigates the chosen alerts and returns the investigation outcomes. The machine updates its belief accordingly and chooses the hypothesis that best explains observed alerts and investigations, leading to improved real-time SA.

## I. Introduction

Maintaining real-time situational awareness (SA) through continuous monitoring of the cyber environment is a prerequisite for the timely detection of cyber attacks, thus enabling prompt mitigation of their potential impact [1], [2]. However, maintaining SA has become more challenging due to the increasing frequency and sophistication of attacks, as well as the high rate of false positives from anomaly-based intrusion detection systems (IDSs) [3], [4]. In practice, security operations center (SOC) operators are faced with a deluge of false alerts, leading to alarm fatigue and significant delays in detection [5]. At the same time, relying solely on machine intelligence for detection can result in missed detections [6].

Existing works on improving situational awareness follow one of two approaches. The first approach is to maintain SA by using human expertise more efficiently via different alert prioritization schemes [3], [7]. The second approach is to develop novel machine learning (ML)-algorithms, without accounting for human interaction [8], [9]. These two approaches do unavoidably end up with two pitfalls; on the one hand, the first approach struggles with identifying the most informative alerts to investigate, and on the other hand, the second approach can lead to inaccurate SA due to false alerts [2]. A promising approach to cope with false positives would be to establish a human-in-the-loop machine intelligence system, which effectively integrates human expertise with machine-based decision support. The advantage of such an integrated approach is that it can leverage the capabilities and expertise of human security analysts, combined with the computational advantages of ML-based inference. Even though efficient coordination between human and machine intelligence would be highly desirable, how to establish it is unclear due to the lack of a formal framework.

To address this issue, in this paper we propose an active learning framework that dynamically prioritizes alerts based on their estimated significance, incorporates the outcomes of human investigations, and integrates them into a dynamic detection scheme built upon sequential hypothesis testing. The proposed scheme is illustrated in Fig. 1. The proposed framework consists of two machine learning components and one human component: 1) dynamic alert prioritization policy, 2) human alert investigation, and 3) sequential hypothesis testing. We model the attacker's actions using states in a hidden Markov model (HMM), which may trigger alerts through an intrusion detection system (IDS). The alerts are used for maintaining a belief about the attacker's progression, which is the basis for dynamic alert prioritization formulated as an active learning problem. The prioritized alerts are investigated by security analysts, who report the investigation outcomes (i.e., `True` or `False`) to be used for improving the belief. Since the prioritized alerts depend on the belief, the high-priority alerts can be the most informative alerts, thereby maximizing the performance of sequential hypothesis testing used for deciding whether an attack is ongoing.

Within the framework, we propose two dynamic policies, *Max Ratio* and *Max KL*, aiming at the quickest detection of attacks by maximizing the model's confidence of whether an attack is ongoing. The design of *Max Ratio* and *Max KL* are based on our intuition of sequential hypothesis testing and on analytical results about the sequential probability ratio test, respectively. Upon receiving the chosen alerts from the machine learning component (i.e., our policies), the security analysts (i.e., human) investigate the alerts to identify the root cause, and the outcomes (i.e., `True` or `False`) are used for improving the belief, accounting for potential investigation errors. Then, we employ sequential hypothesis testing to detect potential attacks. As long as we cannot confirm an ongoing attack, we initiate further investigations, and these investigations will benefit from the outcome of previous investigations. This loop results in timely detection of attacks since the dynamic policies repeatedly select the alerts with the highest influence on the confidence of the detection decision. Our main contributions are as follows:

- **Dynamic alert prioritization policies**: We propose two policies for choosing the most informative alerts for timely attack detection. The two policies are based on the expected change in the likelihood ratio, i.e., the ratio of the likelihoods of the observed alerts given two hypotheses and on the Kullback–Leibler (KL) divergence of past alerts, respectively.
- **Active learning framework**: We propose a detection framework where hypotheses are created by pruning an HMM constructed based on an attack graph as depicted in Fig. 1, and active learning is used on the HMMs for minimizing the time to detection. In our framework, active learning results in updating the observation probabilities in the HMMs for past observations based on a confidence function, which takes into account potential investigation errors, effectively supporting timely detection.
- **Evaluation and analysis**: We evaluate the policies on a real-world attack graph with three different confidence functions. Our results show that our policies reduce the time to detection by up to 79% compared to static priorities.

The rest of the paper is structured as follows. Section II discusses related work. The system model and problem formulation are provided in Section III. In Section IV, the proposed policies are formulated, and they are evaluated in Section V. Section VI concludes the paper.

## II. Related Work

Related to ours are works on sequential multihypothesis testing applied to change detection [10]–[13]. Harroua et al. [11] detected denial of service (DoS) attacks using kernel density estimation to obtain a detection threshold. A generalized likelihood ratio test (GLRT) based method was proposed to detect the injection of malicious data in [12]. The authors in [13] modeled the stochastic behavior of the attacker and compared observations from adversarial behavior with those from benign behavior. Common to these works [11]–[13] is that the interaction, e.g., investigation of alerts, is not considered. Considering interactions with the environment, authors in [10] formulated asymptotically optimal tests for a class of Markovian observation models.

A different line of works has attempted to model the security state, i.e., the progression of attackers [14]–[20]. In [14], a partially observable Markov decision process is used to choose defensive actions focusing on a single machine. Assuming that the system state is observable, Iannucci et al. [15], [16] considered that the exploits available to the attacker may be affected by a defense action and used this for optimizing defense. Using Bayesian attack graphs, Miehling et al. [17], [18] addressed optimal incident response in a framework where response affects the exploits available to the attacker, and thus it provides information for improving SA.

Holgado et al. proposed to build a high-level attack graph by merging multiple exploits into a single attack state, justified by that IDSs cannot differentiate between individual exploits [19]. They then used the high-level attack graph to predict the attacker's progression using the forward-backward algorithm. In [20], a high-level attack representation was used to identify mitigation strategies for severe exploits. Javed et al. developed a multi-layered architecture using high-level attack profiles to detect and to mitigate multi-stage attacks in real-time in [21]. In [22], authors suggested a three-step approach relying on a set of high-level attack models to identify the most likely attack scenario. These works do not, however, consider the investigation of alerts or the minimization of the time to detection.

There have been different approaches to the investigation of alerts to improve the defender's SA [1]–[6]. The efficient allocation of alerts to security analysts was studied by modeling the problem as a game in [3]. Dunstatter et al. [4] used dynamic programming and Q-maximin value iteration for efficient alert allocation. The authors in [1] used reinforcement learning to optimize alert investigations considering the shifts of security analysts. While these works improved the efficiency of alert investigation, they rely on static alert priorities computed based on the criticality of alerts. To tackle the limitation of static alert

TABLE I
TABLE OF FREQUENTLY USED NOTATION

| Notation | Definition |
|---|---|
| $\mathcal{S}$ | Set of states, $\mathcal{S} = \{s_i, i \in \mathcal{I}\}$ |
| $S_t$ | State at time $t$ (r.v.) |
| $\pi_i^{t,t_1}$ | Belief $\mathbb{P}(S_{t_1} = s_i)$ computed at time $t$ |
| $Y_t$ | Alert vector observed at time $t$ (r.v.) |
| $B$ | Investigation budget |
| $\omega$ | Investigation error probability |
| $\gamma_0$ | Alert probability adjustment factor |
| $\mathcal{H}$ | Set of hypotheses |
| $\delta_{i,j}$ | Probability of true alert $j$ in state $i$ |
| $\zeta_j$ | Probability of false alert $j$ |
| $Y_t$ | Alerts observed at time $t$ |
| $Y_{1:t}$ | Sequence of alert vectors observed up to time $t$, $Y_{1:t} = (Y_1, \dots, Y_t)$ |
| $o_t^{t_1,j_1}$ | Investigation outcome at time $t$ of alert $j$ observed at time $t'$ |
| $O_{1:t}$ | Sequence of investigation outcomes up to time $t$ |
| $\mathcal{V}_t$ | Alert(s) to investigate at time $t$ |
| $\mathcal{V}_{1:t}$ | Alert investigations performed up to time $t$ |

priorities, authors in [2] studied the dynamic prioritization of alerts based on their potential information gain. The proposed framework allows the investigation of a subset of the most recent alerts with noisy investigation outcomes. While different from the traditional active learning framework for HMMs [23], which allows additional observations about the system state, the approach in [2] does not scale due to the state representation, and it does not consider the issue of intrusion detection. Shah et al. aimed to investigate all alerts under the tradeoff between the cost to hire analysts and the ideal mix of security analysts' expertise levels [5]. However, investigating all alerts may be impossible, depending on the size of the network. In their following work [6], they used the human-defined significance of each alert to compute the composite risk score and allocated alerts to security analysts. Although this approach dynamically calculates the risk score, it still relies on the human-defined risk score.

Unlike previous works where a large volume of alerts leads to alarm fatigue and human-defined risk score causes significant delays in detection, we propose a framework and algorithms for minimizing the expected time to attack detection through active learning, exploring a novel combination of sequential hypothesis testing with active learning in a HMM. Our framework presents a novel combination of machine learning and human competence for maximizing cyber situational awareness.

## III. SYSTEM MODEL AND PROBLEM FORMULATION

We consider the interaction between an attacker, whose aim is to compromise the system, and a defender, whose aim is to detect the attacker in the system. We consider that time is slotted and use $t \in \mathbb{Z}^+$, where $\mathbb{Z}^+$ is the set of nonnegative integers, as time index.

### A. Attack model

We model the progression of the attacker in the system by an attack graph. The attack graph consists of nodes and edges, where each node corresponds to an attack state (i.e., security state), and involves adversarial activity by the attacker, e.g., according to the MITRE Att&ck model [24]. We denote by $\mathcal{S} = \{s_1, \dots, s_I\}$ the set of attack states, and by $i \in \mathcal{I} = \{1, \dots, I\}$ the index of the attack state where $|\mathcal{S}| = I$. An edge $e_{i,i'} = (s_i, s_{i'})$ means the attacker's transition from state $s_i$ to state $s_{i'}$. We denote by $a_{s_i s_{i'}} = \mathbb{P}(S_{t+1} = s_{i'} | S_t = s_i)$ the state transition probability from state $s_i$ to state $s_{i'}$.

The adversarial activity in state $s_i$ may trigger a set of alerts $\mathcal{J}_i \subseteq \mathcal{J}$ by IDS, where $\mathcal{J} = \{1, \dots, J\}$ is the set of alert indices and $J$ is the number of distinct alerts. The probability that the attacker's activity triggers alert $j \in \mathcal{J}_i$ in security state $s_i$ at time $t$ is $\delta_{ij} = \mathbb{P}(Y_t^j = 1 | S_t = s_i) > 0$ where $Y_t^j$ is the status of alert $j$ at time slot $t$. In addition to true alerts, the IDS raises false alerts with probability $\zeta_j$ regardless of the security state since the false alerts are triggered by legitimate users' activities not relevant to the attacker's security state. Note that the stochastic processes $(S_t, Y_t)$ form a hidden Markov model (HMM), and we denote by $\lambda$ the parameters of the HMM.

### B. Defender Model

The defender can observe alerts triggered by the IDS. We denote by $Y_t = \{Y_t^1, \dots, Y_t^j \dots, Y_t^J\} \in \{0,1\}^J$ the alerts observed during time slot $t$, where $Y_t^j = 1$ corresponds to a positive alert, and $Y_t^j = 0$ corresponds to a negative alert. We denote by $Y_{1:t} = \{Y_1, \dots, Y_t\}$ the alert vectors observed until time $t$.

