Dynamic Information Flow Tracking for Detection of Advanced Persistent Threats: A Stochastic Game Approach

Shana Moothedath, *Member, IEEE,* Dinuka Sahabandu, Joey Allen, Andrew Clark, *Member, IEEE,* Linda Bushnell, *Fellow, IEEE,* Wenke Lee, *Fellow, IEEE* and Radha Poovendran, *Fellow, IEEE*

***Abstract*—Advanced Persistent Threats (APTs) are stealthy attacks by intelligent adversaries. This paper studies the detection of APTs that infiltrate cyber systems and compromise specifically targeted data and/or infrastructures. Dynamic information flow tracking is an information trace-based detection mechanism against APTs that tags suspicious information flows in the system and performs security analysis for unauthorized use of tagged data. In this paper, we develop an analytical model for resource-efficient detection of APTs using an information flow tracking game. The game is a nonzero-sum, turn-based, stochastic game with asymmetric information as the defender cannot distinguish whether an incoming flow is malicious or benign. The payoff functions of the game capture the cost for performing security analysis and the rewards and penalties received by the players. We analyze equilibrium of the game and prove that a Nash equilibrium is given by a solution to the minimum capacity cut set problem on a flow-network derived from the system. The edge capacities of the flow-network are obtained from the cost of performing security analysis. Finally, we implement our algorithm on a real-world dataset for a data exfiltration attack augmented with false-negative and false-positive rates and compute an optimal defender strategy.**

***Index Terms*—Advanced Persistent Threats (APTs), Information flow tracking, Stochastic games, Minimum-cut problem**

I. Introduction

An advanced persistent threat (APT) is a prolonged and targeted cyber attack in which an intruder gains illicit access to a system and remains undetected for an extended period of time. The intention of an APT is to monitor system activity and continuously mine highly sensitive data rather than causing damage to the system or organization. APT attacks consist of multiple stages that are initiated by an initial compromise and reconnaissance stage to establish a foothold in the system. Attackers then progress through the system, exploring and planning an attack strategy to obtain the desired data. This is followed by exfiltration of sensitive data, which is continued over a long period of time. Defending against APTs is a challenging task since APTs are specifically designed to evade conventional security mechanisms such as firewalls, anti-virus software and intrusion-detection systems that rely on signatures and can, therefore, guard only against known threats. However, APTs introduce information flows in the form of data-flow commands and control-flow commands while interacting with the system and these are continuously recorded in the log file of the system.

After an APT attack, when the consequences of the attack has been identified, a forensic investigation is conducted using the data from the log files. The purpose of this postmortem investigation is to understand the incident's root cause, and construct appropriate defense strategies against such APTs and its variants [1]. During the forensic analysis, the system log data is analyzed in an offline setting.

Dynamic Information Flow Tracking (DIFT) [2] is a widely used detection mechanism for offline analysis of APT. DIFT uses the information traces recorded in the system log for performing the security analysis [1]. The key idea behind DIFT is that it tags all suspicious input/data channels and tracks the propagation of the tagged information flows through the system. DIFT generates security analysis using a pre-specified set of heuristic security rules whenever it observes an unauthorized use of tagged data. These heuristic rules are typically defined by practicing experts based on the historical attack knowledge and knowledge about nominal system behavior [1].

Although, security rules incorporated in the DIFT mechanism cover a wide range of attacks, these security rules may not be capable of verifying the authenticity of information flows against all possible attacks, resulting in the generation of false-negatives and false-positives. Incorrectly identifying a benign (nominal user) flow as malicious is a false-positive (FP) and failing to identify a malicious (adversarial) flow is a false-negative (FN). For instance, while the security rules for buffer overflow protection [3] can be verified accurately, the security rules for web application vulnerabilities [4] cannot be accurately verified. Consequently, attacks that exploit web application may lead to incorrect conclusions by DIFT thereby generating FPs and FNs.

Additionally, limited availability of resources for defense along with the performance and memory overhead imposed by the defense mechanism on the system demands a resource-efficient detection technique. An analytical model of DIFT that captures the cost for performing security analysis and

S. Moothedath is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, 50011 USA. mshana@iastate.edu.

D. Sahabandu, L. Bushnell, and R. Poovendran are with the Department of Electrical and Computer Engineering, University of Washington, Seattle, WA 98195, USA. {sm15, sdinuka, lb2, rp3}@uw.edu.

J. Allen and W. Lee are with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30332 USA. jallen309@gatech.edu, wenke@cc.gatech.edu.

A. Clark is with the Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, MO 63130 USA. andrewclark@wustl.edu.

This work was supported by ONR grant N00014-16-1-2710 P00002, DARPA grant FA8650-15-C-7556, and NSF grant 2229876 and was supported in part by funds provided by the National Science Foundation (NSF), by the Department of Homeland Security, and by IBM. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF or its federal agency and industry partners.

the effectiveness of flow-tagging mechanisms will facilitate the trade-off between the effectiveness of defense and the resource efficiency. Further, such a model would enable the design of optimal security strategies while taking into account the generation of FP and FN.

In this paper, we provide an analytical model to enable DIFT to optimally select locations in the system to perform security analysis so as to maximize the probability of detection while minimizing the cost of detection. Our framework is based on the following insights. First, although both APT and DIFT are unaware of the other player's strategy, the effectiveness of the DIFT depends on the APT's strategy and the APT's probability of evading detection depends on the DIFT's strategy. This strategic interaction motivates a game-theoretic approach. Second, the efficiency of detection also depends on the effectiveness of performing security analysis at different locations in the system, and hence is determined by rates of FP and FN. Third, the game unfolds at multiple states between the entry points and the exit points of the attack where each state corresponds to the position of the tagged flows in the system. At each state, the defender decides whether to analyze a tagged flow and which one of the tagged flows to analyze (to avoid generation of FP and FN) while the adversary decides which process to transition to, at the cost of spending the defense resources. We formulate a stochastic game model that is played on an *information flow graph* (IFG). An IFG is a directed graph that expresses the history of a system's execution in terms of the spatio-temporal relationships between processes and subjects. The contributions of this paper are the following.

- We model the interaction of APT and DIFT as a nonzero-sum, two-player, turn-based stochastic game ($\mathbf{G}$) with finite state and action spaces. In the APT vs. DIFT game, the state of the game is the nodes of the IFG at which the tagged flows arrive. The location of the tagged flows is known to both players (APT and DIFT), and this makes the state of the game observable. However, each player is unaware of the other player's strategy. Also, before performing security analysis, DIFT cannot distinguish malicious and benign flows. This results in an asymmetric information structure.
- We analyze Nash equilibrium (NE) strategy of the game and show that an NE can be obtained from a solution to the *minimum capacity cut-set* problem on a flow-network constructed from the IFG of the system.
- We implement our algorithms and results on real-world data obtained for a data exfiltration attack using the Refinable Attack INvestigation (RAIN) system [1] augmented with FN and FP rates of the system.

The rest of the paper is organized as follows. Section II presents related work. Section III introduces preliminary concepts on information flow graph, APT attack, and a description of DIFT-based defense mechanism. Section IV describes the formulation of the turn-based stochastic game model. Section V gives the solution concept of the game and the equilibrium analysis results for computing optimal strategies of the players. Section VI illustrates the numerical results using data exfiltration attack data set collected using RAIN and augmented with FN and FP rates. Section VII concludes the paper.

II. Related Work

Stochastic games model interactions of multiple agents that jointly control the evolution of states of a stochastic dynamical system [5]. Stochastic games have been widely used to model security games [6], economic games [7], and resilience of cyber-physical systems [8], where each player tries to maximize its individual payoff. A brief overview of the existing results in stochastic games is given in [9].

Stochastic games have been used to address system security related problems. The interaction between malicious attackers and the intrusion detection system (IDS) is modeled using a stochastic game in [10]. A nonzero-sum stochastic game model is given in [11] to model network security configuration problem in distributed IDS. Then a value iteration based algorithm is proposed in [11] to find an ε-NE for an attacker model where multiple adversaries simultaneously attack a network. A zero-sum stochastic game is formulated in [12] for IDS in a communication or computer network with interdependent nodes and correlated security assets and vulnerabilities. Note that, [13] considered a finite-horizon game and [10]-[12] dealt with zero-sum stochastic games for IDS.

Game-theoretic models for resource-efficient detection of APTs are given in [14], [15], [16], and [17]. DIFT models with fixed trapping nodes are introduced and analyzed in [14], [15]. Paper [16] extended the models in [14], [15] by considering a DIFT model which selects the trapping nodes in a dynamic manner rather than being fixed and proposed a min-cut based solution approach. Reference [17] considered the detection of APTs when the attack consists of multiple attackers possibly with different capabilities and analyzed the best responses of the players. Note that the focus in [14], [15], [16], and [17] is resource-efficient detection of APTs and they do not consider the false-positives and false-negatives of the detection mechanism. In other words, the game models in [14]-[17] assume that security analysis performed by DIFT can verify the authenticity of tagged information flows accurately.

A stochastic game model for detecting APTs was recently introduced in the conference version of this work [18]. The proposed method in [18] analyzed a discounted stochastic game and presented a value iteration based algorithm to obtain an ε-NE of the discounted game. Due to the discounted nature of the payoff, the Nash equilibrium analysis of the game in [18] reduces to a nonlinear program (NLP). Solving an NLP is computationally challenging and the convergence of the algorithm in [18] is not guaranteed. This necessitates an alternate solution approach for solving the DIFT vs. APT game. In this paper, we solve an average reward (undiscounted) stochastic game and present a min-cut based approach with guaranteed convergence to compute optimal defense strategies.

In this paper we consider that the trapping nodes are generated dynamically and we extend our previous results on min-cut analysis in [16] in the following aspects. Since there can be a wide range of possible APT attacks, we recognize that

the security rules of DIFT may not be capable of verifying the authenticity of the information flows accurately thereby resulting in the generation of FP and FN. Consequently, the game model in this paper is stochastic unlike in [16]. We introduce a model with multiple information flows out of which one is malicious and the remaining are benign unlike in [16] which dealt with one information flow. Note that, while DIFT knows there is exactly one malicious flow in the system, it is unaware which flow is the malicious flow before performing security analysis. This results in an asymmetric information between players and hence the game is imperfect information game. In the current game model, although an information flow is concluded as benign at an initial inspection, it can be inspected again later during its propagation. The model in [16] inspects the flow exactly once as FN and FP rates are assumed to be zero. However, DIFT may analyze a flow multiple times and this will result in added resource cost. The game model considered in this paper captures the added resource cost incurred from reevaluation of the flows.

III. Preliminaries

In this section, we first describe the graphical representation of the system, denoted as the information flow graph, and then present the models of the attacker and the flow tracking-based defense.

A. Information Flow Graph (IFG)

Let $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G})$ represent the IFG of the system. $V_\mathcal{G} = \{v_1, \ldots, v_N\}$ consists of the processes (e.g., an instance of a computer program), files, and objects in the system and $E_\mathcal{G} \subseteq V_\mathcal{G} \times V_\mathcal{G}$ represents the information flows (directed) in the system from one node to the other. IFG is a weakly connected graph. IFG-based auditing is heavily desired by large enterprises and government agencies to detect APTs. We perform our game-theoretic analysis on the IFG of the system and use DIFT as the defense mechanism to detect APTs.

B. Attack Model: Advanced Persistent Threats (APTs)

APTs are intelligent attacks with specific targets. APTs are stealthy attacks that perform data exfiltration at an ultra-low-rate to avoid detection. Unlike classical malware, APT campaigns tend to involve multiple hosts, multiple systems, and extend over a long period of time, up to several months [19]. APTs are characterized by their abilities to render existing security mechanisms ineffective. APTs can evade security protection because existing mechanisms lack sufficient visibility into user, program, and operating system activities to ascertain the authenticity of an activity and the provenance of its data. The timeline and key stages of the APT lifecycle is described below and presented in Figure 1.

1. Reconnaissance: During the reconnaissance phase, the attacker will try to gain information about the system, such as what nodes are accessible on the system and the security defenses that are being used.
2. Initial Compromise: During the initial compromise stage, the attacker's goal is to gain access to an enterprise's network. In most cases, the attacker achieves this by exploiting a vulnerability or a social engineering trick, such as a phishing email.
3. Foothold Establishment: Once the attacker has completed the initial compromise, it will establish a persistent presence by opening up a communication channel with their Command & Control (C&C) server.
4. Lateral Movement: The attacker will increase its control of the system by moving laterally to new nodes in the system.
5. Target Attainment: Next, the attacker will try and escalate its privileges which may be necessary in order to access sensitive information, such as a proprietary source code or customer information.
6. Attack Completion: The final goal of the attacker is to deconstruct the attack, hopefully in a way to minimize its footprint in order to evade detection. For example, attackers may rely on deleting the system's log.

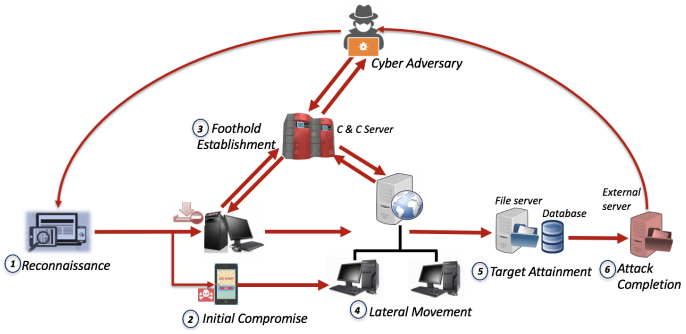


Figure 1: Schematic diagram of an APT attack timeline.