In addition, in each time slot $t$, the defender uses an alert prioritization policy for choosing a set $\mathcal{V}_t \subseteq Y_{1:t}^+ = \{(t_1, j_1) | Y_{t_1}^j = 1, 0 \leq t_1 \leq t, j_1 \in \mathcal{J}\}$ of positive alerts that the human security analysts should investigate, subject to its investigation budget $|\mathcal{V}_t| \leq B$. The same positive alert can be investigated several times in subsequent time slots since the alert investigation is an error-prone process in practice. Hence, we model this as follows. Let us consider the investigation of alert $Y_{t_1}^{j_1}$ at time $t$. For a false positive alert, the investigation outcome is $o_t^{t_1,j_1} = 1$ with probability $\omega$, and $o_t^{t_1,j_1} = 0$ with probability $1 - \omega$. For a true positive alert, the investigation outcome is $o_t^{t_1,j_1} = 0$ with probability $\omega$, and $o_t^{t_1,j_1} = 1$ with probability $1 - \omega$. We refer to $\omega$ as the investigation error probability of a security analyst assuming the uniform error on all alerts $j_1 \in \mathcal{J}$ for simplicity. The defender is not aware of the ground truth, but it has an estimate of $\omega$. Our model of error-prone alert investigations can model human security analysts and potential future machine learning algorithms capable of alert investigation. In our model both are characterised by their respective investigation error probability, which may of course be different.

Fig. 2 shows how the investigation outcome influences the observation probabilities. Depending on the outcome $o_t^{t_1,j_1}$ of the investigation of alert $Y_{t_1}^{j_1}$ at time $t$, the defender updates the likelihood of the alert being a false positive,

$$\zeta_t^{t_1,j_1} = \begin{cases} \min(\gamma(\omega)\zeta_{t-1}^{t_1,j_1}, 1) & \text{for } o_t^{t_1,j_1} = 0, \\ \frac{1}{\gamma(\omega)}\zeta_{t-1}^{t_1,j_1} & \text{for } o_t^{t_1,j_1} = 1, \end{cases} \tag{1}$$

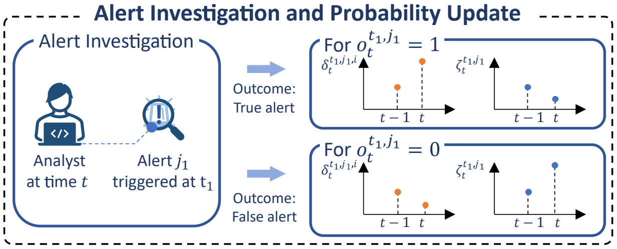

Fig. 2. Human security analysts investigate the prioritized alerts. The outcome of the alert investigation is used for updating the observation probabilities and hence the belief about the attacker's progression.

where $\zeta_{t_1-1}^{t_1,j_1} = \zeta_{j_1}$, i.e., the raw probability of false alert $j_1$, and $\gamma(\omega)$ is the confidence function defined for $\omega \in [0, 0.5]$. Intuitively, $\gamma(\omega)$ captures the confidence attributed to an investigation outcome, as a function of the investigation error probability. To obtain insight into how to design this function, in our model we consider linear, concave, and convex functions as alternatives

$$\gamma(\omega) = \begin{cases} 2(1-\gamma_0)\omega + \gamma_0, & (linear), \\ 4(1-\gamma_0)\omega^2 + \gamma_0, & (concave), \\ 4(\gamma_0 - 1)(\omega - 0.5)^2 + 1 & (convex), \end{cases} \tag{2}$$

where $\gamma_0 > 1$ is a parameter. These alternatives allow us to capture different models of confidence in error-prone investigation outcomes; e.g., a convex function would imply that confidence decreases faster than linear as the error probability increases, while a concave function implies a slower than linear decrease. Similarly, the defender updates the probability of being a true positive alert as

$$\delta_t^{t_1,j_1,i} = \begin{cases} \frac{1}{\gamma(\omega)}\delta_{t-1}^{t_1,j_1,i} & \text{for } o_t^{t_1,j_1} = 0, \\ \min(\gamma(\omega)\delta_{t-1}^{t_1,j_1,i}, 1) & \text{for } o_t^{t_1,j_1} = 1, \end{cases} \tag{3}$$

where $\delta_{t_1-1}^{t_1,j_1,i} = \delta_{ij_1}$, i.e., the raw probability of true alert $j_1$ in state $i$.

Given the outcomes of the alert investigations and conditioned on the attacker being in state $s_i$, we can express the likelihood of alert observations as

$$\begin{aligned} &\mathbb{P}(Y_{t_1} = y_{t_1} | O_{1:t} = o_{1:t}, \mathcal{V}_{1:t} = v_{1:t}, S_{t_1} = s_i) \\ &\quad = \prod_{j=1}^{J} \mathbb{P}(Y_{t_1}^{j_1} = y_t^j | O_{1:t} = o_{1:t}, \mathcal{V}_{1:t} = v_{1:t}, S_{t_1} = s_i), \end{aligned} \tag{4}$$

where each term is the likelihood of observing an individual alert, we denote by $\mathcal{V}_{1:t} = \{\mathcal{V}_1, \ldots, \mathcal{V}_t\}$ the investigated alerts up to time slot $t$, and by $O_{1:t} = \{O_1, \ldots, O_t\}$ the outcome of the investigations up to time slot $t$ where $O_t = \{O_t^{t_1,j_1} | (t_1, j_1) \in \mathcal{V}_t\}$. Observe that the individual alerts are conditionally independent given the attack state and previous investigations, hence the product form. We can express the likelihood of an individual alert as

$$\begin{aligned} \mathbb{P}(Y_{t_1}^{j_1} &= y_{t_1}^{j_1} | O_{1:t} = o_{1:t}, \mathcal{V}_{1:t} = v_{1:t}, S_{t_1} = s_i) \\ &= \begin{cases} (1 - \zeta_t^{t_1,j_1})(1 - \delta_t^{t_1,j_1,i}) & \text{if } y_{t_1}^{j_1} = 0, \\ 1 - (1 - \zeta_t^{t_1,j_1})(1 - \delta_t^{t_1,j_1,i}) & \text{if } y_{t_1}^{j_1} = 1. \end{cases} \end{aligned}$$

We denote by $\mathcal{F}_t = \{Y_{1:t}, O_{1:t-1}, \mathcal{V}_{1:t-1}\}$ the information available to the defender at time $t$, before choosing the alerts $\mathcal{V}_t$ to investigate, consisting of past alerts, investigation outcomes and investigation decisions.

The defender uses the multihypothesis sequential generalized probability ratio test (MSGPRT) to the detect the attacker's progression based on the observations [10], [25], [26]. The defender does so by maintaining a set $\mathcal{H}$, $|\mathcal{H}| = H$ of hypotheses, where each hypothesis $h \in \mathcal{H}$ corresponds to an HMM model $\lambda_h$ generating the observations. Fig. 4 illustrates the hypothesis models constructed by iteratively removing attack states from the complete attack graph $\lambda$, and we denote by $\mathcal{I}_h \subseteq \mathcal{I}$ the indices of states included in hypothesis $h$.

Let us denote by $p_h(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t) = \mathbb{P}(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t, \Lambda_t = \lambda_h)$. Then for hypothesis $\hat{h} = \arg\max_{h \in \mathcal{H}} p_h(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t)$, the defender computes the probability ratio with respect to hypothesis 1 as

$$R_{t|\hat{h}}^{v_t,o_t} = \frac{p_{\hat{h}}(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t)}{p_1(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t)}, \tag{5}$$

and compares it to the detection threshold $\theta_{\hat{h}}$, which results in a decision

$$\eta_t = \begin{cases} \hat{h} & \text{if } R_{t|\hat{h}}^{v_t,o_t} > \theta_{\hat{h}}, \\ \emptyset & \text{otherwise}, \end{cases}$$

where $\emptyset$ means undefined. We can then define the detection time for model $\lambda_{\hat{h}}$ as

$$d^{\hat{h}} = \min\left\{ t | \eta_t = \hat{h} \right\}.$$

### C. Problem Formulation

Let us denote by $\kappa$ the policy of the defender for choosing the set $\mathcal{V}_t$ of alerts to be investigated by human security analysts. We make the reasonable assumption that the policy is causal, i.e., the choice of $\mathcal{V}_t$ is based on $\mathcal{F}_t$,

$$\kappa : \mathcal{F}_t \rightarrow {Y_{1:t}^{+}}^{B}. \tag{6}$$

The defender would like to detect the attacker's progression to state $s_2$, which it does by confirming any hypothesis model $\lambda_h$, $h > 1$. We thus define the detection time under policy $\kappa$ as that of any hypothesis $h > 1$,

$$d^{\kappa} = \min_{h > 1} d^{h,\kappa}.$$

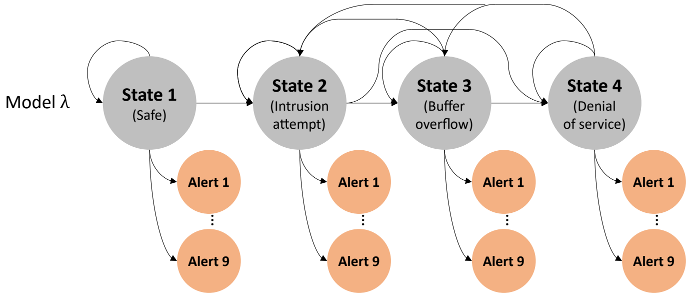

Fig. 3. A HMM $\lambda$ modeling a DoS attack with four states and with nine alerts. The attack state (i.e., grey node) describes the attacker's progression, and the alerts (i.e., orange node) are the observations from states.

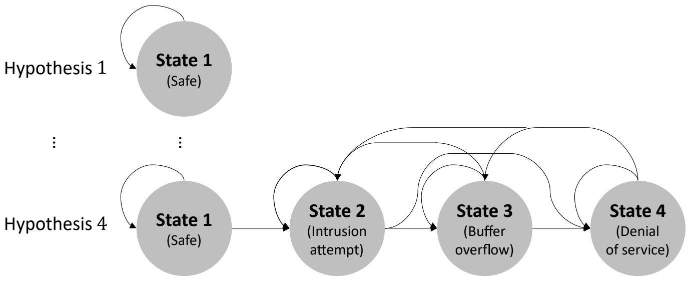

Fig. 4. The pruned models are constructed by iteratively removing a state from the original model $\lambda$.

The objective of the defender is to find a policy that minimizes the expected detection latency, i.e.,

$$\kappa^* = \arg\min_{\kappa\in\mathcal{K}} \sup_{t_{s_1\to s_2}>0} \mathbb{E}^{(t_{s_1\to s_2})}[d^\kappa - t_{s_1\to s_2}], \qquad (7)$$

subject to a constraint on the time between false alerts

$$\mathbb{E}^{(\infty)}[d^\kappa] \geq \tau,$$

where $E^{(t_{s_1\to s_2})}$ and $E^{(\infty)}$ denote the expectations when the true state transition happens at time $t_{s_1\to s_2}$ and when there is no state transition at all, respectively. The problem faced by the defender is an active learning problem for quickest change detection in an HMM.

## IV. Proposed Policies

In what follows, we first outline the sequential probability ratio test (PRT) underlying the proposed policies, we then propose the two dynamic alert prioritization policies to solve (6) subject to (7), i.e., policies for choosing the set $\mathcal{V}_t$ of alerts to be investigated by human security analysts.

### A. Likelihood ratio and sequential probability ratio test (SPRT)

Let us denote by $p^{t,t_1}_{i|h} = \mathbb{P}(Y_{1:t_1} = y_{1:t_1}, S_{t_1} = s_i | \mathcal{F}_t = f_t, \Lambda_{t_1} = \lambda_h)$ the joint probability that the observed alerts up to time $t_1$ are $y_{1:t_1}$ and the state at time $t_1$ is $s_i$ given that the hypothesis is $\lambda_h$ at time $t$ and given past alerts and investigations $f_t$. The defender can compute $p^{t,t_1}_{i|h}$ using the forward algorithm in a recursive manner [27], obtaining the lower triangular matrices

$$p^{t,t_1} = \begin{bmatrix} p^{t,t_1}_{1|1} & 0 & \dots & 0 \\ p^{t,t_1}_{1|2} & p^{t,t_1}_{2|2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ p^{t,t_1}_{1|H} & p^{t,t_1}_{2|H} & \dots & p^{t,t_1}_{I|H} \end{bmatrix},$$

where each column corresponds to a state, each row corresponds to a hypothesis, and $1 \leq t_1 \leq t$. The matrices can be computed recursively, hence the update is computationally efficient. We can use the likelihoods $p^{t,t_1}_{i|h}$ to express the likelihood ratio defined in (5) as

$$R^{\hat{h}}_t = \frac{\sum_{i=1}^{I} p^{t,t_1}_{i|\hat{h}}}{\sum_{i'=1}^{I} p^{t,t_1}_{i'|1}}.$$

In addition, assuming a uniform prior about the hypotheses, we can express the belief $\mathbb{P}(S_{t_1} = s_i | \mathcal{F}_t = f_t)$ at time $t$ about the state $S_{t_1}$ at time $t_1 \leq t$ as

$$\pi^{t,t_1}_i = \frac{\sum_{h=1}^{H} p^{t,t_1}_{i|h}}{\sum_{i'=1}^{I}\sum_{h'=1}^{H} p^{t,t_1}_{i'|h'}}. \qquad (8)$$

We will use the above likelihoods and beliefs for defining the alert investigation policies, but before we do so, let us recall some fundamental concepts related to the sequential PRT (SPRT) under i.i.d. observations.

**Definition 1** (Risk). *Consider a HMM with $\mathcal{S} = \{s_1, \dots, s_I\}$, $a_{s_i s_{i'}} = 0$, i.e., no state transitions, and hypotheses $h_i = \{S_0 = s_i\}$, i.e., i.i.d. observations. The probability of error (misdetection) under hypothesis $h$ at time $t^h_{d_\kappa}$ is defined as*

$$P_{\max} \triangleq \max_{h\in\mathcal{H}} \mathbb{P}_h\{\eta_{t^{h'}_{d_\kappa}} \neq h\},$$

*and the risk of incorrectly confirming hypothesis $h$ is defined as*

$$R_h \triangleq \max_{h'\neq h} \mathbb{P}_{h'}\{\eta_{t^h_{d_\kappa}} = h\}.$$

**Definition 2.** *The KL divergence of distribution $p$ from distribution $q$, both defined over discrete set $\mathcal{X}$, is*

$$D(p||q) = \sum_{x\in\mathcal{X}} p(x)\frac{\log p(x)}{\log q(x)}. \qquad (9)$$

Using these definitions, let us recall a fundamental result about the expected time to detection of the SPRT under i.i.d observations [10], [28].

**Lemma 1** ( [28] Chapter 3.2)**.** *Consider an HMM with $\mathcal{S} = \{s_1, \dots, s_I\}$, $a_{s_i s_{i'}} = 0$, and hypotheses $h_i = \{S_0 = s_i\}$, i.e., i.i.d. observations. Let $D(P_h(Y_t) \parallel P_{h'}(Y_t))$ be the KL divergence of the likelihood of observation vector $Y_t$ without*

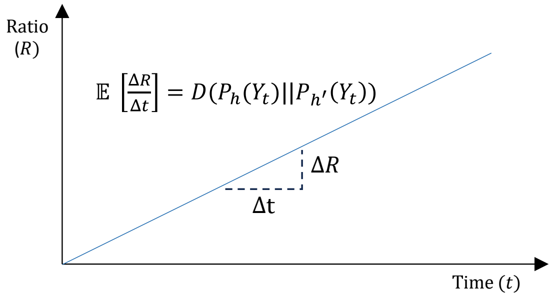

Fig. 5. Relation between KL divergence of observations under two hypotheses $h$ and $h'$ and the slope of the likelihood ratio as a function of the number of i.i.d. observations.

*investigating alerts under hypothesis $h$ and $h'$. Then, the expected time to detection is given by*

$$\mathbb{E}[t_d^h] = \frac{-\log(R_h)}{\min_{h' \neq h} D(P_h(Y_t) \parallel P_{h'}(Y_t))}(1 + o(1)).$$

The above result shows that if observations are i.i.d., then the expected time to detection is inversely proportional to the KL divergence of the distribution of the observed alerts under different hypotheses, as illustrated in Fig. 5. Another interpretation of the result is that the expected change of the probability ratio per time step equals the KL divergence of the distribution of the observed alerts under different hypotheses. In our model, the observations are not i.i.d. due to the dependence of the observation probabilities on the state of the HMM, yet these results provide important intuition for the design of our alert prioritization policies.

Our framework combines the alert prioritization policies with the defender model to detect a potential intrusion. Let us recall that alert investigations update the observation probabilities in (1) and (3), leading to changes in the KL divergence and in the likelihood ratio. Our proposed policies (i.e., the machine) rank the alerts based on the changed KL divergence or likelihood ratio. Security analysts (i.e., the human) then investigate the highest ranked alerts and return the investigation outcomes to the machine. This interaction between machine and human is repeated until the likelihood ratio in (5) reaches the detection threshold.

### *B. Max KL divergence (MKL) policy*

The first alert prioritization policy we propose is based on the above asymptotic result on the relationship between the expected time to detection and the risk of incorrectly confirming a hypothesis. The key tenet of the proposed max KL (MKL) divergence policy is to choose alerts to investigate, aiming to maximize the KL divergence of the distribution of the observed alerts after the investigation.

For formulating the policy, let $\hat{h} = \arg\max_{h \in \mathcal{H} \setminus \{1\}} p_h(Y_{1:t} = y_{1:t} | \mathcal{F}_t = f_t)$ be the most likely hypothesis at time $t$ prior to alert investigation, and $p_h(Y_{t'} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t) = \mathbb{P}(Y_{t'} | \mathcal{F}_t = f_t, O_t = o_t, \mathcal{V}_t = v_t, \Lambda_t = \lambda_h)$ the distribution of observations $Y_{t'}$ under hypothesis $h$, given past observations and investigation outcomes $f_t$ and the choice $v_t$ of alert investigations at time $t$. Then, at time $t$, our objective is to choose a set of investigations that solves

$$\begin{aligned} \max_{v_t \subseteq Y_{1:t}^+, |v_t| \leq B} \; & \mathbb{E}\Big[ \sum_{t'=1}^{t} D\big(p_{\hat{h}}(Y_{t'} | \mathcal{F}_t = f_t, \mathcal{V}_t = v_t) \\ & \qquad \qquad \parallel p_1(Y_{t'} | \mathcal{F}_t = f_t, \mathcal{V}_t = v_t)\big)\Big] \\ & - \sum_{t'=1}^{t} D(p_{\hat{h}}(Y_{t'} | \mathcal{F}_t = f_t) \| p_1(Y_{t'} | \mathcal{F}_t = f_t)), \end{aligned} \tag{10}$$

where the expectation in (10) is taken with respect to the outcome $O_t$ of the investigation of alerts $v_t$.

For ease of exposition, let us first consider a single investigation $|\mathcal{V}_t| = 1$, and let $\mathcal{V}_t = \{(t_1, j_1)\}$, for some $0 \leq t_1 \leq t$. Recall that the investigation outcome $O_t^{t_1, j_1}$ is a random variable, and its realization influences the distribution of the observations made at times $t \geq t_1$, hence the KL divergence of the observations under different hypotheses. We can express the expected KL divergence of the observations at time $t$ when investigating alert $j_1$ observed at time $t_1$ as

$$\begin{aligned} & \mathbb{E}\big[D(p_{\hat{h}}(Y_t | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}) \\ & \qquad \qquad \parallel p_1(Y_t | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}))\big] \\ & = \sum_{o_t^{t_1, j_1} \in \{0,1\}} \mathbb{P}(O_t^{t_1, j_1} = o_t^{t_1, j_1} | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}) \\ & \quad \cdot D\big(p_{\hat{h}}(Y_t | \mathcal{F}_t = f_t, O_t^{t_1, j_1} = o_t^{t_1, j_1}, \mathcal{V}_t = \{(t_1, j_1)\}) \\ & \qquad \parallel p_1(Y_t | \mathcal{F}_t = f_t, O_t^{t_1, j_1} = o_t^{t_1, j_1}, \mathcal{V}_t = \{(t_1, j_1)\})\big), \end{aligned} \tag{11}$$

where the probability of the investigation outcome can be computed as

$$\begin{aligned} & \mathbb{P}(O_t^{t_1, j_1} = o_t^{t_1, j_1} | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}) \\ & = \sum_{i=1}^{I} \mathbb{P}(O_t^{t_1, j_1} = o_t^{t_1, j_1} | S_{t_1} = s_i, \mathcal{V}_t = \{(t_1, j_1)\}) \cdot \pi_i^{t, t_1}, \end{aligned} \tag{12}$$

and $\pi_i^{t, t_1}$ is given by (8). Based on the investigation error probability $\omega$, we can write

$$\begin{aligned} & \mathbb{P}(O_t^{t_1, j_1} = o_t^{t_1, j_1} | S_t = s_i, \mathcal{V}_t = \{(t_1, j_1)\}) \\ & = \begin{cases} \frac{\zeta_t^{t_1, j_1}}{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i})}(1 - \omega) \\ + \frac{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i}) - \zeta_t^{t_1, j_1}}{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i})}\omega, & \text{for } o_t^{t_1, j_1} = 0, \\ \\ \frac{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i}) - \zeta_t^{t_1, j_1}}{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i})}(1 - \omega) \\ + \frac{\zeta_t^{t_1, j_1}}{1 - (1 - \zeta_t^{t_1, j_1})(1 - \delta_t^{t_1, j_1, i})}\omega, & \text{for } o_t^{t_1, j_1} = 1. \end{cases} \end{aligned}$$

Evaluating the KL divergence in (11) is computationally challenging as the set of possible observations grows exponentially with the number $J$ of alerts. In what follows we provide results that enable efficient computation of the KL divergence under certain conditions.

**Proposition 1.** *Consider alert vector $Y_{1:t}$. The KL divergence of $Y_{1:t}$ under hypotheses $\hat{h}$ and $1$ satisfies*

$$
\begin{aligned}
&\sum_{t_1=1}^{t} D(p_{\hat{h}}(Y_{t_1} = y_{t_1} | \mathcal{F}_t = f_t) \parallel p_1(Y_{t_1} = y_{t_1} | \mathcal{F}_t = f_t)) \\
&\qquad = \sum_{t_1=1}^{t} \sum_{j=1}^{J} D(p_{\hat{h}}(Y_{t_1}^{j} = y_{t_1}^{j} | \mathcal{F}_t = f_t) \\
&\qquad\qquad\qquad\qquad \parallel p_1(Y_{t_1}^{j} = y_{t_1}^{j} | \mathcal{F}_t = f_t)),
\end{aligned}
$$

*Proof.* We prove the result by showing that alerts at time $t_1$ are independent. Let us consider alert vector $Y_{t_1}$, and consider alerts $Y_{t_1}^{j_1}$ and $Y_{t_1}^{j_2}$, $j_1, j_2 \in \mathcal{J}$, with probability given by (4). For $h \in \mathcal{H}$, we then have

$$
\begin{aligned}
&p_h(Y_{t_1}^{j_1} = y_{t_1}^{j_1}, Y_{t_1}^{j_2} = y_{t_1}^{j_2} | \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} p_h(Y_{t_1}^{j_1} = y_{t_1}^{j_1}, Y_{t_1}^{j_2} = y_{t_1}^{j_2} | S_{t_1} = s_i, \mathcal{F}_t = f_t) \\
&\qquad \cdot \mathbb{P}(S_{t_1} = s_i | \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} p_h(Y_{t_1}^{j_1} = y_{t_1}^{j_1} | S_{t_1} = s_i, \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(Y_{t_1}^{j_2} = y_{t_1}^{j_2} | S_{t_1} = s_i, \mathcal{F}_t = f_t) p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} [p_h(Y_{t_1}^{j_1} = y_{t_1}^{j_1} | S_{t_1} = s_i, \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t)] \\
&\qquad \cdot \sum_{s_i \in \mathcal{S}} [p_h(Y_{t_1}^{j_2} = y_{t_1}^{j_2} | S_{t_1} = s_i, \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t)] \\
&\quad = p_h(Y_{t_1}^{j_1} = y_{t_1}^{j_1} | \mathcal{F}_t = f_t) p_h(Y_{t_1}^{j_2} = y_{t_1}^{j_2} | \mathcal{F}_t = f_t).
\end{aligned}
$$

The independence of alerts $Y_t^{j}$ and $Y_t^{j'}$ for $j \neq j'$, together with the chain rule of KL divergence [29] implies that the KL divergence under hypotheses $\hat{h}$ and 1 at time $t_1$ satisfies

$$
\begin{aligned}
&D(p_{\hat{h}}(Y_{t_1} = y_{t_1} | \mathcal{F}_t = f_t) \parallel p_1(Y_{t_1} = y_{t_1} | \mathcal{F}_t = f_t)) \\
&\quad = \sum_{j=1}^{J} D(p_{\hat{h}}(Y_{t_1}^{j} = y_{t_1}^{j} | \mathcal{F}_t = f_t) \parallel p_1(Y_{t_1}^{j} = y_{t_1}^{j} | \mathcal{F}_t = f_t)),
\end{aligned}
$$

which proves the result. □

Next, we show the independence of the investigation outcomes for different alerts observed at the same time slot.