Let the set of possible entry points of the APT be denoted as $\lambda \subset V_\mathcal{G}$. During an attack, the goal of the APT is to capture a subset of nodes of the IFG referred to as *destinations* and denoted as $\mathcal{D} \subset V_\mathcal{G}$. Here, $\lambda \cap \mathcal{D} = \emptyset$. Once a foothold is established in the system, APT tries to elevate the privileges and proceeds to the destinations through more internal compromises and performs data exfiltration at an ultra-low-rate. In order to achieve this the APT performs operations in the system to transition through the nodes in $V_\mathcal{G}$ and arrive at some node in $\mathcal{D}$. In other words, the adversary (APT) selects paths in the IFG and performs transitions along the paths to reach the set $\mathcal{D}$ from the set λ .

The adversary in the DIFT vs. APT game has the following properties. We consider an APT attack that generates a single malicious information flow. The malicious flow originates at an entry point of the attack and the objective of the flow is to reach a destination. The state of the game is the set of nodes of the information flow graph at which the tagged flows arrive. APT observes the tagged flows and hence the state of the game is observable to APT. However, APT is unaware of the actions and the strategy of DIFT. Thus the information structure of APT is asymmetric with respect to that of the defender.

C. Defender Model: Dynamic Information Flow Tracking (DIFT)

The objective of the defender is to prevent any adversarial information flow traversing to the destination nodes. DIFT is a dynamic taint analysis based detection mechanism that consists of three main components: (i) tag sources, (ii) tag propagation rules, and (iii) trapping nodes. Trapping nodes are processes and objects (files and network endpoints) in the system that are considered as suspicious sources of information by the system. All data originating from a tag source is labeled or tagged and DIFT tracks the propagation of a tagged data through the system. Propagation of a tagged information flow through the system results in tagging of more information flows based on the tag propagation rules specified by the DIFT [20]. Tag propagation rules are defined based on two kinds of information flows in the system: explicit information flows and implicit information flows [21].

During the execution of a program in the system, DIFT keeps track of the tagged information flows and generates trapping nodes for any unauthorized use of tagged data that indicate a possible attack. DIFT invokes security analysis at the trapping nodes using the specified security rules to verify the authenticity of the tagged flow thereby concluding whether there is an attack or not. These security rules are pre-specified depending on the application running on the system [3]. Note that tagged (suspicious) flows consist of both benign and malicious flows. As the specified security rules may not necessarily cover all possible attacks by APTs, DIFT generates false-positives and false-negatives. An optimal selection of trapping nodes in the IFG is hence critical to detect a wide range of attacks in the system with minimum false-positives and false-negatives. Note that while tag sources and tag propagation rules are known to the attacker, the trapping nodes are dynamically generated during the operation of the system and hence is unknown to the attacker. Figure 2 shows a schematic diagram of the DIFT-based detection framework considered in this paper.

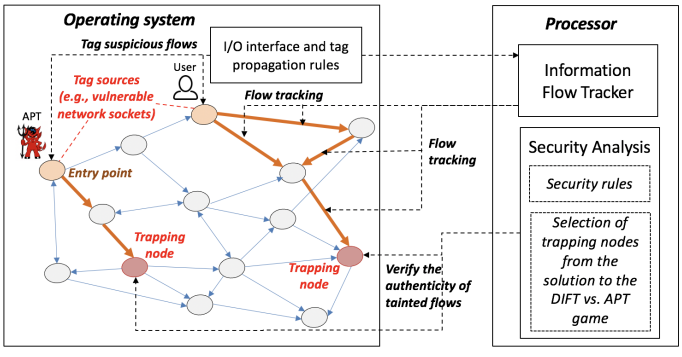


Figure 2: Schematic diagram of the DIFT detection framework. The nodes in the figure denote system components such as processes, files, and subjects.

The defender in the DIFT vs. APT game has the following properties. The state of the game, which is the nodes of the information flow graph at which the tagged flows arrive, is known to DIFT. Thus DIFT observes the state of the game. However, DIFT cannot distinguish a malicious and a benign flow before performing security analysis. Also, while DIFT knows that there is an APT in the system, DIFT is unaware of the actions and the strategy of APT.

IV. Turn-Based APT vs. DIFT Game $\mathbf{G}$

In this section, we model the interaction of the APT with the system during the different stages of the attack as a dynamic stochastic game between the defender (DIFT) and the adversary (APT). Let $\mathcal{P}_D$ be the defender player and $\mathcal{P}_A$ be the adversarial player. We consider a nonzero-sum turn-based game. The information structure of the players is asymmetric as the defender player does not know if a tagged (suspicious) flow is malicious or benign while the adversarial player knows this information. The game $\mathbf{G} = \{\mathbf{S}, s_0, \Sigma, \Delta, P\}$ unfolds on a finite state space, $\mathbf{S}$, with initial state, s_0, finite action space of players, Σ, labeled transitions $\Delta \subseteq \mathbf{S} \times \Sigma \times \mathbf{S}$, and transition probability matrix P.

A. State Space

We consider a turn-based game with the *state* of the game at time $t \in \{0,1,2,\ldots\}$ denoted by the random variable s_t. Let T denote the time horizon of the game, i.e., the game ends at time T and $s_{t'} = s_T$, for all $t' \geqslant T$. The state space $\mathbf{S}$ is partitioned into two sets, $\mathbf{S}_A$ and $\mathbf{S}_D$, such that $\mathbf{S}_A \cap \mathbf{S}_D = \emptyset$ and $\mathbf{S}_A \cup \mathbf{S}_D = \mathbf{S}$. The states that belong to the set $\mathbf{S}_A$ are referred to as the *adversary-controlled* states and the states in $\mathbf{S}_D$ are referred to as the *defender-controlled* states. Specifically, $\mathbf{S}_x$ is the subset of states at which player $\mathcal{P}_x$, where $x \in \{A, D\}$, controls the transitions. The state of the game at time t is the state of the system which corresponds to the position of the tagged (suspicious) flows at time t. Let W be the number of tagged flows that arrive into the system at time $t = 1$. In practice (from the analysis of log data), only a small fraction of nodes in the system receive a system call at the same time. Hence in our model we assume that $W << N$. We denote the state of the game at time t using the location of the W flows in the system and one additional bit which represents whether that state belong to $\mathbf{S}_A$ or $\mathbf{S}_D$.

Specifically, at time t, $s_t = (x, v_{i_1}, \ldots, v_{i_W})$, where $v_{i_k} \in V_{\mathcal{G}} \cup \{\phi, \tau\}$ and $x \in \{A, D\}$. Here, $x = A$ implies that $s_t \in \mathbf{S}_A$ and $x = D$ implies that $s_t \in \mathbf{S}_D$. Moreover, $v_{i_k} = \phi$ if the k^{th} information flow *drops* out, and $v_{i_k} = \tau$ if the k^{th} information flow is *trapped* by DIFT. At time $t \geqslant 1$, the state of the game is $s_t = (x, v_{i_1}, \ldots, v_{i_W})$, where $x \in \{A, D\}$ and $v_{i_k} \in V_{\mathcal{G}} \cup \{\phi, \tau\}$. Note that, out of the W flows one is a malicious flow and the remaining $(W-1)$ are benign flows. For notational convenience we consider the first flow is malicious (note that, defender does not have this information). While DIFT observes all the W flows, it cannot distinguish malicious and benign flows. APT, on the other hand, knows which flow is the malicious flow. Thus in the DIFT vs. APT game, APT and DIFT have asymmetric information.

Also, corresponding to a state s_t if $v_{i_k} = \phi$, for some $k \in \{1, \ldots, W\}$, then for all time $t' > t$ the state corresponding to the k^{th} flow remains ϕ, i.e., if a flow is dropped at time t it remains dropped through out the rest of the game.

B. Action Spaces

We first construct a directed flow-network $\mathcal{F} = (V_\mathcal{F}, E_\mathcal{F})$ from the IFG $\mathcal{G}$ by introducing a source node $s_\mathcal{F}$ with an outgoing edge to all the entry points and a sink node $t_\mathcal{F}$ with an incoming edge from all the destination nodes. Here $V_\mathcal{F} = V_\mathcal{G} \cup \{s_\mathcal{F}, t_\mathcal{F}\}$ and $E_\mathcal{F} = E_\mathcal{G} \cup \{s_\mathcal{F} \times \lambda\} \cup \{\mathcal{D} \times t_\mathcal{F}\}$.

Definition IV.1. *An attack path in the flow-network $\mathcal{F}$ is a simple directed path[1] from $s_\mathcal{F}$ to $t_\mathcal{F}$. The set of attack paths in $\mathcal{F}$ is denoted as $\Omega_\mathcal{D}$.*

The objective of the APT attack is to capture a destination node in set $\mathcal{D}$. To achieve the same, APT chooses transitions along a path from the set λ to the set $\mathcal{D}$ which translates into an attack path in the flow-network $\mathcal{F}$. We note that, any IFG that contains cycles can be converted into an acyclic IFG without losing any causal relationships between the components given in the original IFG. One such dependency-preserving conversion is node versioning [22]. We consider an acyclic IFG and hence all $\omega \in \Omega_\mathcal{D}$ are simple directed paths.

Using the construction of the flow-network, we denote the initial state of the game at $t = 0$ is denoted as $s_0 = (A, s_\mathcal{F}, \ldots, s_\mathcal{F})$, since at $t = 0$ all flows originate at the source node $s_\mathcal{F}$. At a state s_t, either $\mathcal{P}_D$ or $\mathcal{P}_A$ chooses an action from their respective action sets denoted by $\mathcal{A}_D$ and $\mathcal{A}_A$, depending on whether $s_t \in \mathbf{S}_D$ or $s_t \in \mathbf{S}_A$. If $s_t \in \mathbf{S}_A$, the adversary decides whether to quit the attack by dropping the information flow or to continue the attack. If the adversary decides not to quit the attack, then it selects which neighboring node of the IFG to transition from the current node so as to reach set $\mathcal{D}$. In other words, the adversary either drops the malicious flow or performs a transition along a path in $\Omega_\mathcal{D}$ and hence $\mathcal{A}_A := \{\phi\} \cup V_\mathcal{G}$. Specifically, at state $s_t = (A, v_{i_1}, \ldots, v_{i_W})$, where $v_{i_1} \in V_\mathcal{G}$ and $v_{i_1} \notin \mathcal{D}$, $\mathcal{A}_A(s_t) \in \{\phi\} \cup \mathcal{N}(v_{i_1})$, where $\mathcal{N}(v_{i_1}) = \{v_j \in V_\mathcal{G} : (v_{i_1}, v_j) \in E_\mathcal{G}\}$. On the other hand, if $s_t \in \mathbf{S}_D$, the action of the defender is to decide whether to perform security analysis on an information flow or not. If the defender chooses to perform security analysis, it also selects a flow to analyze. A node at which security analysis is performed is referred to as a *trapping node*. Consider a state $s_t = (D, v_{i_1}, \ldots, v_{i_W})$ and let defender decides to perform security analysis on the flow at v_{i_1}. Then v_{i_1} is chosen as a trapping node. Thus, $\mathcal{A}_D := \{0\} \cup V_\mathcal{G}$, where 0 represents not analyzing any flow. The defender performs security analysis on at most one flow in a state as there is only malicious flow and also it is not possible to perform analysis on the information flows at the entry points of the attack, λ. Thus at state $s_t = (D, v_{i_1}, \ldots, v_{i_W})$, $\mathcal{A}_D(s_t) \in \{0\} \cup \{v_{i_k} : v_{i_k} \notin \lambda, k \in \{1, \ldots, W\}\}$. We also note that the information flow chosen by the DIFT to perform security analysis can be either the malicious flow or a benign flow.

A state s_t is said to be an *absorbing state* if the game terminates at s_t and the action sets of the players are empty, i.e., $\mathcal{A}_A(s_t) = \mathcal{A}_D(s_t) = \emptyset$. In $\mathbf{G}$ a state $s_t = (x, v_{i_1}, \ldots, v_{i_W})$, where $x \in \{A, D\}$, is absorbing if one of the three cases holds:

(i) $v_{i_1} \in \mathcal{D} \subset V_\mathcal{G}$,
(ii) $v_{i_k} = \tau$ for some $k \in \{1, \ldots, W\}$,
(iii) $v_{i_k} = \phi$ for all $k \in \{1, \ldots, W\}$.

Case (i) corresponds to adversary reaching a destination, case (ii) corresponds to the case where DIFT performed security analysis on an information flow and also concluded it as malicious, i.e., a flow is trapped, and case (iii) corresponds to all W flows dropping out.

C. State Transitions

Definition IV.2. *Time t is said to be the termination time of the game $\mathbf{G}$ if it satisfies one of the following condition: (a) $t = T$ and (b) the state of the game $s_t = (x, v_{i_1}, \ldots, v_{i_W})$, where $x \in \{A, D\}$, is an absorbing state.*

We use the notation W to denote the number of tagged information flows that are recorded at the first instant of time, i.e., $t = 1$. The log recording system (RAIN) used in our work uses the system call in [23] to create a timestamp for each audit log. At every instant of the log recording, RAIN records at most one system call for each system component (processes, files). As a result, the W information flows arrive at distinct nodes of the information flow graph as it is not possible for multiple flows to arrive at one node at the same instant of time. Let the state of the game at $t = 1$ be $s_t = (D, v_{i_1}, \ldots, v_{i_W})$. Here $\{v_{i_1}, \ldots, v_{i_W}\}$ are the nodes of the IFG at which the flows arrive. The DIFT-based defense mechanism observes the flows. The player $\mathcal{P}_D$ now chooses an action from the set $\mathcal{A}_D(s_t) \in \{0, v_{i_1}, \ldots, v_{i_W}\}$. At state s_t, there are three possibilities: (1) $\mathcal{P}_D$ does not analyze any flow, (2) $\mathcal{P}_D$ analyze the malicious flow, and (3) $\mathcal{P}_D$ analyze a benign flow. Based on the action chosen by $\mathcal{P}_D$, i.e., $\mathcal{A}_D(s_t)$, the next possible state of the game s_{t+1} is defined. At s_{t+1} $\mathcal{P}_A$ has two possibilities to choose from: (a) to quit the flow, ϕ and (b) to transition to an out-neighbor of node v_{i_1}, $\mathcal{N}(v_{i_1})$, i.e., $\mathcal{A}_A(s_{t+1}) \in \{\phi\} \cup \mathcal{N}(v_{i_1})$. Based on the action of $\mathcal{P}_A$ and the distribution of the benign flows in the system, the next state s_{t+2} is arrived. At $t+2$, $\mathcal{P}_D$ again chooses its action and the game continues in a turn-based fashion. Note that the DIFT will analyze only one information flow at a time t as we consider a single malicious flow in the system.