**Proposition 2.** *Let $t_1 \leq t$. Then, the investigation outcome $O_t^{t_1,j_1}$ for alert $j_1$ observed at time $t_1$ is independent of the investigation outcome $O_t^{t_1,j_2}$ of alert $j_2$ observed at time $t_1$.*

*Proof.* Let us consider alerts $Y_{t_1}^{j_1}$ and $Y_{t_1}^{j_2}$, $j_1, j_2 \in \mathcal{J}$, with probability given by (4), and let $v_t = \{(t_1, j_1), (t_1, j_2)\}$. Under hypothesis $h \in \mathcal{H}$, the joint probability of the investigation outcomes is

$$
\begin{aligned}
&p_h(O_t^{t_1,j_1} = o_t^{t_1,j_1}, O_t^{t_1,j_2} = o_t^{t_1,j_2} | \mathcal{V}_t = v_t, \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} p_h(O_t^{t_1,j_1} = o_t^{t_1,j_1}, O_t^{t_1,j_2} = o_t^{t_1,j_2} | S_{t_1} = s_i, \\
&\qquad \mathcal{V}_t = v_t, \mathcal{F}_t = f_t) \\
&\qquad \cdot \mathbb{P}(S_{t_1} = s_i | \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} p_h(O_t^{t_1,j_1} = o_t^{t_1,j_1} | S_{t_1} = s_i, \mathcal{V}_t = \{(t_1, j_1)\}, \\
&\qquad \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(O_t^{t_1,j_2} = o_t^{t_1,j_2} | S_{t_1} = s_i, \mathcal{V}_t = \{(t_1, j_2)\}, \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t) \\
&\quad = \sum_{s_i \in \mathcal{S}} [p_h(O_t^{t_1,j_1} = o_t^{t_1,j_1} | S_{t_1} = s_i, \mathcal{V}_t = \{(t_1, j_1)\}, \\
&\qquad \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t)] \\
&\qquad \cdot \sum_{s_i \in \mathcal{S}} [p_h(O_t^{t_1,j_2} = o_t^{t_1,j_2} | S_{t_1} = s_i, \mathcal{V}_t = \{(t_1, j_2)\}, \\
&\qquad \mathcal{F}_t = f_t) \\
&\qquad \cdot p_h(S_{t_1} = s_i | \mathcal{F}_t = f_t)] \\
&\quad = p_h(O_t^{t_1,j_1} = o_t^{t_1,j_1} | \mathcal{V}_t = \{(t_1, j_1)\}, \mathcal{F}_t = f_t) \\
&\qquad\quad \cdot p_h(O_t^{t_1,j_2} = o_t^{t_1,j_2} | \mathcal{V}_t = \{(t_1, j_2)\}, \mathcal{F}_t = f_t).
\end{aligned}
$$

Thus, the investigation outcomes are independent. □

An important consequence of Proposition 1 and Proposition 2 is the following.

**Proposition 3.** *Let $v_t = \{(t_1, j_1), (t_1, j_2)\}$. Then the expected change of the KL divergence is additive up to time $t_1$, i.e.,*

$$
\begin{aligned}
&\mathbb{E}[\sum_{t'=1}^{t_1} D(p_h(Y_{t'} | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1), (t_1, j_2)\}) \\
&\qquad\qquad \parallel p_1(Y_{t'} | \mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1), (t_1, j_2)\}))] \\
&\quad = \sum_{t'=1}^{t_1} D(p_h(Y_{t'} | \mathcal{F}_t = f_t) \| p_1(Y_{t'} | \mathcal{F}_t = f_t)) \\
&\qquad + \Delta\overline{D}_{t_1}^{t_1,j_1} + \Delta\overline{D}_{t_1}^{t_1,j_2}.
\end{aligned}
$$

*Proof.* Let us consider the investigation outcomes $O_t^{t_1,j_1}$ and $O_t^{t_1,j_2}$. Observe that the investigation outcomes $O_t^{t_1,j_1}$ and $O_t^{t_1,j_2}$ only affect the probability of the corresponding alerts

**Algorithm 1:** MKL policy

```
Input: Past observations $f_t = (y_{1:t}, o_{1:t-1}, v_{1:t-1})$;
       Investigation budget $B$
Output: $v_t$ (Alerts to investigate)
1  $n_A = 0$; $\vec{v} = \emptyset$
2  for $t_1 \leftarrow 1$ to $t$ do
3      for $j_1 \leftarrow 1$ to $J$ do
4          if $Y_{t_1}^{j_1} = 1$ then
5              Compute $\Delta\overline{D}_{t_1}^{t_1,j_1}$ using (16)
6              $\vec{v}$.insert($(t_1, j_1)$)
7              $n_A + +$
8      end
9  end
10 Sort $\vec{v}$ in the descending order of $\Delta\overline{D}_{t_1}^{t_1,j_1}$
11 $v_F = \{\vec{v}(1), \ldots, \vec{v}(B)\}$
   $v_L = \{\vec{v}(n_A - B + 1), \ldots, \vec{v}(n_A)\}$
   $\sigma_F = \sum_{(t_1,j_1)\in v_F} \Delta\overline{D}_{t_1}^{t_1,j_1}$
   $\sigma_L = \sum_{(t_1,j_1)\in v_L} \Delta\overline{D}_{t_1}^{t_1,j_1}$
12 if $|\sigma_F| \geq |\sigma_L|$ then
13     $v_t = v_F$
14 else
15     $v_t = v_L$
16 end
```

$Y_{t_1}^{j_1}$ and $Y_{t_1}^{j_2}$, respectively, we can thus write

$$\begin{aligned}
&\mathbb{E}[\sum_{t'=1}^{t_1} D(p_h(Y_{t'} = y_{t'}|\mathcal{F}_t = f_t, \mathcal{V}_t\})\\
&\qquad\qquad \| \; p_1(Y_{t'} = y_{t'}|\mathcal{F}_t = f_t, \mathcal{V}_t))]\\
&\quad = \sum_{t'=1}^{t_1}\sum_{j=1}^{J} D(p_h(Y_{t'}^{j} = y_{t'}^{j}|\mathcal{F}_t = f_t) \\
&\qquad\qquad \| p_1(Y_{t'}^{j} = y_{t'}^{j}|\mathcal{F}_t = f_t)) \qquad (13)\\
&\qquad - D(p_h(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t)\|p_1(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t))\\
&\qquad - D(p_h(Y_{t_1}^{j_2} = 1|\mathcal{F}_t = f_t)\|p_1(Y_{t_1}^{j_2} = 1|\mathcal{F}_t = f_t))\\
&\qquad + \mathbb{E}[D(p_h(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})\\
&\qquad\qquad \|p_1(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}))] \qquad (14)\\
&\qquad + \mathbb{E}[D(p_h(Y_{t_1}^{j_2} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_2))\\
&\qquad\qquad \|p_1(Y_{t_1}^{j_2} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_2)\}))], \qquad (15)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{t'=1}^{t_1}\sum_{j=1}^{J} D(p_h(Y_{t'}^{j} = y_{t'}^{j}|\mathcal{F}_t = f_t)\|p_1(Y_{t'}^{j} = y_{t'}^{j}|\mathcal{F}_t = f_t))\\
&\quad + \Delta\overline{D}_{t_1}^{t_1,j_1} + \Delta\overline{D}_{t_1}^{t_1,j_2},
\end{aligned}$$

where (13) follows from Proposition 1 and (14) and (15) follow from Proposition 2. □

Thus, the change of the KL divergence is additive when investigating alerts that occurred during the same time slot. Additivity does not hold in general, i.e., the expected change due to investigations of alerts observed at different time slots is not additive. Nonetheless, for computational efficiency, we propose to approximate the expected change $\Delta\overline{D}_{t}^{t_1,j_1}$ by the expected change $\Delta\overline{D}_{t_1}^{t_1,j_1}$, based on the assumption that the expected change in the KL divergence is the same regardless of whether the alert is investigated immediately upon receiving an observation or at a later time, i.e., $\Delta\overline{D}_{t}^{t_1,j_1} \approx \Delta\overline{D}_{t_1}^{t_1,j_1}$.

Algorithm 1 shows the pseudocode of the proposed MKL policy. At time $t$, the policy considers the expected change of the KL divergence for each positive alert (Line 5),

$$\begin{aligned}
\Delta\overline{D}_{t_1}^{t_1,j_1} &= \mathbb{E}\big[D(p_{\hat{h}}(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})\\
&\qquad \| \; p_1(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\}))\big]\\
&\quad - D(p_{\hat{h}}(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t) \,\|\, p_1(Y_{t_1}^{j_1} = 1|\mathcal{F}_t = f_t)).
\end{aligned} \tag{16}$$

The policy then sorts the alerts in descending order of $\Delta\overline{D}_{t_1}^{t_1,j_1}$ (Line 10), and computes the sum of the first $B$ alerts and the sum of the last $B$ alerts (Line 11). It then picks the $B$ alerts with the largest absolute sum value (Lines 12 to 16), which are the alerts to be investigated by the security analysts at time $t$.

### C. Max Ratio (MR) policy

The proposed max ratio (MR) policy builds on the intuition that maximization of the likelihood ratio through investigations would minimize the time to detection, as it would expedite reaching the detection threshold. To define the policy, let us define the random variable

$$R_{t|\hat{h}}^{v_t} = \frac{p_{\hat{h}}(Y_{1:t} = y_{1:t}|\mathcal{F}_t = f_t, \mathcal{V}_t = v_t)}{p_1(Y_{1:t} = y_{1:t}|\mathcal{F}_t = f_t, \mathcal{V}_t = v_t)},$$

i.e., the probability ratio as a function of the investigation outcome if the defender investigates alert $v_t$. As before, we denote by $\hat{h} = \arg\max_{h\in\mathcal{H}\setminus\{1\}} p_h(Y_{1:t} = y_{1:t}|\mathcal{F}_t = f_t)$ the most likely hypothesis at time $t$, prior to the investigation. Formally, the MR policy $\kappa^{MR}$ at time slot $t$ aims to choose a set of alerts that solves

$$\max_{v_t\subseteq Y_{1:t}^{+}, |v_t|\leq B} \left|\mathbb{E}\left[R_{t|\hat{h}}^{v_t}\right]\right|, \tag{17}$$

where the expectation is taken with respect to the distribution of the investigation outcomes, given in (12).

Computing the expectation in (17) is challenging for two reasons. First, computing the likelihood ratio $R_{t|\hat{h}}^{v_t,o_t}$ defined in (5) depending on the outcome $o_t$ of the investigation of alerts $v_t$ observed in the past involves recomputing the likelihoods up to time $t$ using the forward algorithm. As a consequence, computing the expected likelihood ratio $E[R_{t|\hat{h}}^{v_t}]$ over all possible outcomes is computationally intensive. Second, solving (17) requires the computation of the expected change of the likelihood ratio due to the outcome of the investigation of *sets* *of alerts*, hence (17) is a combinatorial optimization problem. To overcome these computational challenges, in what follows we propose a low complexity approximation.

For ease of exposition, let us first consider the expected likelihood ratio after the investigation of a single alert $v_t = (t_1, j_1)$ chosen at time $t$,

$$\mathbb{E}[R^{v_t}_{t|\hat{h}}] = \mathbb{E}\left[\frac{p_{\hat{h}}(Y_{1:t} = y_{1:t}|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})}{p_1(Y_{1:t} = y_{1:t}|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})}\right].$$

The approximation we introduce is based on the assumption that when investigating an alert observed at time $t_1$, the expected change of the likelihood ratio up to time $t$ is close to that at time $t_1$, i.e., when the alert was observed, formally

$$\Delta\overline{R}^{t_1,j_1}_{t|\hat{h}} = \mathbb{E}[R^{v_t}_{t|\hat{h}}] - R_{t|\hat{h}} \approx \mathbb{E}[R^{v_t}_{t_1|\hat{h}}] - R_{t_1|\hat{h}} = \Delta\overline{R}^{t_1,j_1}_{t_1|\hat{h}}. \quad (18)$$

The expected likelihood ratio of observations up to time $t_1$ after investigating alert $(t_1, j_1)$ can be expressed as

$$\begin{aligned}\mathbb{E}[R^{v_t}_{t_1|\hat{h}}] &= \mathbb{E}\left[\frac{p_{\hat{h}}(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})}{p_1(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, \mathcal{V}_t = \{(t_1, j_1)\})}\right] \\ &= \sum_{o^{t_1,j_1}_t \in\{0,1\}} \left( \frac{\begin{array}{r} p_{\hat{h}}(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, \\ \mathcal{V}_t = \{(t_1, j_1)\}) \end{array}}{\begin{array}{r} p_1(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, \\ \mathcal{V}_t = \{(t_1, j_1)\}) \end{array}} \right. \\ &\qquad\qquad \left. \cdot\, \mathbb{P}(O^{t_1,j_1}_t = o^{t_1,j_1}_t|\mathcal{F}_t = f_t) \right).\end{aligned}$$

The likelihood ratio, conditioned on the investigation outcome, can itself be computed as

$$\begin{aligned}&\frac{p_{\hat{h}}(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, \mathcal{V}_t = \{(t_1, j_1)\})}{p_1(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, \mathcal{V}_t = \{(t_1, j_1)\})} \\ &\qquad = \frac{\sum_{i\in\mathcal{I}_{\hat{h}}} p^{t,t_1}_{i|\hat{h}} \cdot c^{t_1,j_1,i}_t}{p^{t,t_1}_{1|1} \cdot c^{t_1,j_1,1}_t},\end{aligned}$$

where $c^{t_1,j_1,i}_t$ is the factor of likelihood change due to the outcome of the alert investigation. The likelihood change factor itself can be expressed as

$$c^{t_1,j_1,i}_t = \frac{1 - (1 - \zeta^{t_1,j_1}_t)(1 - \delta^{t_1,j_1,i}_t)}{1 - (1 - \zeta^{t_1,j_1}_{t-1})(1 - \delta^{t_1,j_1,i}_{t-1})},$$

where $\zeta^{t_1,j_1}_t$ and $\delta^{t_1,j_1,i}_t$ are given by (1) and (3), respectively. The numerator and the denominator are the likelihoods after the investigation and before the investigation, respectively. The denominator cancels out the likelihood before the investigation in $p^{t,t_1}_{i|h}$.

Let us now extend our focus to multiple alerts observed at the same time. The following result shows that computing the likelihood ratio change due to alerts that were observed at the same time can be done efficiently.

**Proposition 4.** *Let $v_t = \{(t_1, j_1), (t_1, j_2)\}$. Then*

$$\begin{aligned}&\mathbb{E}\left[\frac{p_{\hat{h}}(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, \mathcal{V}_t = v_t)}{p_1(Y_{1:t_1} = y_{1:t_1}, |\mathcal{F}_t = f_t, \mathcal{V}_t = v_t)}\right] \\ &\qquad = \sum_{i\in\mathcal{I}_{\hat{h}}} \left\{ \frac{p^{t,t_1}_{i|\hat{h}}}{p^{t,t_1}_{1|1}} \mathbb{E}\left[\frac{c^{t_1,j_1,i}_t}{c^{t_1,j_1,1}_t}\right] \mathbb{E}\left[\frac{c^{t_1,j_2,i}_t}{c^{t_1,j_2,1}_t}\right] \right\},\end{aligned}$$

*where the first expectation is taken with respect to $\mathbb{P}(O^{t_1,j_1}t = o^{t_1,j_1}_t|\mathcal{F}_t = f_t)$, and the second expectation is taken with respect to $\mathbb{P}(O^{t_1,j_2}t = o^{t_1,j_2}_t|\mathcal{F}_t = f_t)$.*

*Proof.* Let us express the likelihood ratio conditional on the investigation outcomes,

$$\begin{aligned}&\frac{\begin{array}{r} p_{\hat{h}}(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, O^{t_1,j_2}_t = o^{t_1,j_2}_t, \\ \mathcal{V}_t = v_t) \end{array}}{\begin{array}{r} p_1(Y_{1:t_1} = y_{1:t_1}|\mathcal{F}_t = f_t, O^{t_1,j_1}_t = o^{t_1,j_1}_t, O^{t_1,j_2}_t = o^{t_1,j_2}_t, \\ \mathcal{V}_t = v_t) \end{array}} \\ &\qquad = \frac{\sum_{i\in\mathcal{I}_{\hat{h}}} p^{t,t_1}_{i|\hat{h}} \cdot c^{t_1,j_1,i}_t \cdot c^{t_1,j_2,i}_t}{p^{t,t_1}_{1|1} \cdot c^{t_1,j_1,1}_t \cdot c^{t_1,j_2,1}_t}.\end{aligned}$$

Recall that by Proposition 2 the outcomes of the investigations of alerts $(t_1, j_1)$ and $(t_1, j_2)$ are independent, and hence

$$\mathbb{E}\left[\frac{c^{t_1,j_1,i}_t c^{t_1,j_2,i}_t}{c^{t_1,j_1,1}_t c^{t_1,j_2,1}_t}\right] = \mathbb{E}\left[\frac{c^{t_1,j_1,i}_t}{c^{t_1,j_1,1}_t}\right] \mathbb{E}\left[\frac{c^{t_1,j_2,i}_t}{c^{t_1,j_2,1}_t}\right],$$

where the first, second and third expectation is taken with respect to $\mathbb{P}(O^{t_1,j_1}_t = \{o^{t_1,j_1}_t, o^{t_1,j_2}_t\}|\mathcal{F}_t = f_t)$, $\mathbb{P}(O^{t_1,j_1}_t = o^{t_1,j_1}_t|\mathcal{F}_t = f_t)$ and $\mathbb{P}(O^{t_1,j_2}_t = o^{t_1,j_2}_t|\mathcal{F}_t = f_t)$, respectively. □

It is easy to see that the above result holds for any number of alerts observed at the same time. It does, however, not hold in general for $\mathcal{V}_t = \{(t_1, j_1), (t_2, j_2)\}$ if $t_2 \neq t_1$. Nonetheless, for computational efficiency, we propose using (18) to approximate the expected change in the probability ratio when an alert observed at time slot $t_1$ is investigated at time $t$. This approximation assumes that the expected change in the probability ratio is the same, regardless of whether the alert is investigated immediately upon observation or at a later time.

The MR policy is based on the above, and its pseudocode is shown in Algorithm 2. At time $t$ the policy takes $\Delta\overline{R}^{t_1,j_1}_{t_1|\hat{h}}$ as input, i.e., the expected change of the probability ratio for every past positive alert $\mathcal{V}_t = \{(t_1, j_1)\}$ (Line 5). It sorts the alerts in descending order of $\Delta\overline{R}^{t_1,j_1}_{t_1|\hat{h}}$ (Line 10) and picks the $B$ alerts with highest expected probability ratio changes and the $B$ alerts with lowest probability ratio changes (Line 11). It then chooses the set with the largest absolute sum (Line 12 to 16), which is then the set of alerts to be investigated by the security analysts at time $t$.

**Algorithm 2:** MR policy

```
Input: Past observations $f_t = (y_{1:t}, o_{1:t-1}, v_{1:t-1})$;
       Investigation budget $B$
Output: $v_t$ (Alerts to investigate)
$n_A = 0$; $\vec{v} = \emptyset$
for $t_1 \leftarrow 1$ to $t$ do
    for $j_1 \leftarrow 1$ to $J$ do
        if $Y_{t_1}^{j_1} = 1$ then
            Compute $\Delta\overline{R}_{t_1|\hat{h}}^{t_1,j_1}$
            $\vec{v}$.insert($(t_1, j_1)$)
            $n_A++$
    end
end
Sort $\vec{v}$ in descending order of $\Delta\overline{R}_{t_1|\hat{h}}^{t_1,j_1}$
$v_F = \{\vec{v}(1), \ldots, \vec{v}(B)\}$
$v_L = \{\vec{v}(n_A - B + 1), \ldots, \vec{v}(n_A)\}$
$\sigma_F = \sum_{(t_1,j_1) \in v_F} \Delta\overline{R}_{t_1|\hat{h}}^{t_1,j_1}$,
$\sigma_L = \sum_{(t_1,j_1) \in v_L} \Delta\overline{R}_{t_1|\hat{h}}^{t_1,j_1}$.
if $|\sigma_F| \geq |\sigma_L|$ then
    $v_t = v_F$
else
    $v_t = v_L$
end
```

### D. Discussion

While the MKL and the MR policies are similar in that they maximize the ability of the MSGPRT to make a decision by greedily choosing alerts to be investigated, they are conceptually different in how they try to achieve this. The MKL policy bases its decisions on the dissimilarity of the distribution of the alerts that would be observed after the alert investigation, i.e., it does not aim at maximizing the likelihood of the already made observations. On the contrary, the MR policy is directly concerned with how well the model is aligned with the already made observations, quantified by the likelihood of the observations, and tries to find alerts to investigate that would allow the model to better explain the existing observations. In addition, it is worth noting that the computational complexity of the MR policy is lower than that of the MKL policy due to the multiplication and $\log$ operations in (9).

## V. Numerical Results

In this section, we evaluate the proposed policies using simulations on a realistic attack graph.

### A. Attack graph and attack scenarios

Our framework assumes that the defender has a predefined attack graph that describes the attacker's behavior. This assumption implies that the performance of our model could degrade if the attacker adopts a dynamic strategy or if vulnerabilities in the network change.

We evaluate the proposed policies on a DoS attack graph, shown in Fig. 3. DoS attacks aim at making network services unavailable to legitimate users. This is achieved by overwhelming the target with excessive traffic or requests, exhausting the resources of the target system. DoS attacks are relatively common and may have significant impact in financial institutes [30], power systems [31], and e-government infrastructures [32]; thus their timely detection is important. We use the DoS attack graph based on the LLDDOS1.0 attack in [19], which consists of five attack steps: 1) IPsweep, 2) sadmind ping, 3) break into, 4) installation, and 5) launch, of which only three steps (i.e., sadmind ping, break into, and launch) raise alerts by the IDS. We thus obtain an attack graph with three states, where states 2 to 4 in the attack graph correspond to the following attack steps: sadmind ping, break into, and launch, respectively. Each state represents the use of potentially multiple exploits. We then add a safe state ($s_1$) and a transition from the safe state $s_1$ to the intrusion attempt state $s_2$, resulting in an attack graph with $|\mathcal{S}| = 4$ states. Table II shows the transition probabilities $a_{s_i, s_{i'}}$ for $i, i' \in \mathcal{I}$, and Table III shows the probability of true and false alerts, adopted from [19]. We refer to the resulting HMM as the *non-stealthy* attack scenario. As a second scenario, we consider a modification of the above HMM, where the true alert probability for Alert 3 is 10 times smaller, making attack states 2 and 3 more difficult to observe. We refer to this modified HMM as the *stealthy* attacker scenario. Thus, we use two scenarios for the evaluation: the *non-stealthy* attacker and the *stealthy* attacker.

In our simulation, each time slot represents 20 minutes because security analysts may need up to 20 minutes to investigate an alert, based on [33]. In practice, alert investigations typically take only a few minutes [34], suggesting that the time slot is sufficient for investigating a single alert. Additionally, we note that each state in our attack tree abstracts multiple exploits, which increases the probability of remaining in a given state. Thus, generating an alert vector every 20 minutes is sufficient to track the attacker's progression.

### B. Evaluation methodology

The initial belief of the defender is $\pi_1^{1,1} = \mathbb{P}(S_0 = s_1) = 1$, and we use $\gamma_0 = 2$ for the alert probability adjustment factor.