For case (1) the action of $\mathcal{P}_D$ is $\mathcal{A}_D(s_t) = 0$, i.e., DIFT does not perform security analysis on any of the W information flows. Then the next state of the game $s_{t+1} = (A, v_{i_1}, \ldots, v_{i_W})$. For case (2), $\mathcal{P}_D$ chooses action $\mathcal{A}_D(s_t) = v_{i_1}$ and the next state of the game is $s_{t+1} = (A, v_{i_1}, \ldots, v_{i_W})$ with probability FN and $s_{t+1} = (A, \tau, v_{i_2}, \ldots, v_{i_W})$ with probability $1 - FN$. For case (3), the player $\mathcal{P}_D$ chooses action $\mathcal{A}_D(s_t) = v_{i_k}$, $k \neq 1$, i.e., $k \in \{2, \ldots, W\}$. Then the next state of the game is $s_{t+1} = (A, v_{i_1}, \ldots, v_{i_{k-1}}, \tau, v_{i_{k+1}}, \ldots, v_{i_W})$ with probability FP and $s_{t+1} = (A, v_{i_1}, \ldots, v_{i_W})$ with probability $1 - FP$. Here, FN and FP are the rate of false-negatives and false-positives in the DIFT-architecture which are empirically computed. The values of FP and FN depend on the security rules and vary across the different DIFT-architectures and they determine the transition probabilities P of the stochastic game. In order to reduce false-positives and false-negatives in the system, the security rules must be capable of identifying the behavior and predicting the intend of information flows, i.e., determine

[1] A directed path is said to be a simple directed path if there are no cycles or loops in the path.

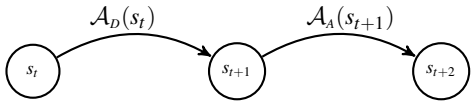


Figure 3: Schematic diagram showing the turn-based game $\mathbf{G}$. At state $s_t \in \mathbf{S}_D$ the defender player $\mathcal{P}_D$ chooses an action and the game transitions to the next state s_{t+1}. At time $t+1$, the adversarial player $\mathcal{P}_A$ observes the current state $s_{t+1} \in \mathbf{S}_A$ and then chooses an action based on which the next state $s_{t+2} \in \mathbf{S}_D$ is defined. At $t+2$ $\mathcal{P}_D$ again chooses its action and the game continues.

whether an unknown flow is indeed malicious, or whether it is benign flow that is exhibiting malware-like behavior [24].

If $t+1$ is not a terminating time instant, then the adversarial player $\mathcal{P}_A$ chooses action $\mathcal{A}_A(s_{t+1}) \in \{\phi\} \cup \mathscr{N}(v_{i_1})$. The remaining $W-1$ flows follow the benign flow distribution in the system, which is computed empirically from the nominal system operation and known. Let $\pi_B : v_i \in V_{\mathcal{G}} \to [0,1]^{|\mathscr{N}(v_i)|+1}$ denote the benign flow distribution. If the action of $\mathcal{P}_A$ is ϕ, then $s_{t+2} = (D, \phi, v_{j_2}, \ldots, v_{j_W})$, where $v_{j_2}, \ldots, v_{j_W}$ depend on the distribution π_B. Here $\{v_{j_2}, \ldots, v_{j_W}\} \in \{\phi\} \cup V_{\mathcal{G}}$. Note that a benign flow can also drop out. Now $\mathcal{P}_D$ chooses its action and the game continues. To summarize, given $s_t = (D, v_{i_1}, \ldots, v_{i_W})$ (w.p stands for with probability), state transitions and the corresponding transition probabilities, given by P, are

$$s_{t+1} = \begin{cases} (A, v_{i_1}, \ldots, v_{i_W}), & \text{w.p } 1 \quad \text{if } \mathcal{A}_D(s_t) = 0, \\ (A, v_{i_1}, \ldots, v_{i_W}), & \text{w.p } FN \quad \text{if } \mathcal{A}_D(s_t) = v_{i_1}, \\ (A, \tau, \ldots, v_{i_W}), & \text{w.p } 1-FN \text{ if } \mathcal{A}_D(s_t) = v_{i_1}, \\ (A, v_{i_1}, \ldots, v_{i_W}), & \text{w.p } 1-FP \text{ if } \mathcal{A}_D(s_t) = v_{i_k}, k \neq 1 \\ (A, v_{i_1}, \ldots, v_{i_{k-1}}, \tau, \ldots, v_{i_W}), & \text{w.p } FP \text{ if } \mathcal{A}_D(s_t) = v_{i_k}, k \neq 1. \end{cases} \tag{1}$$

and

$$s_{t+2} = \begin{cases} (D, v_{j_1}, \ldots, v_{j_W}), & \text{if } \mathcal{A}_A(s_{t+1}) \in \mathscr{N}(v_{i_1}), \\ (D, \phi, v_{j_2}, \ldots, v_{j_W}), & \text{if } \mathcal{A}_A(s_{t+1}) = \phi. \end{cases} \tag{2}$$

A schematic diagram that illustrates the transitions in the turn-based game $\mathbf{G}$ is given in Figure 3.

D. Strategies of the Players

A strategy is a rule that each player uses to select actions at every step of the game. We consider *mixed* (stochastic) and *behavioral* player strategies. Since the strategy is mixed, at a state, $\mathcal{P}_D$ and $\mathcal{P}_A$ select an action from the action set $\mathcal{A}_D$ and $\mathcal{A}_A$, respectively, based on some probability distribution. Further, since the strategy is behavioral, the probability distribution on the action set at a time instant t depends on all the states traversed by the game and the actions taken by the players until time t.

Let d_t, a_t be the action of the defender and the attacker, respectively, at time t. We denote the information available to the players $\mathcal{P}_D$ and $\mathcal{P}_A$ at time $t \leqslant T$ by $\mathbf{Y}_t$ and $\mathbf{Z}_t$, respectively. Then

$$\begin{aligned} \mathbf{Y}_t &:= \{s_0, d_0, s_1, d_1, \ldots, s_{t-1}, d_{t-1}, s_t\}, \\ \mathbf{Z}_t &:= \{s_0, a_0, s_1, a_1, \ldots, s_{t-1}, a_{t-1}, s_t\}. \end{aligned} \tag{3}$$

We denote the set of all possible outcomes for $\mathbf{Y}_t$ and $\mathbf{Z}_t$ at time t using $\mathcal{Y}_t^\star$ and $\mathcal{Z}_t^\star$, respectively. Let the set of all pure behavioral strategies of $\mathcal{P}_D$ and $\mathcal{P}_A$ for game $\mathbf{G}$ be $\bar{\mathbf{P}}_D$ and $\bar{\mathbf{P}}_A$, respectively. Define $\Delta\bar{\mathbf{P}}_D$ ($\Delta\bar{\mathbf{P}}_A$) as the simplex of $\bar{\mathbf{P}}_D$ ($\bar{\mathbf{P}}_A$), i.e., the set of probability distributions over $\bar{\mathbf{P}}_D$ ($\bar{\mathbf{P}}_A$). A mixed behavioral strategy of player $\mathcal{P}_D$ is a mapping given by $\mathcal{Y}_t^\star \to \Delta\bar{\mathbf{P}}_D$, for $t \geq 0$, where $\Delta\bar{\mathbf{P}}_D$ denote the set of all mixed behavioral strategies for $\mathcal{P}_D$. Similarly, a mixed behavioral strategy of player $\mathcal{P}_A$ is given by the mapping $\mathcal{Z}_t^\star \to \Delta\bar{\mathbf{P}}_A$, for $t \geq 0$, where $\Delta\bar{\mathbf{P}}_A$ denote the set of all mixed behavioral strategies for $\mathcal{P}_A$. Further, let the set of all mixed policies of $\mathcal{P}_D$ and $\mathcal{P}_A$ for game $\mathbf{G}$ be $\mathbf{P}_D$ and $\mathbf{P}_A$, respectively. Then a policy $p_D \in \mathbf{P}_D$ and $p_A \in \mathbf{P}_A$ are said to be *pure* or *deterministic* policy if all the entries of the vectors p_D and p_A, respectively, belong to the set $\{0,1\}$.

E. Payoffs to the Players

The payoff functions of $\mathcal{P}_D$ and $\mathcal{P}_A$ are denoted by U_D and U_A, respectively. U_D consists of three components: (i) resource cost $\mathcal{C}_D(v_i) < 0$ for performing security analysis of a flow at a node $v_i \in V_{\mathcal{G}}$, (ii) penalty $\beta_D < 0$ for adversary reaching a destination node in $\mathcal{D}$, and (iii) reward $\alpha_D > 0$ for detecting the adversary. Similarly, U_A consists of two components: (i) a reward $\beta_A > 0$ for reaching a destination node in the set $\mathcal{D}$, and (ii) penalty $\alpha_A < 0$ for getting detected by the defender. The reward and penalty parameters and the resource cost of DIFT and APT do not necessarily induce a zero-sum scenario where the payoff that is gained by one player is lost by the other. Therefore, we considered a non-zero sum payoff structure for the DIFT vs. APT game.

Let the payoff of player $\mathcal{P}_x$ at an absorbing state s_t be denoted as $c^x(s_t)$, where $x \in \{A, D\}$. Also, let the payoff of $\mathcal{P}_D$ at a non-absorbing defense-controlled state s_t with action d_t be $r^D(s_t, d_t)$, and the payoff of $\mathcal{P}_A$ at a non-absorbing adversary-controlled state s_t with action a_t be $r^A(s_t, a_t)$. At each state in the game, s_t at time t, where $t < T$ and s_t is a non-absorbing state, player chooses its action (d_t for $s_t \in \mathbf{S}_D$ and a_t for $s_t \in \mathbf{S}_A$) and receives payoff $r^D(s_t, d_t)$ and $r^A(s_t, a_t)$, respectively, and the game transitions to a next state s_{t+1}. This is continued until they reach an absorbing state and incur $c^A(s_t)$ and $c^D(s_t)$, respectively, or the game arrives at the horizon, i.e., $t = T$. Then,

$$r^A(s_t, a_t) = 0 \quad \text{for all} \quad s_t, a_t, \tag{4}$$

$$c^A(s_t) = \begin{cases} \alpha_A, & s_t \in \mathbf{S}_A,\ s_t = (A, \tau, \ldots, v_{i_W}) \\ \beta_A, & s_t \in \mathbf{S}_A,\ s_t = (A, v_{i_1}, \ldots, \tau, \ldots, v_{i_W}) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

$$c^D(s_t) = \begin{cases} \beta_D, & s_t \in \mathbf{S}_D,\ s_t = (D, v_{i_1}, \ldots, v_{i_W}), v_{i_1} \in \mathcal{D} \\ \alpha_D, & s_t \in \mathbf{S}_D,\ s_t = (A, \tau, \ldots, v_{i_W}) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

$$r^D(s_t, d_t) = \begin{cases} 0, & s_t \in \mathbf{S}_D,\ d_t = 0 \\ \mathcal{C}_D(v_{i_k}), & s_t \in \mathbf{S}_D,\ d_t = v_{i_k}, k \in \{1, \ldots, W\}. \end{cases} \tag{7}$$

As the initial state of the game is $\mathbf{s}_0$, for a strategy pair (p_D, p_A) the expected payoffs of the players are

$$U_A(p_D,p_A) = \mathbb{E}_{\mathbf{s}_0,p_A,p_D}\left[\sum_{t=0}^{T}(R_t^A)\right] \text{ and} \quad (8)$$

$$U_D(p_D,p_A) = \mathbb{E}_{\mathbf{s}_0,p_A,p_D}\left[\sum_{t=0}^{T}(R_t^D)\right], \quad (9)$$

where $\mathbb{E}_{\mathbf{s}_0,p_A,p_D}$ denotes the expectation with respect to $\mathbf{s}_0, p_A$, and p_D and from Eqs. (4)-(7)

$$R_t^x = \begin{cases} r^D(s_t,d_t), & x=D,\ s_t\in\mathbf{S}_D,\ t<T \\ r^A(s_t,a_t), & x=A,\ s_t\in\mathbf{S}_A,\ t<T \\ c^D(s_t), & s_t\in\mathbf{S}_D,\ t=T, \\ c^A(s_t), & s_t\in\mathbf{S}_A,\ t=T. \end{cases} \quad (10)$$

We incorporate the idea of maximizing the probability of detection and minimizing the probability of adversary evading detection using the terms p_T and p_R, respectively. Further, we capture the false-positives generated in the system using the term p_{FP}. Note that, generation of false-positive implies that the defender failed to capture the adversary and concluded a benign flow is malicious. This means that the adversary will attain the target as the malicious flow evaded detection. Given a set of strategies (p_D, p_A), where $p_D \in \mathbf{P}_D$ and $p_A \in \mathbf{P}_A$, and benign distribution π_B, the payoff functions of the players, i.e., Eqs.(8), (9), can be rewritten using Eqs. (4)-(7), (10) as

$$U_D(p_D,p_A) = p_T\,\alpha_D + (p_R+p_{FP})\,\beta_D + \sum_{\mathbf{s}\in\mathbf{S}_D}\sum_{v_i\in\mathbf{s}}\Big(p_D(v_i)\mathcal{C}_D(v_i)\Big), \quad (11)$$

$$U_A(p_D,p_A) = \Big(p_T\,\alpha_A + (p_R+p_{FP})\,\beta_A\Big). \quad (12)$$

Here p_T is the cumulative probability that the adversarial flow is detected by the defender. The term $p_T\,\alpha_D$ captures the reward received by DIFT for detecting the attack and the term $p_T\,\alpha_A$ captures the penalty incurred by APT for getting detected. Also, p_R denotes the cumulative probability that the adversarial flow reaches a destination and p_{FP} denotes the cumulative probability that a benign flow is concluded as malicious (i.e., trapped) by the defender, i.e., false-positive. The term $(p_R+p_{FP})\,\beta_D$ captures the penalty incurred by DIFT for not detecting the attack and the term $(p_R+p_{FP})\,\beta_A$ captures the reward received by the APT for not getting detected. Recall that $p_D(v_i)$ denotes the probability with which DIFT selects node v_i as a security check point (trap). The term $\sum_{\mathbf{s}\in\mathbf{S}_D}\sum_{v_i\in\mathbf{s}}(p_D(v_i)\mathcal{C}_D(v_i))$ captures the total resource cost associated with DIFT for performing security analysis. Note that p_T, p_R are functions of p_D and p_A, and p_{FP} is a function of p_D and π_B. Our focus is to compute a limiting average equilibrium of the nonzero-sum stochastic game.