TABLE II
Transition probabilities of the HMM from [19].

| **Current** | **Next state ($i'$)** | | | |
|---|---|---|---|---|
| **State ($i$)** | **1** | **2** | **3** | **4** |
| **1** | 0.95 | 0.05 | 0 | 0 |
| **2** | 0 | 0.974 | 0.013 | 0.013 |
| **3** | 0 | 0.010 | 0.962 | 0.028 |
| **4** | 0 | 0.011 | 0.011 | 0.978 |

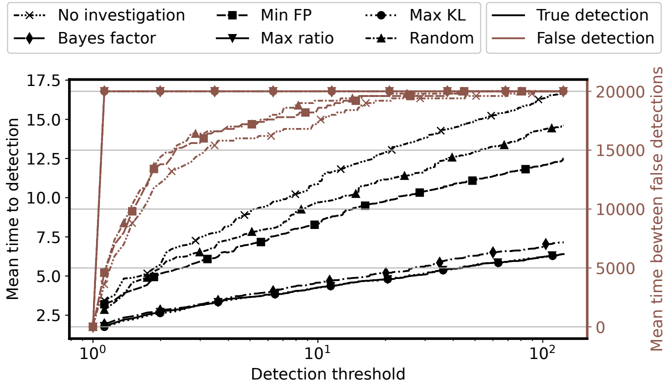

Fig. 6. Mean detection delay and mean time between false detections as a function of the detection threshold for the linear confidence function, $\omega = 0$, the *non-stealthy* attack scenario, and investigation budget $B = 1$. (Each time step is equivalent to 20 minutes.)

We consider investigation error probabilities $\omega \in [0, 0.5]$ for the evaluation.

We use four baseline algorithms for the evaluation. The first baseline does not perform investigation, referred to as *No investigation*. The second baseline investigates $B$ alerts chosen uniform at random, referred to as *Random*. The third baseline chooses the $B$ alerts with smallest false alert probabilities among the positive alerts, referred to as *Min FP*. The fourth baseline is a dynamic investigation policy called Bayes factor policy, proposed in [2], which prioritizes the most ambiguous alerts observed at time $t$, referred to as *Bayes factor*.

We compute the detection latency in each simulation and for each policy as the difference between the time of confirming the attacker's intrusion $d^{\kappa}$ and the time the intrusion started $t_{s_1 \to s_2}$. The presented results are based on 100 simulations of $20,000$ time steps each; we refer to the mean of the detection latency as the *mean time to detection*. Besides the time to detection, we use the mean squared error (MSE) of the belief $\pi_i^{t,t}$ to evaluate the learning process, defined as

$$MSE(\pi_i^{t,t}, S_t) = \sum_{i \in \mathcal{I}} \left( \mathbf{1}_{\{s_i = S_t\}} - \pi_i^{t,t} \right)^2,$$

where $\mathbf{1}_{\{s_i = S_t\}}$ is the indicator function.

We compute the true positive rate (TPR) and false positive rate (FPR) at time $t = 20$ to demonstrate the improved accuracy of intrusion detection using a human-in-the-loop approach [35]. The TPR and FPR represent the proportion of correct and incorrect intrusion detections identified by our framework, respectively. These metrics are calculated as follows:

$$TPR = \frac{TP}{TP + FN},$$
$$FPR = \frac{FP}{FP + TN},$$

where $TP$ is the number of true positive detections, $FN$ is the number of false negative detections, $FP$ is the number of false positive detections, and $TN$ is the number of true negative detections.

### C. Mean time to detection

Fig. 6 shows the mean time to (true) detection (left axis) and the mean time between false detections (MTBFD) (right axis) as a function of the detection threshold $\theta_{\hat{h}}$ (log.scale) for the *non-stealthy* scenario under the linear model. The figure shows that any form of alert investigation reduces the time to detection compared to *No investigation* even for an investigation budget as low as $B = 1$. More importantly, the two proposed policies achieve significantly lower time to detection and significantly higher MTBFD for all values of the detection threshold than the baselines. The proposed policies and *Bayes factor* perform almost equally well for an investigation budget of $B = 1$, and provide a high MTBFD for very low values of the detection threshold. This shows that they can effectively mitigate the effect of false alerts through alert investigation. We note that the MTBFD levels out at $20,000$ due to the length of the simulations. Comparing the baselines, we observe that *MinFP* performs consistently better than *Random*, although both perform poorly compared to the proposed policies. The mean time to detection increases approximately logarithmically with the detection threshold (note the log. scale on the horizontal axis) for all policies. In what follows, we set the detection threshold to achieve an MTBFD $\tau$ that is 90% of the number of time steps in the simulations, i.e., $\tau = 18,000$ time steps.

Fig. 7 shows the mean time to (true) detection as a function of the investigation error $\omega$ for the *non-stealthy* scenario under linear, concave, and convex confidence functions (defined in (2)). The figure shows that the proposed policies and *Bayes factor* reduce the mean time to detection compared to the *Random* and the *Min FP* policies by up to a factor of four, except when the investigation error is close to 0.5, i.e., when investigations are uninformative. We note that *Bayes factor* significantly reduces the mean time to detection, although the design of *Bayes factor* aims at the improvement of belief [2]. This can be explained by the fact that an improved belief increases the probability of hypothesis $h \geq 2$, leading to a reduction in the mean time to detection. Comparing the results obtained using different confidence functions, we note that the convex confidence function performs worst, hence we do not show results for this confidence function in the

TABLE III
TRUE AND FALSE ALERT RATES USED IN THE EVALUATION [19].

| | | | | | True alert rate ($\delta_{ij}$) | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Attack** | | | | | **Alert index ($j$)** | | | | |
| **state ($i$)** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **2** | 0 | 0 | 0.962 | 0 | 0.038 | 0 | 0 | 0 | 0 |
| **3** | 0 | 0 | 0.667 | 0 | 0 | 0.333 | 0 | 0 | 0 |
| **4** | 0.011 | 0.064 | 0 | 0.372 | 0 | 0.021 | 0.021 | 0.489 | 0.021 |
| | | | | | **False alert rate ($\zeta_j$)** | | | | |
| | 0.03 | 0.1 | 0.53 | 0.43 | 0.13 | 0.43 | 0.06 | 0.48 | 0.06 |

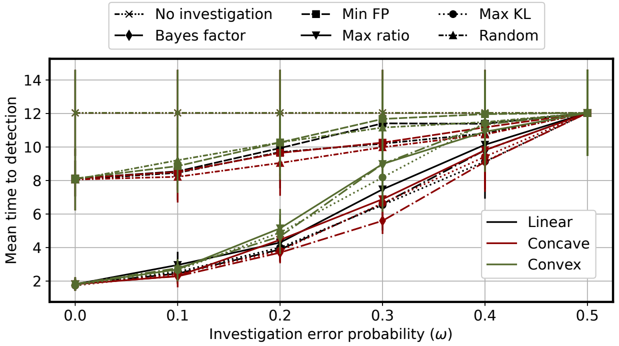

Fig. 7. Mean time to detection with 95% confidence intervals vs. investigation error probability ($\omega$) for the *non-stealthy* attack scenario, mean time between false dectections $\tau = 18,000$ and investigation budget $B = 1$. (One time step is equivalent to 20 minutes.)

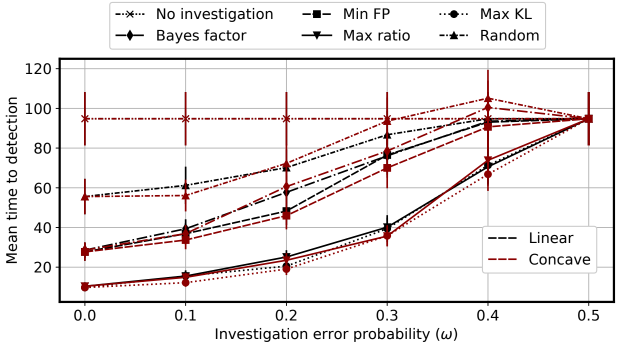

Fig. 8. Mean time to detection with 95% confidence intervals vs. investigation error probability ($\omega$) for the *stealthy* attack scenario, mean time between false dectections $\tau = 18,000$ and investigation budget $B = 1$. (Each time step is equivalent to 20 minutes.)

subsequent figures. In the considered *non-stealthy* scenario, the results for the linear and for the concave confidence functions are similar, and the figure shows that the time to detection increases approximately linearly with the investigation error probability. That is, the proposed policies and *Bayes factor* exhibit graceful degradation under error-prone investigations, which is a desirable property in general.

Fig. 8 shows corresponding results for the *stealthy* attacker scenario under the linear and the concave confidence functions. Comparing Fig. 8 to Fig. 7, we can observe that the mean time to detection is almost an order of magnitude higher for all policies due to the reduction in the true positive rate of alert 3, confirming that the scenario is more difficult from the defender's perspective. Despite the difficulty of the scenario, the proposed policies outperform the baselines by up to a factor of three to five. Focusing on the baselines, the figure shows that while the *Random* and *Bayes factor* perform relatively poorly, the *Min FP* policy performs relatively close to the proposed policies, especially for low investigation error probability $\omega$. On the one hand, the big improvement in the relative performance of *Min FP* compared to the *non-stealthy* scenario is due to the modification of alert 3, which has a high false positive rate and high true positive rate, and hence those alerts rarely get investigated under the *Min FP* policy, even though they are very informative in certain attack states. On the other hand, the degraded performance of *Bayes factor* compared to the *non-stealthy* scenario is because the stealthiness and the investigation errors significantly degrade the accuracy of the belief for state $s_2$ despite using *Bayes factor*, and the inaccurate belief in turn increases the time to detection. This result shows that the proposed policies are more robust to changes in the correlation structure of false and true alerts compared to static policies.

Fig. 8 also shows that for the *Min FP* and for the proposed policies, it is the concave confidence function that performs best, slightly better than the linear confidence function. Surprisingly, however, using the concave confidence function together with the *Random* and *Bayes factor* policies results in a longer mean time to detection than *No investigation* when the error probability $\omega$ is high. We attribute this to the fact that the *Random* and *Bayes factor* policies mostly choose false alerts to investigate in the *stealthy* scenario (since there are many more of those), and due to the high investigation error $\omega = 0.4$ the false alerts are often confirmed as true alerts. These mistakenly confirmed alerts, in turn, increase the likelihood ratio in (5), which leads to false detections. Hence, the detection threshold has to be significantly increased to maintain $\tau = 18,000$, which leads to an increase in the mean time to detection. To verify this hypothesis, we simulated a modified *Random* policy, which reinvestigates an alert two times when the investigation outcome is true. This modification reduced the mean time to detection to that of *No investigation*, confirming our hypothesis. Motivated by its superior performance when using the proposed policies, in what follows we show results for the concave confidence function only, and we use the *Min FP* policy for comparison, which is best among the baselines.

Table V summarizes the reduction in detection time with respect to the best baseline policy, i.e., *Min FP*,

$$\mathrm{RDT}^{\kappa} = \frac{\begin{array}{r}\mathbb{E}^{(t_{s_1\to s_2})}[d^{\kappa^{MinFP}} - t_{s_1\to s_2}] \\ - \mathbb{E}^{(t_{s_1\to s_2})}[d^{\kappa} - t_{s_1\to s_2}]\end{array}}{\mathbb{E}^{(t_{s_1\to s_2})}[d^{\kappa^{MinFP}} - t_{s_1\to s_2}]}.$$

Comparing the proposed policies *Max KL* and *Max Ratio*, we can conclude that there is no significant difference for $B = 1$.