V. Computation of Optimal Strategy

A. Solution Concept

In this subsection, we present the solution concept of the game.

Definition V.1. *Let $p_A : \mathcal{Z}^\star \to [0,1]^{|\mathcal{A}_A|}$ denote a strategy of the adversary. Also let $p_D : \mathcal{Y}^\star \to [0,1]^{|\mathcal{A}_D|}$ denote a strategy of the defender, i.e., the probability of performing security analysis at a state. Then the best response of the defender is given by*

$$BR(p_A) = \arg\max_{p_D\in\mathbf{P}_D}\{U_D(p_D,p_A)\}.$$

Similarly, the best responses of the adversary are given by

$$BR(p_D) = \arg\max_{p_A\in\mathbf{P}_A}\{U_A(p_D,p_A)\}.$$

The best response of the defender is the set of defense strategies that maximize the payoff of the defender for a given adversarial strategy and known benign flow distribution. The best response of the adversary is a set of transition strategies, for a given defender strategy and known benign flow distribution, that maximizes the probability of adversary reaching a destination node without getting detected.

Definition V.2. *A pair of strategies (p_D, p_A) is said to be a Nash equilibrium (NE) if*

$$p_D \in BR(p_A) \text{ and } p_A \in BR(p_D).$$

An NE is a pair of strategies such that no player can benefit through unilateral deviation, i.e., by changing the strategy while the other player keeps the strategy unchanged. We prove existence of an NE for the APT vs. DIFT game in Proposition V.5.

Lemma V.3. *Consider the APT vs. DIFT game $\mathbf{G} = \{\mathbf{S}, \mathbf{s}_0, \Sigma, \Delta, P\}$. Let N be the number of nodes in the IFG and W be the number of flows that arrive into the system at time $t = 1$. Then $|\mathbf{S}| = O(N^W)$.*

Proof. We prove the result by showing that $|\mathbf{S}_A| = O(N^W)$ and $|\mathbf{S}_D| = O(N^W)$. Consider an arbitrary state $\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W}) \in \mathbf{S}\setminus\mathbf{s}_0$, where $x \in \{A, D\}$. Here, $v_{i_k} \in V_\mathcal{G} \cup \{\phi, \tau\}$ for all $k \in \{1, \ldots, W\}$. All the W tagged flows arrive at distinct nodes in the set $V_\mathcal{G}$ of the IFG as a process or file in the system receive exactly one information flow at a particular time. Therefore, for a state $\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W})$, where $v_{i_k} \in V_\mathcal{G}$ for all $k \in \{1, \ldots, W\}$, $i_a \neq i_b$ for $a, b \in \{1, \ldots, W\}$. Also, the defender performs security analysis on one tagged flow at a time and hence there is no state $\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W})$ with $v_{i_a} = v_{i_b} = \tau$ for $a \neq b$. Repetition of v_{i_k}'s is possible only for states with ϕ. Hence $\mathbf{s}$ belongs to one of the two types: (a) $\{\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W}) : v_{i_k} \in V_\mathcal{G} \cup \{\tau\}\}$ and (b) $\{\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W}) : v_{i_k} \in V_\mathcal{G} \cup \{\phi\}$ such that $v_{i_k} = \phi$ for at least some $k \in \{1, \ldots, W\}\}$.

Case (a) corresponds to selecting W items from a set of $N+1$ items without any repetitions. The cardinality of this set is $\binom{N+1}{W} = \frac{(N+1)!}{(W)!(N+1-W)!} = O(N^W)$. Case (b) corresponds to states with one or more ϕ entries. Note that, here repetitions are allowed only for ϕ. The cardinality of this set is $\binom{N+1}{W-1} + \binom{N+1}{W-2} + \ldots + \binom{N+1}{2} + \binom{N+1}{1} = O(N^{W-2})$. Additionally, there is an initial state $\mathbf{s}_0$. From (a) and (b), the number of possible cases for $\mathbf{s}$ is $O(N^W)$. Since $\mathbf{s} = (x, v_{i_1}, \ldots, v_{i_W})$, where $x \in \{A, D\}$, we get $|\mathbf{S}_A| = O(N^W)$ and $|\mathbf{S}_D| = O(N^W)$. As $\mathbf{S} = \mathbf{S}_A \cup \mathbf{S}_D$, we get $|\mathbf{S}| = O(N^W)$. □

It is shown that there exists an NE for nonzero-sum discounted stochastic games [25]. However, the existence of NE for nonzero-sum undiscounted stochastic games is open when

the time horizon is infinite, i.e., $T = \infty$. In the result below we prove the existence of NE for the game **G**. The existence is shown by proving that time horizon is indeed finite for **G** and then invoking the existence of NE for finite horizon undiscounted games.

Proposition V.4. *Let T denote the termination time of the APT vs. DIFT game. The APT vs. DIFT game,* **G***, terminates in at most is $2(N-1)$ steps, i.e., $T \leqslant 2(N-1)$.*

Proof. The action set $\mathcal{A}_A$ of the adversary player (APT) is to choose a transition along an attack path of the flow-network $\mathcal{F}$ (Definition IV.1) or to drop out at some point. Let $\Omega_{\mathcal{D}}$ denote the set of attack paths in $\mathcal{F}$. Consider an arbitrary attack path $\hat{\omega} \in \Omega_{\mathcal{D}}$. Recall that $\hat{\omega}$ is a simple path as an APT due to its stealthy behavior will not traverse in cycles through the system. Hence the length of $\hat{\omega}$ is at most $N-1$, as $\hat{\omega}$ is a simple directed path. Thus in any run of the game, the adversary can take at most $N-1$ transitions. As **G** is a turn-based game, the game terminates in at most $2(N-1)$ steps. □

Proposition V.5. *There exists a Nash equilibrium for the APT vs. DIFT game* **G**.

Proof. It is shown in [13] that there exists a Nash equilibrium for a nonzero-sum stochastic game with asymmetric information structure, under stochastic behavioral strategies, when the time horizon is finite. Proposition V.4 proves that the APT vs. DIFT game will terminate in $2N$ number of steps. Further, the behavioral strategy space is a subset of the strategy space $\mathbf{P}_A \times \mathbf{P}_D$. Hence by the result in [13], the proof follows. □

B. Solution Approach

In this section, we compute an NE of the DIFT vs. APT game **G**. We first derive the *BR*s of the players separately and then use this derivation to construct an NE of the overall game. Our approach is based on a minimum capacity cut-set formulation on a flow-network constructed from the information flow graph of the system. Consider the flow-network $\mathcal{F} = (V_{\mathcal{F}}, E_{\mathcal{F}})$, where $V_{\mathcal{F}} = V_{\mathcal{G}} \cup \{s_{\mathcal{F}}, t_{\mathcal{F}}\}$ and $E_{\mathcal{F}} = E_{\mathcal{G}} \cup \{s_{\mathcal{F}} \times \lambda\} \cup \{\mathcal{D} \times t_{\mathcal{F}}\}$. Then, a *cut* of $\mathcal{F}$ is defined below.

Definition V.6. *In a flow-network $\mathcal{F}$ with vertex set $V_{\mathcal{F}}$ and directed edge set $E_{\mathcal{F}}$, the cut induced by $\hat{\mathcal{S}} \subset V_{\mathcal{F}}$ is a subset of edges $\kappa(\hat{\mathcal{S}}) \subseteq E_{\mathcal{F}}$ such that for every $(u,v) \in \kappa(\hat{\mathcal{S}})$, $|\{u,v\} \cap \hat{\mathcal{S}}| = 1$. Further, given edge capacity vector $c_{\mathcal{F}} : E_{\mathcal{F}} \to \mathbb{R}_+$, the capacity of a cut $\kappa(\hat{\mathcal{S}})$, is defined as the sum of the capacities of the edges in the cut, i.e., $c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}})) = \sum_{e \in \kappa(\hat{\mathcal{S}})} c_{\mathcal{F}}(e)$.*

The *(source-sink)-min-cut problem* aims to find a cut $\kappa(\hat{\mathcal{S}}^\star)$ of $\hat{\mathcal{S}}^\star \subset V_{\mathcal{F}}$ such that $c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}}^\star)) \leqslant c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}}))$ for any cut $\kappa(\hat{\mathcal{S}})$ of $\hat{\mathcal{S}} \subset V_{\mathcal{F}}$ satisfying $s_{\mathcal{F}} \in \hat{\mathcal{S}}$ and $t_{\mathcal{F}} \notin \hat{\mathcal{S}}$. The (source-sink)-min-cut problem is well studied and there exist algorithms to find a min-cut in time polynomial in $|V_{\mathcal{F}}|$ and $|E_{\mathcal{F}}|$ [26]. In our approach to compute NE, which is detailed later in the section, we find a min-cut through nodes of the flow-network instead of the edges. Hence we transform the node version of the min-cut problem to an equivalent edge version. For this, we introduce an edge corresponding to each node in $\mathcal{F}$ except the source and sink nodes. The transformed flow-network is denoted as $\overline{\mathcal{F}} = (\overline{V}_{\mathcal{F}}, \overline{E}_{\mathcal{F}})$, where $\overline{V}_{\mathcal{F}} = V_{\mathcal{F}} \cup V'_{\mathcal{G}}$ and $\overline{E}_{\mathcal{F}} = E'_{\mathcal{F}} \cup E'_{\mathcal{G}} \cup E_{\lambda} \cup E_D$ with $V'_{\mathcal{G}} = \{v'_1, \ldots, v'_N\}$, $E'_{\mathcal{F}} = \{(v'_i, v_j) : (v_i, v_j) \in E_{\mathcal{G}}\}$, $E_{\lambda} = \{(s_{\mathcal{F}}, v_i) : v_i \in \lambda\}$, $E_D = \{(v'_i, t_{\mathcal{F}}) : v_i \in \mathcal{D}\}$, and $E'_{\mathcal{G}} = \{(v_i, v'_i) : i = 1, \ldots, N\}$. The capacity vector associated with the edges in $\overline{\mathcal{F}}$ is given by

$$c_{\mathcal{F}}(e) := \begin{cases} C_D(v_i), & e \in E'_{\mathcal{G}} \\ \infty, & \text{otherwise} \end{cases} \tag{13}$$

Note that $E'_{\mathcal{G}}$ is a cut of $\overline{\mathcal{F}}$ and $\sum_{e \in E'_{\mathcal{G}}} c_{\mathcal{F}}(e) < \infty$. Hence any minimum capacity cut in $\overline{\mathcal{F}}$ corresponds to edges from the set $E'_{\mathcal{G}}$ as the capacity of the remaining edges is ∞ as shown in Eq. (13). Further, the capacity of an edge in set $(v_i, v'_i) \in E'_{\mathcal{G}}$ corresponds to the cost of conducting the security analysis at node v_i. Thus a minimum capacity cut in $\overline{\mathcal{F}}$ corresponds to a cut node set of the IFG with minimum total cost of performing security analysis.

Let $\kappa(\hat{\mathcal{S}}^\star)$ denotes an optimal solution to the (source-sink)-min-cut problem on $\overline{\mathcal{F}}$. Then $\kappa(\hat{\mathcal{S}}^\star) \subseteq E'_{\mathcal{G}}$ and $c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}}^\star)) < \infty$. The nodes corresponding to the min-cut is

$$\hat{\mathcal{S}}^\star := \{v_i : (v_i, v'_i) \in \kappa(\hat{\mathcal{S}}^\star)\}. \tag{14}$$

In the APT vs. DIFT game, the aim of the defender is to optimally select trapping nodes in the IFG, i.e., nodes of IFG to perform security analysis, such that no adversarial flow reaches some node in $\mathcal{D}$. In other words, defender ensures that all adversarial flows that originate in node $s_{\mathcal{F}}$ gets detected before reaching node $t_{\mathcal{F}}$. In order to ensure security, the defender must select at least one node in all possible paths from $s_{\mathcal{F}}$ to $t_{\mathcal{F}}$ as a trapping node. In the equilibrium analysis of the game we prove that an optimal strategy of the defender is indeed to select the min-cut nodes of the flow-network as trapping nodes. The objective of the adversary is to optimally choose the transitions in such a way that the probability of reaching $t_{\mathcal{F}}$ is maximum. The adversary hence plans its transitions to select an *attack path* with least probability of detection.

The result below proves that an NE of game **G** is represented by the nodes corresponding to a minimum capacity cut in the flow-network $\overline{\mathcal{F}}$. Note that, the solution to the min-cut problem may not be unique. Consequently, there may exist multiple NE for the game **G**.

Theorem V.7. *Every NE (p_A, p_D) of the APT vs. DIFT game* **G** *satisfies the following properties:*

1) *The defender's strategy p_D selects all the nodes in $\hat{\mathcal{S}}^\star$ as trapping nodes, where $\hat{\mathcal{S}}^\star$ is a set of min-cut nodes of the flow network $\overline{\mathcal{F}}$.*
2) *The adversary's strategy p_A chooses transitions such that each attack path passes through exactly one node in the set $\hat{\mathcal{S}}^\star$.*

We prove Theorem V.7 by invoking the property that at NE every player plays a best response against the other players simultaneously. We first present prove the best response results, Lemma V.8, Lemma V.10, and Lemma V.11, and then present the proof of Theorem V.7.

Lemma V.8 gives the best response of the adversary for a given defender strategy that selects the min-cut nodes of $\overline{\mathcal{F}}$ as the trapping nodes for analyzing the flows.

Lemma V.8. *Let $\Omega_{\mathcal{D}}$ be the set of all attack paths in $\mathcal{F}$. Consider a defender strategy p_D in which only the min-cut nodes $\hat{\mathcal{S}}^\star$ of the flow-network $\overline{\mathcal{F}}$ are chosen as trapping nodes with nonzero probability. Then, the best response of the adversary, $BR(p_D)$, is to choose the transitions in such a way that all attack paths which have nonzero probability under $BR(p_D)$ pass through exactly one node in $\hat{\mathcal{S}}^\star$.*

Proof. Consider the payoff function of the adversary, $U_A(p_D, p_A) = (p_T\,\alpha_A + (p_R + p_{FP})\,\beta_A)$. Here, p_T and p_R are functions of p_D and p_A. However, p_{FP} depends only on p_D and π_B. Thus for a given p_D, π_B, the probabilities p_T and p_R vary depending on p_A, however, p_{FP} is a constant. We prove the result using a contradiction argument. Suppose the best response of $\mathcal{P}_A$ is a strategy p'_A such that there exists a path $\hat{\omega} \in \Omega_{\mathcal{D}}$ that passes through two nodes, say $v_i, v_j \in \hat{\mathcal{S}}^\star \subset V_{\mathcal{G}}$, and $\pi_A(\hat{\omega}) \neq 0$, where $\pi_A(\hat{\omega})$ is the probability of attack path $\hat{\omega}$ under p'_A. Note that $\hat{\mathcal{S}}^\star$ is a min-cut and $v_i, v_j \in \hat{\mathcal{S}}^\star$. Further, $\mathcal{C}_D(v_i) \neq 0$ and $\mathcal{C}_D(v_j) \neq 0$. Thus there exists at least one directed path, say $\hat{\omega}'$, from v_i to $t_{\mathcal{F}}$ that does not pass through any node in $\hat{\mathcal{S}}^\star \setminus \{v_i\}$. Similarly, there exists at least one directed path, say $\hat{\omega}''$, from v_j to $t_{\mathcal{F}}$ that does not pass through any node in $\hat{\mathcal{S}}^\star \setminus \{v_j\}$. As per the given defender strategy p_D the only trapping node in $\hat{\omega}', \hat{\omega}''$ is v_i, v_j, respectively. Thus there exists a strategy p_A such that all paths in $\Omega_{\mathcal{D}}$ with nonzero probability under p_A pass through exactly one node in $\hat{\mathcal{S}}^\star$ and $U_A(p_D, p_A) > U_A(p_D, p'_A)$ (since p_{FP} remains same and p_T is higher and p_R is lower under p'_A when compared to the values under strategy p_A). This contradicts the assumption that p'_A is a best response and completes the proof. □

Recall that $\Omega_{\mathcal{D}}$ is the set of all source-to-sink paths in the flow-network $\mathcal{F}$ (Definition IV.1). Note that, any path in $\mathcal{F}$ that does not belong to $\Omega_{\mathcal{D}}$ is not a valid attack, as the attacker cannot reach a destination node. Every attack path in $\Omega_{\mathcal{D}}$, depending on the defender strategy and the transition structure of the game, induces a set of paths in the state space graph of the APT vs. DIFT game. Let Ω be the set of all paths induced by $\Omega_{\mathcal{D}}$. In other words, Ω is the set of all possible state transition paths in the state space graph corresponding to all attack paths in $\mathcal{F}$.

Definition V.9. *Let the set of paths induced in the state space $\mathbf{S}$ by the attack paths $\Omega_{\mathcal{D}}$ in $\mathcal{F}$ be denoted as Ω. Then the probability of selecting a path $\omega \in \Omega$, denoted by $\pi(\omega)$, is $\pi(\omega) = \pi_A(\omega)\pi_B(\omega)$, where $\pi_A(\omega)$ is the probability with which an adversary chooses ω (product of the adversary transition probabilities along path ω) and $\pi_B(\omega)$ is the probability of benign flows in ω under distribution π_B.*

The following result proves that, under certain conditions, for a given adversary strategy the best response of the defender is to select one node in every attack path as a trapping node.

Lemma V.10. *Let $\Omega_{\mathcal{D}}$ be the set of attack paths in $\mathcal{F}$, Ω be the set of paths in $\mathbf{S}$ induced by $\Omega_{\mathcal{D}}$, and p_A be a given strategy of $\mathcal{P}_A$. For $\omega \in \Omega$, let $\omega(A)$ denote the set of nodes in ω corresponding to the adversarial (malicious) flow and $\omega(B)$ denote the set of nodes in ω corresponding to the benign flows. Also, let $p(\omega)$ denote the probability of detecting the adversary along path ω, i.e., $p(\omega) = \Big[1 - \prod_{v_i \in \omega(A)}(1 - p_D(v_i))\Big](1 - FN)$. Also, let $f(\omega)$ denote the probability of trapping a benign flow (false-positive), i.e., $f(\omega) = \Big[1 - \prod_{v_i \in \omega(B)}(1 - p_D(v_i))\Big] FP$. If the defender's strategy satisfies the following two conditions:*

(a) $p(\omega) = p(\omega')$, for all $\omega, \omega' \in \Omega$ and
(b) $f(\omega) = f(\omega')$, for all $\omega, \omega' \in \Omega$,

then the best response of the defender, $BR(p_A)$, is to select with nonzero probability exactly one node in every $\hat{\omega} \in \Omega_{\mathcal{D}}$ as a trapping node.

Proof. Let $\pi(\omega)$ denote the probability of a path $\omega \in \Omega$ under the given strategy p_A and benign distribution π_B. For a path ω in the state space $\mathbf{S}$, $\omega(A)$ denotes the set of nodes in ω corresponding to the adversarial (malicious) flow and $\omega(B)$ denotes the set of nodes in ω corresponding to the benign flows. Let $p(\omega)$ denote the probability of detecting the adversary along path ω, i.e., $p(\omega) = \Big[1 - \prod_{v_i \in \omega(A)}(1 - p_D(v_i))\Big](1 - FN)$. Also, let $f(\omega)$ denote the probability of trapping a benign flow, i.e., false-positive. Then $f(\omega) = \Big[1 - \prod_{v_i \in \omega(B)}(1 - p_D(v_i))\Big] FP$. The defender's payoff

$$\begin{aligned} U_D(p_D, p_A) &= \sum_{\omega \in \Omega} \pi(\omega) \Big[p(\omega)\,\alpha_D + (1 - p(\omega) + f(\omega))\beta_D \Big] \\ &+ \sum_{\omega \in \Omega} \Big(\sum_{v_i \in \omega} p_D(v_i) \mathcal{C}_D(v_i) \Big) \end{aligned}$$

Given $p(\omega)$'s and $f(\omega)$'s are equal for all $\omega \in \Omega$. Thus

$$\begin{aligned} U_D(p_D, p_A) &= \Big[p(\omega)\,(\alpha_D - \beta_D) + (1 + f(\omega))\beta_D \Big] \Big(\sum_{\omega \in \Omega} \pi(\omega) \Big) \\ &+ \sum_{\omega \in \Omega} \Big(\sum_{v_i \in \omega} p_D(v_i) \mathcal{C}_D(v_i) \Big) \\ &= \Big[p(\omega)\,\alpha_D + (1 - p(\omega) + f(\omega))\beta_D \Big] + \sum_{\omega \in \Omega} \Big(\sum_{v_i \in \omega} p_D(v_i) \mathcal{C}_D(v_i) \Big) \end{aligned} \tag{15}$$

Eq. (15) holds as $\sum_{\omega \in \Omega} \pi(\omega) = 1$. Consider a defender strategy p_D in which exactly one node in every $\hat{\omega} \in \Omega_{\mathcal{D}}$ is chosen as the trapping node. Assume that the defender strategy is modified to p'_D such that more than one node in some path are chosen as trapping nodes. This variation updates the probabilities of nodes in a set of paths in Ω. Note that, due to the constraints on $p(\omega)$ and $f(\omega)$ (conditions (a) and (b)), the defender's probabilities (strategy) at two nodes in a path are dependent. Hence for all paths $\omega \in \Omega$ whose probabilities are modified in p'_D, $\sum_{v_i \in \omega} p'_D(v_i)\mathcal{C}_D(v_i) < \sum_{v_i \in \omega} p_D(v_i)\mathcal{C}_D(v_i)$. This holds as the probability in the single node case is less than the sum of the probabilities of more than one node case as the events are dependent and the sum of the values of $\mathcal{C}_D(\cdot) < 0$ are also minimum (since min-cut). This implies

$$\sum_{\omega \in \Omega} \Big(\sum_{v_i \in V_{\mathcal{G}}} p'_D(v_i) \mathcal{C}_D(v_i) \Big) < \sum_{\omega \in \Omega} \Big(\sum_{v_i \in V_{\mathcal{G}}} p_D(v_i) \mathcal{C}_D(v_i) \Big). \tag{16}$$

From Eq. (15) and by conditions (a) and (b), we get $U_D(p'_D, p_A) < U_D(p_D, p_A)$. Therefore, p'_D is not a best response

for the defender and no best response of the defender has more than one node chosen as trapping node, if $p(\omega)$'s and $f(\omega)$'s are equal for all $\omega \in \Omega$. □

The result below proves that if Lemma V.10 holds, then for a given adversary strategy the best response of the defender is to select the min-cut nodes of $\overline{\mathcal{F}}$ as trapping nodes.

Lemma V.11. *Let $\Omega_{\mathcal{D}}$ be the set of attack paths in $\mathcal{F}$. Assume that the defender's strategy satisfies conditions (a) and (b) in Lemma V.10 for a given adversary strategy p_A. Then the best response of the defender, $BR(p_A)$, is to select with nonzero probability a min-cut node set of the flow-network $\overline{\mathcal{F}}$ as the trapping nodes.*

Proof. Under conditions (a) and (b) in Lemma V.10, the best response of the defender is to select one node in every attack path as trapping node. Note that all attack paths in $\mathcal{F}$ pass through some node in the min-cut node set $\hat{\mathcal{S}}^\star$. By selecting the nodes in $\hat{\mathcal{S}}^\star$ as trapping nodes, all possible attack paths have some nonzero probability of getting detected. We prove the result using a contradiction argument. Suppose that the best response of the defender is not to select the nodes in $\hat{\mathcal{S}}^\star$ as trapping nodes. Then, there exists a subset of nodes $\hat{\mathcal{S}} \subset V_{\mathcal{G}}$ such that $\sum_{v_i \in \hat{\mathcal{S}}} \mathcal{C}_D(v_i) < \sum_{v_j \in \hat{\mathcal{S}}^\star} \mathcal{C}_D(v_j)$. Further, all possible attack paths in $\mathcal{F}$ pass through some node in $\hat{\mathcal{S}}$. Then, $\hat{\mathcal{S}}$ is a (source-sink)-cut-set and let $\kappa(\hat{\mathcal{S}}) := \{(v_i, v_i') : v_i \in \hat{\mathcal{S}}\}$. Then, $\kappa(\hat{\mathcal{S}})$ is a cut set and $c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}})) < c_{\mathcal{F}}(\kappa(\hat{\mathcal{S}}^\star))$. This contradicts the fact that $\kappa(\hat{\mathcal{S}}^\star)$ is an optimal solution to the (source-sink)-min-cut problem. Hence under Lemma V.10 the best response of the defender is to select the nodes in $\hat{\mathcal{S}}^\star$ as trapping nodes. □

Using Lemma V.8, Lemma V.10, and Lemma V.11, we present below the proof of Theorem V.7.

Proof of Theorem V.7: Consider a defender's strategy that selects the min-cut nodes as the trapping nodes. Then, by Lemma V.8 the best response of the adversary is to select transitions in such a way that all attack paths with nonzero probability pass through exactly one node in the min-cut. Further Let $\Omega_{\mathcal{D}}$ be the set of attack paths in $\mathcal{F}$ and Ω be the set of paths in $\mathbf{S}$ induced by $\Omega_{\mathcal{D}}$. Lemma V.11 concludes that the best response of the defender is to select the min-cut nodes of $\mathcal{F}$ as trapping nodes, provided the probability of detecting the adversary and the probability of trapping a benign flow are equal for all $\omega \in \Omega$. This implies that, if NE strategy pair satisfy the conditions that $p(\omega)$'s and $f(\omega)$'s are equal for all $\omega \in \Omega$, then the defender's strategy at NE is to select the min-cut nodes as the trapping nodes and the adversary's strategy is to choose an attack path such that it passes through exactly one node in the min-cut node set. By Proposition V.5, there exists an NE for $\mathbf{G}$. Consequently, if $p(\omega)$'s and $f(\omega)$'s are equal at NE, the proof follows.