To further analyze the policies, Fig. 9 shows the MSE of the belief as a function of time for the *stealthy* attacker scenario. The figure shows that all investigation policies reduce the MSE compared to *No investigation*, confirming that alert investigations improve the belief. Importantly, the figure shows that *Random*, *MinFP* and *Bayes factor* results in similar MSE values, indicating *Bayes factor* cannot improve the MSE if the attacker is stealthy. This supports our explanation for the results shown in Fig. 8, i.e., that the *Bayes factor* policy cannot detect

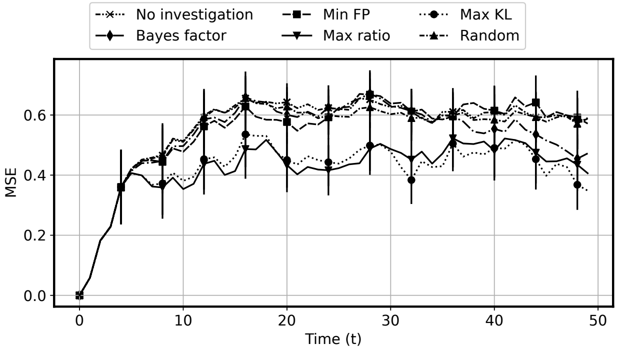

Fig. 9. Learning process for the concave confidence function, $\omega = 0$ , the *stealthy* attack scenario, and investigation budget $B = 1$. (Each time step is equivalent to 20 minutes.)

the attacker's intrusion due to the degradation in the accuracy of the belief. On the contrary, our policies maintain a low MSE and hence they can choose the most informative alerts for timely detection.

Finally, Table IV shows the impact of our proposed policies on the TPR and the FPR at time $t = 20$ under the *stealthy* attacker scenario. Overall, we observe that all policies improve the TPR compared to *No investigation*, and our proposed policies show the highest TPR, confirming their superior performance. Recall that our proposed policies set lower detection thresholds compared to the baseline policies, which implies that our proposed policies may have a high FPRs. This is, however, not the case. Comparing the proposed policies, *Max KL* has a lower FPRs than *Max ratio*, while it has a similar mean time to detection (c.f., Fig. 7 and Fig. 8). We attribute this to the fact that *Max ratio* directly maximizes the likelihood ratio, while *Max KL* aims to maximize distinguishability. Hence, *Max ratio* may result in a quick change in the likelihood ratio, making it more vulnerable to investigation errors. Our results thus indicate that the *Max KL* policy minimizes two objectives: the mean time to detection and the FPR.

TABLE IV
TPR AND FPR UP TO TIME STEP $t = 20$ WITH THE CONCAVE CONFIDENCE FUNCTION. THE BOLDFACE MEANS THE MAXIMUM TPR AND FPR FOR EACH INVESTIGATION ERROR $\omega$ AND ATTACK SCENARIO.

| **Attacker** | **Investigation** | **Investigation error probability ($\omega$)** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Type** | **policy** | **0** | | **0.1** | | **0.2** | | **0.3** | | **0.4** | |
| | **($\kappa$)** | **TPR** | **FPR** | **TPR** | **FPR** | **TPR** | **FPR** | **TPR** | **FPR** | **TPR** | **FPR** |
| **Non Stealthy** | **Max KL** | **1.00** | 0.00 | **0.98** | 0.15 | **0.94** | 0.06 | **0.88** | 0.09 | 0.70 | 0.06 |
| | **Max ratio** | **1.00** | 0.00 | **0.98** | **0.21** | **0.94** | 0.09 | 0.85 | 0.03 | **0.77** | **0.12** |
| | **Min FP** | 0.73 | **0.12** | 0.71 | 0.12 | 0.70 | **0.12** | 0.71 | **0.12** | 0.67 | **0.12** |
| | **Random** | 0.77 | 0.09 | 0.77 | 0.09 | 0.76 | 0.06 | 0.76 | 0.06 | 0.68 | 0.09 |
| | **No investigation** | 0.64 | 0.09 | 0.64 | 0.09 | 0.64 | 0.09 | 0.64 | 0.09 | 0.64 | 0.09 |
| **Stealthy** | **Max KL** | **0.70** | 0.00 | **0.64** | 0.00 | **0.42** | 0.00 | **0.17** | 0.00 | 0.00 | 0.00 |
| | **Max ratio** | 0.68 | 0.00 | 0.53 | **0.03** | **0.42** | 0.00 | 0.14 | 0.00 | **0.02** | 0.00 |
| | **Min FP** | 0.27 | 0.00 | 0.18 | 0.00 | 0.15 | 0.00 | 0.02 | 0.00 | 0.00 | 0.00 |
| | **Random** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| | **No investigation** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

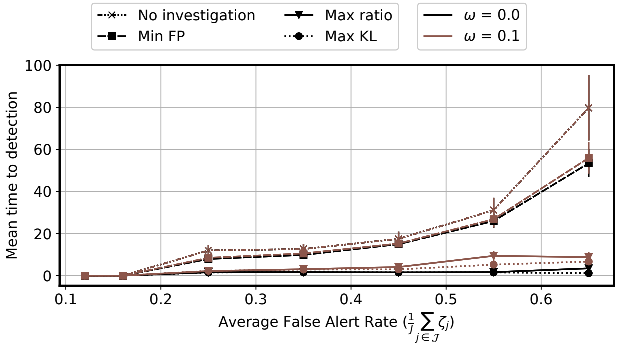

Fig. 10. Mean time to detection with 95% confidence intervals vs. average false positive rate for the *non-stealthy* attack scenario under mean time between false dectections $\tau = 18,000$ and investigation budget $B = 1$. (Each time step is equivalent to 20 minutes.)

## D. Impact of false alert rates

In this subsection, we explore the impact of the false alert rate on the mean time to detection by considering different values for the average false positive rate. Fig. 10 shows the mean time to detection as a function of the average false alert rate for the *non-stealthy* scenario. To adjust the average false alert rate, which is 0.25 in Table III, increase or decrease the probability $\zeta_j$ of each false alert by the same value, as long as $\zeta_j \in [0, 1]$. The figure shows that the mean time to detection increases significantly using the *No investigation* and the *Min FP* policies as the false positive rate increases. On the contrary, our proposed policies are almost insensitive to the increase in the false positive rate. We also notice that *Max KL* shows slightly better performance than *Max ratio* when the false positive rate is very high, indicating that the policy is superior to the *Max ratio* policy.

## E. Impact of investigation budget

In this subsection, we explore the impact of the investigation budget $B$ on the mean time to detection. Fig. 11 shows the mean time to detection as a function of the investigation budget $B$ under the *stealthy* scenario using the concave confidence function. We observe that the proposed policies make much more efficient use of the investigation budget. As an example, an investigation budget of $B = 1$ with the proposed policies achieves lower or equivalent time to detection than the baseline *Min FP* policy for $B = 4$. We can also observe that increasing the investigation budget has a decreasing marginal gain for

TABLE V
REDUCTION IN DETECTION TIME BY OUR POLICIES UNDER THE CONCAVE MODEL COMPARED TO MIN FP POLICY.

| **Policy ($\kappa$)** | **Simulated difficulty** | **Invesetigation error probability ($\omega$)** 0 | **0.1** | **0.2** | **0.3** | **0.4** | **Average reduction** |
|---|---|---|---|---|---|---|---|
| **Max ratio** | **Non-stealthy** | 77.69% | 72.98% | 53.76% | 32.89% | 12.19% | 49.90% |
| | **Stealthy** | 62.52% | 55.77% | 48.90% | 49.07% | 18.55% | 46.96% |
| **Max KL** | **Non-stealthy** | 78.62% | 71.51% | 60.49% | 35.20% | 15.66% | 52.30% |
| | **Stealthy** | 64.79% | 63.78% | 58.53% | 48.86% | 26.33% | 52.46% |

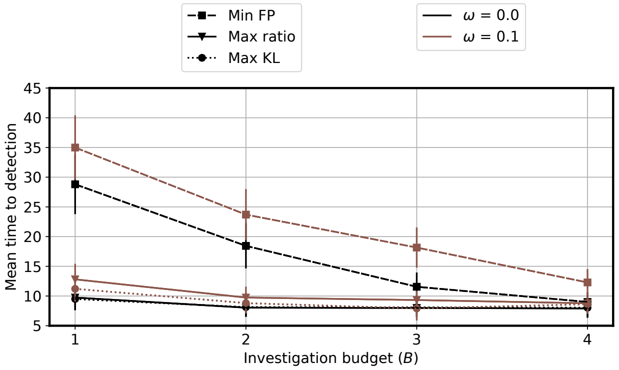

Fig. 11. Mean time to detection with 95% confidence intervals vs. investigation budget ($1 \leq B \leq 4$) for the *stealthy* attack scenario under mean time between false dectections $\tau = 18,000$. (Each time step is equivalent to 20 minutes.)

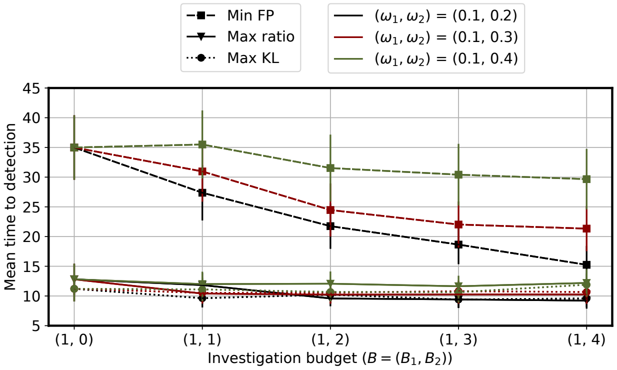

Fig. 12. Mean time to detection with 95% confidence intervals vs. investigation budget ($B_1 = 1$ with $\omega_1 = 0.1$ and $1 \leq B_2 \leq 4$ with $\omega_2 \in \{0.2, 0.3, 0.4\}$) for the *stealthy* attack scenario for a mean time between false dectections of $\tau = 18,000$. (Each time step is equivalent to 20 minutes.)

all policies, implying that adding more security analysts has negligible impact on the time to detection beyond a certain point. The figure also confirms that the *Max KL* policy slightly outperforms the *Max Ratio* policy at the expense of a slightly higher computational overhead imposed by the computation of the KL divergence.

### F. Impact of diverse investigation errors

In this subsection, we explore the impact of the collaboration of security analysts with different investigation error probabilities on the mean time to detection. For the evaluation, we consider that there are two security analysts, one with investigation error probability $\omega_1 = 0.1$ and budget $B_1 = 1$ and one with investigation error probability $\omega_2 \in \{0.2, 0.3, 0.4\}$ and budget $B_2 \in \{0, \ldots, 4\}$. Fig. 12 shows the mean time to detection as a function of the investigation budget of security analyst 2 for the *stealthy* scenario using the concave confidence function. The figure shows that the security analyst with high investigation error probability improves the mean time to detection if using the proposed policies. Comparing our proposed policies, the figure shows that *Max KL* outperforms *Max ratio* consistently. On the contrary, adding a security analyst when using the *Min FP* policy may be detrimental to the time to detection unless its investigation budget $B_2$ is high. This is due to the fact that adding an analyst with a high investigation error probability may mislead attack detection. Comparing Figures 11 and 12, we observe that having an inexperienced analyst ($\omega_2 = 0.3$) with a budget of $B_2 = 4$ reduces the mean time to detection approximately as much as adding a single experienced analyst with $\omega = 0.1$, which highlights the importance of the skills of the security analysts employed.

We next evaluate the case of having two security analysts with total investigation error probability $\omega_s = \omega_1 + \omega_2$. Fig. 13 shows the mean time to detection as a function of the investigation error probability $\omega_1$ of the first security analyst, for $\omega_s \in \{0.4, 0.5\}$. Interestingly, the figure shows that the mean time to detection increases with the investigation error probability of the most skilled analyst $\min(\omega_1, \omega_2)$.We can also observe that the mean time to detection of our proposed policies is the same for $\omega_1 = 0$ regardless of $\omega_s$, further highlighting the importance of the most skilled security analyst $\omega_1$. The figure also shows that *Max ratio* exhibits a higher mean time to detection compared to *Max KL*, again confirming the superior performance of *Max KL.*

## VI. Conclusion

We considered the problem of timely attack detection using dynamic alert prioritization, formulated as an active learning problem for sequential detection in a hidden Markov model. Besides considering active learning and quickest detection in a single framework, a key novelty of the proposed model is the adjustment of the observation probabilities in response to alert investigations. We introduced two alert prioritization policies that aim to minimize the time detection under a false detection time target. Our numerical results indicate that the proposed policies significantly reduce detection delay compared to baseline policies, effectively ignoring false alerts. The proposed framework has several interesting extensions. First, one could consider multiple attack graphs, each corresponding to a different attack scenario, in which case analysts have to be allocated to alerts from different attack graphs. Second, one could take into account the time needed for investigating different types of alerts and its impact on the investigation error probability. Third, one could consider a strategic attacker that tries to deceive the alert prioritization schemes, leading to a game theoretic model. Last but not least, our framework could be extended to include automated response actions, e.g., for temporarily isolating suspicious hosts, which would require joint consideration of the cost of response actions and time to detection.