Let (p_A, p_D) be an NE of $\mathbf{G}$. Consider a unilateral deviation in the strategy of the adversary. Let $\pi(\omega)$'s for $\omega \in \Omega$ are modified due to change in transition probabilities of the adversary such that the updated probabilities of the attack paths are $\pi(\omega_i) + \varepsilon_i$, for $i = 1, \ldots, |\Omega|$. Here, ε_i's can take positive values, negative values or zero such that $\sum_{i=1}^{|\Omega|} \varepsilon_i = 0$. Consider two arbitrary paths, say $\omega_1 \in \Omega$ and $\omega_2 \in \Omega$, such that a unilateral change in the adversary strategy changes $\pi(\omega_1)$ and $\pi(\omega_2)$ and the probabilities of the other paths remain unchanged. Without loss of generality, assume that $\pi(\omega_1)$ increases by ε while $\pi(\omega_2)$ decreases by ε and all other $\pi(\omega)$'s remain the same. As (p_D, p_A) is an NE, $(\pi(\omega_1)+\varepsilon)\Big(p(\omega_1)(\alpha_A-\beta_A)+\beta_A+f(\omega_1)\beta_A\Big)+(\pi(\omega_2)-\varepsilon)\Big(p(\omega_2)(\alpha_A-\beta_A)+\beta_A+f(\omega_2)\beta_A\Big) \leqslant \pi(\omega_1)\Big(p(\omega_1)(\alpha_A-\beta_A)+\beta_A+f(\omega_1)\beta_A\Big)+\pi(\omega_2)\Big(p(\omega_2)(\alpha_A-\beta_A)+\beta_A+f(\omega_2)\beta_A\Big)$. Thus

$$(p(\omega_1)-p(\omega_2))(\alpha_A-\beta_A)+(f(\omega_1)-f(\omega_2))\beta_A \leqslant 0. \quad (17)$$

Now consider another unilateral deviation of adversary strategy such that $\pi(\omega_1)$ decreases by ε while $\pi(\omega_2)$ increases by ε and all other $\pi(\omega)$'s remain the same. This gives

$$(p(\omega_1)-p(\omega_2))(\alpha_A-\beta_A)+(f(\omega_1)-f(\omega_2))\beta_A \geqslant 0. \quad (18)$$

Eqs. (17) and (18) imply

$$(p(\omega_1)-p(\omega_2))(\alpha_A-\beta_A)+(f(\omega_1)-f(\omega_2))\beta_A = 0. \quad (19)$$

Note that $(\alpha^A-\beta^A) < 0$ and $\beta_A > 0$. Therefore, there are three possible cases where Eq. (19) holds.

(i) $(p(\omega_1)-p(\omega_2)) > 0$ and $(f(\omega_1)-f(\omega_2)) > 0$,
(ii) $(p(\omega_1)-p(\omega_2)) < 0$ and $(f(\omega_1)-f(\omega_2)) < 0$, and
(iii) $(p(\omega_1)-p(\omega_2)) = 0$ and $(f(\omega_1)-f(\omega_2)) = 0$.

Consider case (i). For a path ω in the state space let $\omega(A)$ denote the set of nodes in ω corresponding to the adversarial flow and $\omega(B)$ denote the set of nodes in ω corresponding to the benign flows. Rewriting$(p(\omega_1)-p(\omega_2)) > 0$, we get

$$\Big(\Big[1-\prod_{v_i\in\omega_1(A)}(1-p_D(v_i))\Big]-\Big[1-\prod_{v_j\in\omega_2(A)}(1-p_D(v_j))\Big]\Big)(1-FN) > 0 \quad (20)$$

Similarly rewriting$(f(\omega_1)-f(\omega_2)) > 0$, we get

$$\Big(\Big[1-\prod_{v_i\in\omega_1(B)}(1-p_D(v_i))\Big]-\Big[1-\prod_{v_j\in\omega_2(B)}(1-p_D(v_j))\Big]\Big)FP > 0 \quad (21)$$

As $0 < FP < 1$ and $0 < FN < 1$, Eqs. (20) and (21) imply

$$\Big[\prod_{v_i\in\omega_2(A)}(1-p_D(v_i))-\prod_{v_j\in\omega_1(A)}(1-p_D(v_i))\Big] > 0 \quad (22)$$

$$\Big[\prod_{v_i\in\omega_2(B)}(1-p_D(v_i))-\prod_{v_j\in\omega_1(B)}(1-p_D(v_j))\Big] > 0 \quad (23)$$

Eqs. (22) and (23) imply

$$\prod_{v_i\in\omega_2(A)}(1-p_D(v_i)) > \prod_{v_j\in\omega_1(A)}(1-p_D(v_i)) \text{ and } \quad (24)$$

$$\prod_{v_i\in\omega_2(B)}(1-p_D(v_i)) > \prod_{v_j\in\omega_1(B)}(1-p_D(v_j)) \quad (25)$$

Note that at every state in $\mathbf{s} \in \mathbf{S}$ with $\mathbf{s} = \{v_{i_1}, \ldots, v_{i_k}\}$ $0 \leqslant \sum_{v_{i_k}\in\mathbf{s}} p_D(v_{i_k}) \leqslant 1$. Thus Eqs. (24) and (25) cannot be together satisfied. Thus case (i) does not hold at an NE. Following the similar arguments one can show that case (ii) also does not hold at an NE. This implies $(p(\omega_1)-p(\omega_2)) = 0$ and $(f(\omega_1)-f(\omega_2)) = 0$.

Since ω_1 and ω_2 are arbitrary, one can show that for the

general case

$$\sum_{i=1}^{|\Omega|} \varepsilon_i p(\omega_i) = 0 \text{ and } \sum_{i=1}^{|\Omega|} \varepsilon_i f(\omega_i) = 0. \tag{26}$$

Eq. (26) should hold for all possible values of ε_i's satisfying $\sum_{i=1}^{|\Omega|} \varepsilon_i = 0$. This gives $p(\omega_i) = p(\omega_j)$ and $f(\omega_i) = f(\omega_j)$, for all $i, j \in \{1, \ldots, |\Omega|\}$, at NE. This completes the proof. □

Theorem V.7 concludes that the set of all solutions to the min-cut problem characterizes the set of NE of game **G**. Two key properties of an equilibrium strategy pair is that defender chooses the min-cut nodes as trapping nodes and indeed with equal probability, which is proved in Lemma V.12.

Lemma V.12. *Consider the APT vs. DIFT game* **G** *and let* (p_A, p_D) *be an NE of* **G**. *Then under* p_D *all the min-cut nodes of the flow-network have equal probability of selecting as a trapping node.*

Proof. At NE, all the paths in the state space have equal probability of getting detected, i.e., $p(\omega)$'s are same for all $\omega \in \Omega$ (by Theorem V.7). For a path $\omega \in \Omega$ with $\omega(A)$ denoting the set of nodes corresponding to the adversarial flow, we know

$$p(\omega) = \Big[1 - \prod_{v_i \in \omega(A)} (1 - p_D(v_i))\Big](1 - FN), \text{ for all } \omega \in \Omega. \tag{27}$$

Also, there is exactly one node corresponding to an attack path that has nonzero probability of selecting as a trapping node (Theorem V.7). Hence in set $\omega(A)$ exactly one node, say $v_i \in \omega(A)$, has nonzero value of p_D. Eq. (27) hence implies that at NE all the min-cut nodes of the flow-network have equal probability of selecting as a trapping node. □

As a consequence of Theorem V.7 and Lemma V.12, the following corollary holds.

Corollary V.13. *Let* $\hat{\mathcal{S}}^\star$ *be a set of min-cut nodes of the flow-network* $\overline{\mathcal{F}}$. *Then, at an NE of the game* **G**, *the defender's strategy is to choose all the nodes in* $\hat{\mathcal{S}}^\star$ *as trapping nodes with equal probability and the adversary's strategy is to choose transitions in such a way that each attack path passes through exactly one node in* $\hat{\mathcal{S}}^\star$.

Proof. By Theorem V.7 and Lemma V.12 the defender's strategy at NE is to select min-cut nodes as trapping nodes with equal probability. We know adversary's payoff

$$U_A(p_D, p_A) \quad = \quad \sum_{\omega \in \Omega} \pi(\omega) \Big[p(\omega)\,\alpha_A + (1 - p(\omega) + f(\omega))\beta_A\Big],$$

and at NE $p(\omega)$'s and $f(\omega)$'s are equal for all $\omega \in \Omega$. Thus the adversary's optimal strategy is to select any path (or set of paths if mixed) that passes through exactly one min-cut node. □

Using Theorem V.7 and Corollary V.13 we present below an approach to compute an equilibrium strategy pair of **G**. Firstly, we solve the node version of the (source-sink)-min-cut problem on $\mathcal{F}$. Let an optimal solution be $\kappa(\hat{\mathcal{S}}^\star)$ and the corresponding vertex set be $\hat{\mathcal{S}}^\star = \{\tilde{v}_1, \ldots, \tilde{v}_r\}$, where $\hat{\mathcal{S}}^\star := \{v_i : (v_i, v_i') \in \kappa(\hat{\mathcal{S}}^\star)\}$. By Theorem V.7, at NE the defender only selects the nodes in $\hat{\mathcal{S}}^\star$ as trapping nodes, with equal probability, and the adversary chooses transitions such that each attack path passes through only one node in $\hat{\mathcal{S}}^\star$. Therefore, the attack paths chosen by the adversary is characterized by the nodes in $\hat{\mathcal{S}}^\star$. The set of paths in the state space **S** can also be characterized by the nodes in $\hat{\mathcal{S}}^\star$. In other words, the set of paths in the set Ω can be grouped such that each group corresponds to a set of paths in Ω in which attack path (not necessarily benign flows) passes through exactly one node in $\hat{\mathcal{S}}^\star$, i.e., at least one among the W tagged flows passes through exactly one min-cut node. We denote the set of paths in Ω that correspond to node $\tilde{v}_i \in \hat{\mathcal{S}}^\star$ as $\Omega(\tilde{v}_i)$. By Corollary V.13 any distribution over paths in $\Omega(\tilde{v}_i)$, for $i \in \{1, \ldots, r\}$ is an optimal strategy for $\mathcal{P}_A$.

Remark V.14. *At NE, all attack paths in* $\mathcal{F}$ *with nonzero probability pass through exactly one node in the min-cut node set* $\hat{\mathcal{S}}^\star = \{\tilde{v}_1, \ldots, \tilde{v}_r\}$. *Further, the probabilities of selecting them as trapping nodes are also equal (since* $p(\omega)$*'s are equal). Thus, without loss of generality, one can characterize the action space of the adversary as the set of attack paths in* $\mathcal{F}$ *that pass through* $\hat{\mathcal{S}}^\star$, *disjoint with respect to nodes in* $\hat{\mathcal{S}}^\star$, *and the adversary strategizes over this set of paths.*

In the theorem below, we derive a closed-form expression for the optimal defender strategy for the game **G**.

Theorem V.15. *Consider the APT vs. DIFT game* **G**. *At NE the defender selects all the min-cut nodes as trapping nodes with probability* $\dfrac{1}{\min\{W, r\}}$, *where* W, r *respectively denote the number of tagged flows at a time in the system and the cardinality of the min-cut nodes.*

Proof. Consider a non-terminal state $\mathbf{s} \in \mathbf{S}$, where $\mathbf{s} = \{v_{i_1}, \ldots, v_{i_W}\}$. The action set of $\mathcal{P}_D$ at $\mathbf{s}$ is $\mathcal{A}_D(\mathbf{s}) = \{0\} \cup \{v_{i_1}, \ldots, v_{i_W}\}$, for$\{v_{i_1}, \ldots, v_{i_W}\} \subset V_{\mathcal{G}}$. Any defender strategy must satisfy for all $\mathbf{s} \in \mathbf{S}$, $p_D(0) + \sum_{j=1}^{W} p_D(v_{i_j}) = 1$ if $v_{i_j} \neq \phi$. By Corollary V.13, at NE, $\mathcal{P}_D$ only selects min-cut nodes as trapping nodes. Let $\hat{\mathcal{S}}^\star = \{\tilde{v}_1, \ldots, \tilde{v}_r\}$ be the solution obtained for the min-cut problem. Thus the constraint on p_D (that it should add upto 1 at all states) boils down to the following. For all states $\mathbf{s} = \{v_{i_1}, \ldots, v_{i_W}\} \subseteq \{\tilde{v}_1, \ldots, \tilde{v}_r\}$,

$$p_D(0) + \sum_{\substack{v_{i_j} : v_{i_j} \in \{\tilde{v}_1, \ldots, \tilde{v}_r\} \\ j \in \{1, \ldots, W\}}} p_D(v_{i_j}) = 1, \tag{28}$$

Moreover, we know all min-cut nodes are chosen as trapping nodes with equal probability (Corollary V.13), i.e., $p_D(\tilde{v}_1) = \ldots = p_D(\tilde{v}_r) := \theta$. For all states $\mathbf{s} = \{v_{i_1}, \ldots, v_{i_W}\} \subseteq \{\tilde{v}_1, \ldots, \tilde{v}_r\}$

$$p_D(0) + \theta \Big(\sum_{\substack{v_{i_j} : v_{i_j} \in \{\tilde{v}_1, \ldots, \tilde{v}_r\} \\ j \in \{1, \ldots, W\}}} 1 \Big) = 1, \tag{29}$$

For all states $\mathbf{s} \in \mathbf{S}$ with $\mathbf{s} = \{v_{i_1}, \ldots, v_{i_W}\} \subseteq \{\tilde{v}_1, \ldots, \tilde{v}_r\}$,

$$\sum_{\substack{v_{i_j} : v_{i_j} \in \{\tilde{v}_1, \ldots, \tilde{v}_r\} \\ j \in \{1, \ldots, W\}}} 1 \leqslant \min\{W, r\}. \tag{30}$$

Eq. (30) must hold for all states and hence the maximum value

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2024.3403675

that θ can take is when $p_D(0) = 0$ and $\sum_{v_{i_j}:v_{i_j}\in\{\tilde{v}_1,\ldots,\tilde{v}_r\}, j\in\{1,\ldots,W\}} 1 = \min\{W,r\}$ at a state $\mathbf{s}\in\mathbf{S}$. Thus $\theta = \frac{1}{\min\{W,r\}}$.

As the defender's and adversary's action sets are characterized by the min-cut nodes, the defender's payoff is

$$\begin{aligned} U_D(p_D,p_A) &= \sum_{i=1}^{r}\sum_{\omega\in\Omega(\tilde{v}_i)}\pi(\omega)\Big[p(\omega)\,\alpha_D + (1-p(\omega)+f(\omega))\beta_D\Big] \\ &+ \sum_{i=1}^{r}\sum_{\omega\in\Omega(\tilde{v}_i)}\Big(\sum_{\tilde{v}_i\in\omega} p_D(\tilde{v}_i)\mathcal{C}_D(\tilde{v}_i)\Big) \\ &= \sum_{i=1}^{r}\sum_{\omega\in\Omega(\tilde{v}_i)}\pi(\omega)\Big[\theta(1-FN)\,(\alpha_D-\beta_D)+\beta_D+f(\omega)\beta_D\Big] \\ &+ \sum_{i=1}^{r}\sum_{\omega\in\Omega(\tilde{v}_i)}\Big(\sum_{\tilde{v}_i\in\omega}\theta\mathcal{C}_D(\tilde{v}_i)\Big) \end{aligned} \tag{31}$$

Eq. (31) holds from Lemma V.12 by substituting $p(\omega) = [1-\prod_{\tilde{v}_i\in\omega(A)}(1-p_D(\tilde{v}_i))](1-FN) = (1-(1-\theta))(1-FN) = \theta(1-FN)$. Here $0\leqslant\theta(1-FN)\leqslant 1$ and $0\leqslant f(\omega)\leqslant 1$. Also, $(\alpha_D-\beta_D) >> \beta_D$ with $(\alpha_D-\beta_D) >> 1$ and $\beta_D < 0$. Also note that any distribution, $\pi(\omega)$, over paths in $\Omega(\tilde{v}_i)$, for $i\in\{1,\ldots,r\}$, is optimal for adversary (Corollary V.13). Thus Eq. (31) is maximum for the maximum possible θ. Hence at NE the defender selects a set of min-cut nodes as trapping nodes with probability $1/\min\{W,r\}$. □

In this work, we provide a characterization for the equilibrium of the DIFT vs. APT game using the min-cut analysis. Min-cut of a flow-network need not necessarily be unique. For an application whose IFG results in a flow-network with multiple min-cut sets, equilibrium of the DIFT vs. APT game will be nonunique. The game considers a strategic adversary and the defense strategy computed as equilibrium solution to the game provides a selection of the trapping nodes that maximizes the probability of detection while minimizing the cost of security analysis. During a real attack scenario, the adversary may choose an attack path which is different from the equilibrium of the game. We note that, by definition of Nash equilibrium, such a strategy will not improve the payoff of the adversary. Moreover, for a defender that is unaware of the actions of the adversary, implementing a solution to the game as the trapping nodes maximizes the probability of detection.

VI. Numerical Study

We validate our theoretical results using real-world attack data obtained using RAIN [1] for a data exfiltration attack. We use system log data collected during DARPA Transparent computing (DARPA-TC) red team engagement that evaluated the RAIN log recording system. A brief description of the dataset used and the steps involved in the construction of the IFG for that attack is given below.

The goal of the attack is to exfiltrate sensitive information from the system. The attack uses stolen credentials to exfiltrate sensitive files from the targeted machine. The attack exfiltrates sensitive system files using *scp*. The resulting IFG of the data consists of 299 nodes and 6398 edges. Figure 4 shows a portion of the IFG, obtained after performing a pruning technique, that captures all possible attack paths in the IFG. The details of the type of the nodes in the IFG in Figure 4 is presented in Table I. Entry points of the attack are identified as */bin/bash* and */usr/sbin/sshd* (nodes 1 and 5 in Figure 4). Destination node of the attack is */etc/passwd* (node 3 in Figure 4).

Table I: Description of the nodes of the IFG for the data exfiltration attack

Node ID	Node Name	Node Type
1	/bin/bash	Process
2	/usr/bin/sudo	Process
3	/etc/passwd	Files
4	Unknown	IPC Object
5	/usr/sbin/sshd	Process
6	/etc/group	File
7	/usr/sbin/console-kit-daemon	Process
8	Unknown	IPC Object
9	/usr/sbin/avahi-daemon	Process
10	Unknown	IPC Object
11	/run/ConsoleKit/database~	File
12	/usr/lib/policykit-1/polkitd	Process
13	/bin/run-parts	File
14	/run/motd.new	File
15	/home/theia/secrets.tar.gz	File
16	/bin/dash	File
17	/usr/bin/apt-config	File

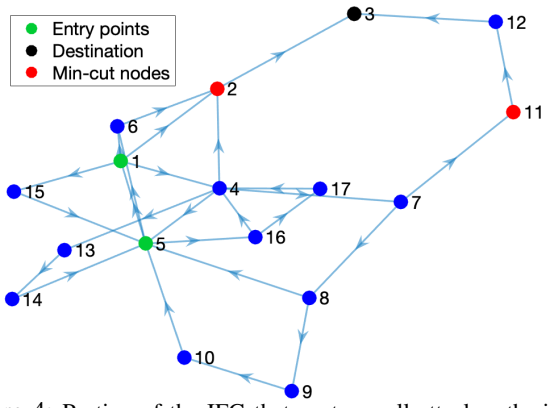
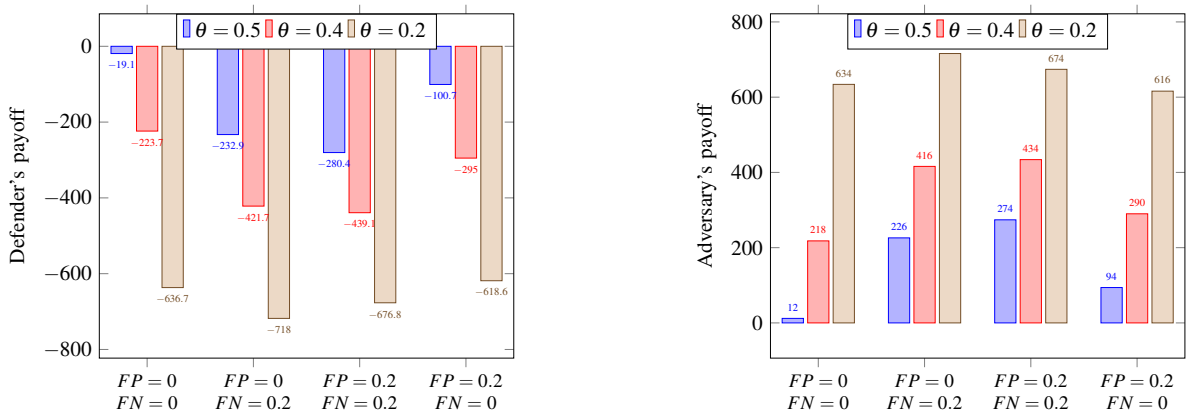


Figure 4: Portion of the IFG that captures all attack paths in IFG.

First we solve the min-cut problem on the flow-network constructed from the IFG of the data. Resulting min-cut nodes, */usr/bin/sudo* and */run/consolekit/data*, are indicated in red color in Figure 4. The min-cut nodes are chosen as trapping nodes with probability $\theta = 1/\min\{W,r\}$. W and r denote the number of tagged flows at a time in the system and the cardinality of the obtained solution to the min-cut problem, respectively. Here $W = 3$ and $r = 2$. Thus $\theta = 0.5$.

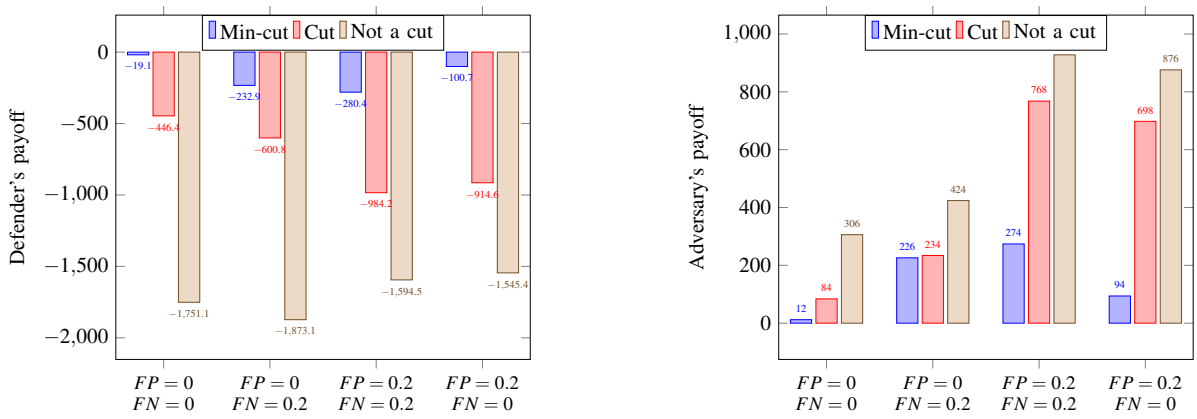
Then we simulate an attack in the IFG and perform security analysis to evaluate the performance of the DIFT model. We conduct two experiments which are detailed below. Each payoff value represented in Figure 5 and Figure 6 are obtained after averaging over 1000 trials. We have identified 13 distinct

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2024.3403675



(a) (b)

Figure 5: The parameters chosen are: $\alpha_A = -1000$, $\beta_A = 1000$, $\alpha_D = 1000$, $\beta_A = -1000$. The resource cost for the nodes are chosen such that $C_D(v_i)$ is proportional to the total number of information flows at node v_i for the whole logging period, where $i \in \{1,\ldots,N\}$. Figures 5 (a) and 5 (b) shows the payoffs of the defender and adversary, respectively, for different values of false-positives (FP) and false-negatives (FN) when min-cut nodes are chosen as trapping nodes. The probability of selecting a min-cut nodes as trapping node is varied from $\theta = 1/\min\{W,r\} = 1/\min\{3,2\} = 0.5$, $\theta = 0.4$ and $\theta = 0.2$ for all the experiments. Also, each case is averaged over 1000 runs.



(a) (b)

Figure 6: The parameters chosen are: $\alpha_A = -1000$, $\beta_A = 1000$, $\alpha_D = 1000$, $\beta_A = -1000$. The resource cost for the nodes are chosen such that $C_D(v_i)$ is proportional to the total number of information flows at node v_i for the whole logging period, where $i \in \{1,\ldots,N\}$. Figures 6 (a) and 6 (b) shows the payoffs of the defender and adversary, respectively, for different values of false-positives (FP) and false-negatives (FN) when (i) min-cut nodes are chosen as trapping nodes, (ii) cut nodes (not min-cut), and (iii) nodes that are not a cut. Also, each case is averaged over 1000 runs.

attack paths related to the adversary in the underlying IFG and uniformly picked one attack path in each trial. Each of these attack paths consist only one min-cut node (Theorem V.7). Further, we set $\alpha_A = -1000$, $\beta_A = 1000$, $\alpha_D = 1000$, and $\beta_A = -1000$. $C_D(v_i)$ values are set to be proportional to the number of information flows passing through each node, v_i for $i \in \{1,\ldots,N\}$, in the IFG through out the logging period.

A. Case Study 1

In this experiment setup, we vary the probability of selecting min-cut nodes as trapping nodes. Recall that any probability greater than θ is not a valid defender's strategy (Theorem V.15). Hence for comparing the performance we select probabilities $\theta \leqslant 1/\min\{W,r\} = 0.5$. Note that at NE $\theta = 0.5$. Figures 5 (a) and 5 (b) plot the payoff values for both players averaged over 1000 trials for different values of false-positives and false-negatives. Figure 5 (a) shows that defender performs better when following NE strategy, i.e., $\theta = 0.5$, which validates the theoretical analysis.

B. Case Study 2

In this case study we compare the performance of the DIFT by varying the locations of trapping nodes. In provenance-based analysis, various heuristics have been proposed for selecting the trapping nodes [27], [28]. In [27], a heuristic that assigns weights to the nodes of the IFG is proposed. In this case study, we compare the performance of the defense strategy obtained as solution to the DIFT vs. APT game with two cases. For the first case, we assign nonzero weights to the cut nodes of the IFG, since cut nodes are the smallest set of nodes through which all the flows pass through, and assign zero weights to all non-cut nodes. For the second case, we select the non-cut nodes as the trapping nodes, since all

information flows will pass through the non-cut nodes, by assigning nonzero weights to all non-cut nodes and assign zero weights to all cut nodes. Figures 6 (a) and 6 (b) show that the expected payoff of the defender takes higher values when the min-cut nodes are chosen as the trapping nodes. Also note that choosing a cut is better than selecting set of nodes that are not a cut.