## Acknowledgement

The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in

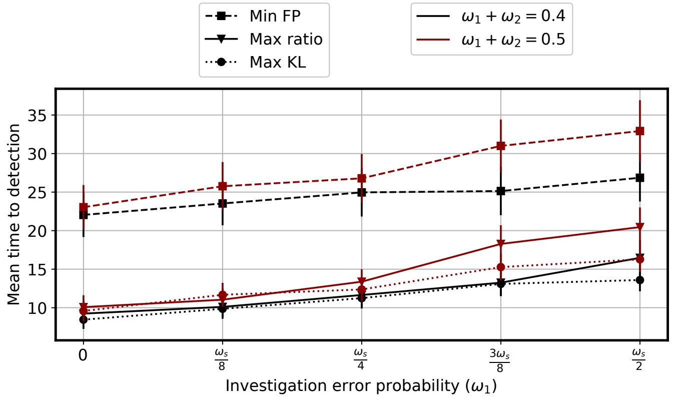

Fig. 13. Mean time to detection with 95% confidence intervals vs. total investigation error probability ($\omega_1 + \omega_2 \in \{0.4, 0.5\}$) under the *stealthy* attack scenario for a mean time between false dectections of $\tau = 18,000$. (Each time step is equivalent to 20 minutes.)

Sweden (NAISS) at Linköping University partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

## References

[1] A. Shah, R. Ganesan, S. Jajodia, and H. Cam, "Dynamic optimization of the level of operational effectiveness of a CSOC under adverse conditions," *ACM Trans. on Intelligent Syst. and Techn. (TIST)*, vol. 9, no. 5, pp. 1–20, 2018.

[2] Y. Kim and G. Dán, "An active learning approach to dynamic alert prioritization for real-time situational awareness," in *IEEE Conference on Communications and Network Security (CNS)*, 2022, pp. 154–162.

[3] A. Schlenker, H. Xu, M. Guirguis, C. Kiekintveld, A. Sinha, M. Tambe, S. Sonya, D. Balderas, and N. Dunstatter, "Don't bury your head in warnings: A game-theoretic approach for intelligent allocation of cyber-security alerts," in *Proc. of IJCAI*, 2017.

[4] N. Dunstatter, M. Guirguis, and A. Tahsini, "Allocating security analysts to cyber alerts using Markov games," in *Proc. of National Cyber Summit (NCS)*, 2018, pp. 16–23.

[5] A. Shah, R. Ganesan, S. Jajodia, and H. Cam, "Understanding tradeoffs between throughput, quality, and cost of alert analysis in a CSOC," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1155–1170, 2018.

[6] ——, "A two-step approach to optimal selection of alerts for investigation in a CSOC," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1857–1870, 2018.

[7] R. Ganesan and A. Shah, "A strategy for effective alert analysis at a cyber security operations center," *From Database to Cyber Security: Essays Dedicated to Sushil Jajodia on the Occasion of His 70th Birthday*, pp. 206–226, 2018.

[8] Y. Chen, J. Hong, and C.-C. Liu, "Modeling of intrusion and defense for assessment of cyber security at power substations," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 2541–2552, 2016.

[9] P. Xie, J. H. Li, X. Ou, P. Liu, and R. Levy, "Using Bayesian networks for cyber security analysis," in *2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN)*. IEEE, 2010, pp. 211–220.

[10] S. Nitinawarat and V. V. Veeravalli, "Controlled sensing for sequential multihypothesis testing with controlled Markovian observations and non-uniform control cost," *Sequential Analysis*, vol. 34, no. 1, pp. 1–24, 2015.

[11] F. Harrou, B. Bouyeddou, Y. Sun, and B. Kadri, "A method to detect DoS and DDoS attacks based on generalized likelihood ratio test," in *Intl. Conf. on Applied Smart Systems (ICASS)*, 2018, pp. 1–6.

[12] S. Li, Y. Yilmaz, and X. Wang, "Sequential cyber-attack detection in the large-scale smart grid system," in *IEEE Intl. Conf. on Smart Grid Communications (SmartGridComm)*, 2015, pp. 127–132.

[13] J. Grana, D. Wolpert, J. Neil, D. Xie, T. Bhattacharya, and R. Bent, "A likelihood ratio anomaly detector for identifying within-perimeter computer network attacks," *Journal of Network and Computer Applications*, vol. 66, pp. 166–179, 2016.

[14] O. P. Kreidl and T. M. Frazier, "Feedback control applied to survivability: a host-based autonomic defense system," *IEEE Transactions on Reliability*, vol. 53, no. 1, pp. 148–166, 2004.

[15] S. Iannucci, Q. Chen, and S. Abdelwahed, "High-performance intrusion response planning on many-core architectures," in *Intl. Conf. on Computer Communication and Networks (ICCCN)*, 2016, pp. 1–6.

[16] S. Iannucci and S. Abdelwahed, "A probabilistic approach to autonomic security management," in *Proc. of IEEE International Conference on Autonomic Computing (ICAC)*, 2016, pp. 157–166.

[17] E. Miehling, M. Rasouli, and D. Teneketzis, "Optimal defense policies for partially observable spreading processes on Bayesian attack graphs," in *Proc. of ACM Workshop on Moving Target Defense*, 2015, pp. 67–76.

[18] ——, "A POMDP approach to the dynamic defense of large-scale cyber networks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2490–2505, 2018.

[19] P. Holgado, V. A. Villagrá, and L. Vazquez, "Real-time multistep attack prediction based on hidden Markov models," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 1, pp. 134–147, 2017.

[20] M. Girdhar, J. Hong, H. Lee, and T.-J. Song, "Hidden Markov models-based anomaly correlations for the cyber-physical security of EV charging stations," *IEEE Transactions on Smart Grid (TSG)*, vol. 13, no. 5, pp. 3903–3914, 2021.

[21] Y. Javed, M. A. Khayat, A. A. Elghariani, and A. Ghafoor, "PRISM: a hierarchical intrusion detection architecture for large-scale cyber networks," *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 2023.

[22] H. Liu, R. Jiang, B. Zhou, X. Rong, J. Li, and A. Li, "Multiple sequential network attacks detection based on DTW-HMM," in *IEEE International Conference on Data Science in Cyberspace (DSC)*. IEEE, 2022, pp. 134–141.

[23] B. Anderson and M. Andrew, "Active learning for hidden Markov models: Objective functions and algorithms," in *Proc. of Intl. Conf. on Machine Learning (ICML)*, 2005.

[24] "MITRE ATT&CK®," accessed: 2023-05-29. [Online]. Available: https://attack.mitre.org/

[25] C.-D. Fuh, "SPRT and CUSUM in hidden Markov models," *The Annals of Statistics*, vol. 31, no. 3, pp. 942 – 977, 2003.

[26] C.-D. Fuh and Y. Mei, "Quickest change detection and Kullback-Leibler divergence for two-state hidden Markov models," in *IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 141–145.

[27] B. Ghojogh, F. Karray, and M. Crowley, "Hidden Markov Model: Tutorial," 2019.

[28] B. C. Levy, *Principles of signal detection and parameter estimation*. Springer, 2008.

[29] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[30] M. Vasek, M. Thornton, and T. Moore, "Empirical analysis of denial-of-service attacks in the bitcoin ecosystem," in *Financial Cryptography and Data Security: FC 2014 Workshops, BITCOIN and WAHC 2014, Christ Church, Barbados, March 7, 2014, Revised Selected Papers 18*. Springer, 2014, pp. 57–71.

[31] A. G. Wermann, M. C. Bortolozzo, E. G. da Silva, A. Schaeffer-Filho, L. P. Gaspary, and M. Barcellos, "ASTORIA: A framework for attack simulation and evaluation in smart grids," in *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2016, pp. 273–280.

[32] J. Nazario, "DDoS attack evolution," *Network Security*, no. 7, pp. 7–10, 2008.

[33] M. Sharif, P. Datta, A. Riddle, K. Westfall, A. Bates, V. Ganti, M. Lentz, and D. Ott, "Drsec: Flexible distributed representations for efficient endpoint security," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 2024, pp. 145–145.

[34] J. Ghadermazi, A. Shah, and S. Jajodia, "A machine learning and optimization framework for efficient alert management in a cybersecurity operations center," *Digital Threats: Research and Practice*, 2024.

[35] A. Tharwat, "Classification assessment methods," *Applied computing and informatics*, vol. 17, no. 1, pp. 168–192, 2020.

**Yeongwoo Kim** received his M.Sc. degree in Electrical Engineering from both KTH Royal Institute of Technology, Sweden, and the Technical University of Berlin, Germany, in 2020. He is currently pursuing a Ph.D. in the Division of Network and Systems Engineering at KTH Royal Institute of Technology in Stockholm, Sweden. His research interests icnludes human-in-the-loop systems, machine learning algorithms, and analytic approaches for cybersecurity.

**György Dán** (Senior Member, IEEE) received the M.Sc. degree in computer engineering from the Budapest University of Technology and Economics, Hungary, in 1999, the M.Sc. degree in business administration from the Corvinus University of Budapest, Hungary, in 2003, and the Ph.D. degree in telecommunications from KTH in 2006. He was a Consultant in the field of access networks, streaming media, and videoconferencing from 1999 to 2001. He was a Visiting Researcher at the Swedish Institute of Computer Science in 2008, a Fulbright Research Scholar at the University of Illinois at Urbana–Champaign from 2012 to 2013, and an Invited Professor at EPFL in from 2014 to 2015. He is a Professor with the KTH Royal Institute of Technology, Stockholm, Sweden. His research interests include the design and analysis of content management and computing systems, game theoretical models of networked systems, and cyber-physical system security and resilience. He was area editor of Computer Communications from 2014 to 2021 and of IEEE Trans. on Mobile Computing 2019-2023.

**Quanyan Zhu** Quanyan Zhu (Senior Member, IEEE) received the B.Eng. degree (Hons.) in electrical engineering from McGill University in 2006, the M.A.Sc. degree from the University of Toronto in 2008, and the Ph.D. degree from the University of Illinois at Urbana-Champaign (UIUC) in 2013. After stints at Princeton University, he is currently an Associate Professor with the Department of Electrical and Computer Engineering, New York University (NYU), where he is an Affiliated Faculty Member with the Center for Urban Science and Progress (CUSP). He is the coauthor of two recent books published by Springer: *Cyber-Security in Critical Infrastructures: A Game-Theoretic Approach* (with S. Rass, S. Schauer, and S. König) and *A Game- and Decision-Theoretic Approach to Resilient Interdependent Network Analysis and Design* (with J. Chen). His current research interests include game theory, machine learning, cyber deception, network optimization and control, smart cities, the Internet of Things, and cyber-physical systems. He was a recipient of many awards, including the NSF CAREER Award, the NYU Goddard Junior Faculty Fellowship, the NSERC Postdoctoral Fellowship (PDF), the NSERC Canada Graduate Scholarship (CGS), and the Mavis Future Faculty Fellowships. He spearheaded and chaired the INFOCOM Workshop on Communications and Control on Smart Energy Systems (CCSES), the Midwest Workshop on Control and Game Theory (WCGT), and the ICRA workshop on Security and Privacy of Robotics. He served as the General Chair or the TPC Chair for the Seventh and the 11th Conference on Decision and Game Theory for Security (GameSec) in 2016 and 2020, the Ninth International Conference on NETwork Games, COntrol and OPtimization (NETGCOOP) in 2018, the Fifth International Conference on Artificial Intelligence and Security (ICAIS 2019) in 2019, and the 2020 IEEE Workshop on Information Forensics and Security (WIFS). He also spearheaded the IEEE Control System Society (CSS) Technical Committee on Security, Privacy, and Resilience, in 2020.