C. Case Study 3

In this case study we compare the performance of the game theoretic approach with a heuristic rule-based method. For comparing the two approaches, we use the knowledge of the actual attack path as the ground truth. The attack path for the data exfiltration attack is $1 \to 15 \to 5 \to 16 \to 17 \to 4 \to 2 \to 3$ in the IFG given in Figure 4. The heuristic rules used for identifying the trapping nodes are presented in Table III. Using these rules, for the data exfiltration attack, we identified the set of nodes $\{2, 6, 12, 13, 16, 17\}$ as the trapping nodes. Solution to the DIFT vs. APT game returns the min-cut node set $\{2, 11\}$ as the trapping nodes. In this case study, we set $W = 1$ and consequently $\theta = 1$. The parameters of the game are set as $\alpha_D = 1$, $\beta_D = -1$, $FN = 0.2$, and $FP = 0$.

Resource cost for performing security analysis at a node v_i is given by

$$\mathcal{C}_D(v_i) = c\ q(v_i), \tag{32}$$

where c is the cost parameter for performing security analysis and $q(v_i)$ is the fraction of information flows passing through node v_i. The resource cost for conducting security analysis is more for a node with higher fraction of flows, i.e., a busy node. Figure 7 presents a plot showing the variation of the DIFT's payoff with respect to c. For the data exfiltration attack considered and for the selection of the trapping nodes $\{2, 6, 12, 13, 16, 17\}$, heuristic rule-based method will conduct security analysis on the flow at nodes $2, 16$, and 17. On the other hand, the game theoretic modeling will conduct security analysis on the flow at node 2. Table II presents the fraction of the information flows passing through nodes $2, 16$, and 17. Figure 7 shows that the hueristic rule-based method performs better for lower values of c. This is expected since when cost for conducting security analysis is negligible it is better to analyze the flow at multiple locations, as it increases the probability of detection. As the cost parameter increases, the game theoretic solution outperforms the heuristic. The performance of the heuristic decreases drastically for higher values of c as heuristic conducts security analysis at node 16 which is a busier node and hence incurs high resource cost. The game, on the other hand, considers the resource cost while computing the min-cut and hence chooses node 2 for conducting the security analysis. This validates that the game theoretic approach provides a trade-off between the effectiveness of detection and the cost of detection.

Table II: Fraction of flows traversing nodes in IFG

Node v_i	2	16	17
$q(v_i)$	0.005705	0.167162	0.016177

Table III: Heuristic rules used for identifying trapping nodes in Case study 3

System Call	Performed on File	Description
read	/dev/mem	Potentially anomalous actions on sensitive files/folder
read	/etc/passwd	
read	/etc/shadow	
read	*.ssh/id_rsa	Potentially anomalous actions on sensitive SSH files
read	*.default/key3.db	Potentially anomalous actions on sensitive SSH files
read	*.default/logins.json	
read	*.default/signons.sqlite	
write	/dev/mem	
write	/etc/*	
write	/usr/local/*	
write	/usr/local/bin/*	
write	/var/spool/cron/crontabs/root	
write	/bin/*	
write	/boot/*	
write	/var/*	
write	/usr/bin/*	
write	/dev/*	
write	/etc/security/*	
write	/usr/spool/*	
write	/usr/etc/*	
write	/usr/kvm/*	
write	/usr/*	
write	/usr/lib/*	
write	*.default/key3.db	
write	*.default/logins.json	
write	*.default/signons.sqlite	
write	/etc/hosts	Important DNS information
write	*.default/extensions	Stores extensions for Firefox
open	*libpam*	Used for authentication
exec	bin/ip	Used for gathering information about network
exec	/bin/ifconfig	
exec	/bin/netstat	
exec	/bin/chmod	Changes permissions of files/folders
exec	/bin/chown	Changes ownership of files/folders

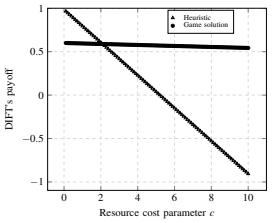


Figure 7: Plot shows variation of DIFT's payoff for (i) min-cut solution obtained as equilibrium of the game and (ii) heuristic rules-based method, with respect to c.

VII. Conclusion

This paper provided a game-theoretic model for the detection of Advanced Persistent Threats (APTs) using Dynamic Information Flow Tracking (DIFT). The interaction of APT and DIFT is modeled as a stochastic game in which the defender dynamically chooses locations (trapping nodes) to

perform security analysis and the adversary chooses transitions along attack paths in order to maximize the probability of reaching the desired target. Before performing security analysis, the defender is unable to distinguish whether a tagged information flow at a node in the information flow graph is malicious or not and this results in the information asymmetry of the players. Further, the defender is not accurate in its detection process, which results in the generation of false-positives and false-negatives. We modeled the strategic interactions between DIFT and APT as a nonzero-sum, turn-based stochastic game. The game captures the information asymmetry among the players along with false-positives and false-negatives of DIFT. Then we provided an approach to compute Nash equilibrium of the game, using a minimum capacity cut-set formulation. Finally, we implemented our algorithm on real-world data for a data exfiltration attack obtained using the Refinable Attack INvestigation (RAIN) system. Extending the game model to address a multi-attacker case with multiple malicious flows is a part of future work.

REFERENCES

[1] Y. Ji, S. Lee, E. Downing, W. Wang, M. Fazzini, T. Kim, A. Orso, and W. Lee, "RAIN: Refinable attack investigation with on-demand inter-process information flow tracking," *ACM SIGSAC Conference on Computer and Communications Security*, pp. 377–390, 2017.

[2] J. Newsome and D. Song, "Dynamic taint analysis: Automatic detection, analysis, and signature generation of exploit attacks on commodity software," *Network and Distributed Systems Security Symposium*, 2005.

[3] M. Dalton, H. Kannan, and C. Kozyrakis, "Real-world buffer overflow protection for userspace and kernelspace." *USENIX Security Symposium*, pp. 395–410, 2008.

[4] M. Dalton, C. Kozyrakis, and N. Zeldovich, "Nemesis: Preventing authentication & access control vulnerabilities in web applications," *USENIX Security Symposium*, pp. 267–282, 2009.

[5] L. S. Shapley, "Stochastic games," *Proceedings of the National Academy of Sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.

[6] K.-w. Lye and J. M. Wing, "Game strategies in network security," *International Journal of Information Security*, vol. 4, no. 1-2, pp. 71–86, 2005.

[7] R. Amir, "Stochastic games in economics and related fields: An overview," *Stochastic Games and Applications*, pp. 455–470, 2003.

[8] Q. Zhu and T. Başar, "Robust and resilient control design for cyber-physical systems with an application to power systems," *IEEE Decision and Control and European Control Conference (CDC-ECC)*, pp. 4066–4071, 2011.

[9] A. Jaśkiewicz and A. S. Nowak, "Non-zero-sum stochastic games," *Handbook of Dynamic Game Theory*, pp. 1–64, 2016.

[10] T. Alpcan and T. Başar, "An intrusion detection game with limited observations," *International Symposium on Dynamic Games and Applications*, vol. 26, 2006.

[11] Q. Zhu, H. Tembine, and T. Başar, "Network security configurations: A nonzero-sum stochastic game approach," *American Control Conference (ACC)*, pp. 1059–1064, 2010.

[12] K. C. Nguyen, T. Alpcan, and T. Başar, "Stochastic games for security in networks with interdependent nodes," *International Conference on Game Theory for Networks*, pp. 697–703, 2009.

[13] J. P. Hespanha and M. Prandini, "Nash equilibria in partial-information games on Markov chains," *IEEE Conference on Decision and Control (CDC)*, vol. 3, pp. 2102–2107, 2001.

[14] D. Sahabandu, B. Xiao, A. Clark, S. Lee, W. Lee, and R. Poovendran, "DIFT games: dynamic information flow tracking games for advanced persistent threats," *IEEE Conference on Decision and Control (CDC)*, pp. 1136–1143, 2018.

[15] S. Moothedath, D. Sahabandu, A. Clark, S. Lee, W. Lee, and R. Poovendran, "Multi-stage dynamic information flow tracking game," *Conference on Decision and Game Theory for Security*, vol. 11199, pp. 80–101, 2018.

[16] S. Moothedath, D. Sahabandu, J. Allen, A. Clark, L. Bushnell, W. Lee, and R. Poovendran, "A game-theoretic approach for dynamic information flow tracking to detect multi-stage advanced persistent threats," *IEEE Transactions on Automatic Control*, 2020.

[17] D. Sahabandu, S. Moothedath, J. Allen, A. Clark, L. Bushnell, W. Lee, and R. Poovendran, "Dynamic information flow tracking games for simultaneous detection of multiple attackers," *IEEE Conference on Decision and Control (CDC)*, pp. 567–574, 2019.

[18] D. Sahabandu, S. Moothedath, J. Allen, A. Clark, L. Bushnell, W. Lee, and R. Poovendran, "A game theoretic approach for dynamic information flow tracking with conditional branching," *American Control Conference (ACC)*, pp. 2289–2296, 2019.

[19] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[20] G. E. Suh, J. W. Lee, D. Zhang, and S. Devadas, "Secure program execution via dynamic information flow tracking," *ACM SIGPLAN Notices*, vol. 39, no. 11, pp. 85–96, 2004.

[21] J. Clause, W. Li, and A. Orso, "Dytan: a generic dynamic taint analysis framework," *International Symposium on Software Testing and Analysis*, pp. 196–206, 2007.

[22] M. N. Hossain, J. Wang, R. Sekar, and S. D. Stoller, "Dependence-preserving data compaction for scalable forensic analysis," *USENIX Security Symposium*, pp. 1723–1740, 2018.

[23] Linux Kernel 3.7.1. (2021, June 17). [Online]. Available: https://docs.huihoo.com/doxygen/linux/kernel/3.7/timekeeping_8c.html#a43b6023a02e25bd465f61768418271b1

[24] H. Yin, D. Song, M. Egele, C. Kruegel, and E. Kirda, "Panorama: Capturing system-wide information flow for malware detection and analysis," *ACM conference on Computer and communications security*, pp. 116–127, 2007.

[25] M. J. Sobel, "Noncooperative stochastic games," *The Annals of Mathematical Statistics*, vol. 42, no. 6, pp. 1930–1935, 1971.

[26] J. B. Orlin, "A faster strongly polynomial minimum cost flow algorithm," *Operations Research*, vol. 41, no. 2, pp. 338–350, 1993.

[27] S. M. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. Venkatakrishnan, "Holmes: Real-time APT detection through correlation of suspicious information flows," *IEEE Symposium on Security and Privacy*, pp. 1137–1152, 2019.

[28] T. Chen, L.-A. Tang, Y. Sun, Z. Chen, and K. Zhang, "Entity embedding-based anomaly detection for heterogeneous categorical events," *International Joint Conference on Artificial Intelligence*, pp. 1396–1403, 2016.

Shana Moothedath is a Postdoctoral Research Scholar in the Department of Electrical and Computer Engineering at the University of Washington - Seattle. She received her B.Tech. and M.Tech. degrees in Electrical and Electronics Engineering from the Kerala University, India, in 2011 and 2014 respectively, and Ph.D. degree in Electrical Engineering from Indian Institute of Technology Bombay, India, in 2018. Her research interests include network security analysis, structural analysis of large-scale control systems, and applications of systems theory to complex networks.

Dinuka Sahabandu is a Ph.D. candidate in the Department of Electrical and Computer Engineering at the University of Washington - Seattle. He received the B.S. degree and M.S. degree in Electrical Engineering from the Washington State University - Pullman in 2013 and 2016, respectively. His research interests include game theory for network security and control of multi-agent systems.

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2024.3403675

Joey Allen is a Ph.D. candidate in the College of Computing at Georgia Institute of Technology, where he is studying information security under Dr. Wenke Lee. He received the B.S. degree and M.S. degree in Computer Engineering from the University of Tennessee - Knoxville in 2014 and 2016, respectively. His research interests include forensic analysis, mobile security, and information tracking.

Andrew Clark (M'15) is an Assistant Professor in the Department of Electrical and Computer Engineering at Worcester Polytechnic Institute. He received the B.S. degree in Electrical Engineering and the M.S. degree in Mathematics from the University of Michigan - Ann Arbor in 2007 and 2008, respectively. He received the Ph.D. degree from the Network Security Lab (NSL), Department of Electrical and Computer Engineering, at the University of Washington - Seattle in 2014. He is author or co-author of the IEEE/IFIP William C. Carter award-winning paper (2010), the WiOpt Best Paper (2012), the WiOpt Student Best Paper (2014), and an IEEE GameSec Outstanding Paper (2018), and was a finalist for the IEEE CDC 2012 Best Student-Paper Award and IEEE/ACM ICCPS Best Paper (2016, 2018). He received the University of Washington Center for Information Assurance and Cybersecurity (CIAC) Distinguished Research Award (2012) and Distinguished Dissertation Award (2014), and an NSF CRII award (2016). He holds a patent in privacy-preserving constant-time identification of RFID. His research interests include control and security of complex networks, submodular optimization, control-theoretic modeling of network security threats, and deception-based network defense mechanisms.

Linda Bushnell (F'17) is a Research Professor in the Department of Electrical and Computer Engineering at the University of Washington - Seattle. She received her Ph.D. in Electrical Engineering from University of California - Berkeley in 1994, her M.A. in Mathematics from University of California - Berkeley in 1989, her M.S. in Electrical Engineering from University of Connecticut - Storrs in 1987, and her B.S. in Electrical Engineering from University of Connecticut - Storrs in 1985. She also received her MBA from the University of Washington Foster School of Business in 2010. Her research interests include networked control systems and secure-control. She is a Fellow of the IEEE for contributions to networked control systems. She is a Fellow of IFAC for contributions to the analysis and design of networked control systems. She is a recipient of the US Army Superior Civilian Service Award, NSF ADVANCE Fellowship, and IEEE Control Systems Society Distinguished Member Award. She has been a member of the IEEE since 1985, and a member of the IEEE CSS since 1990. She is currently the Treasurer of the American Automatic Control Council, Member of the Technical Board for the International Federation on Automatic Control, Associate Editor for *Automatica*, and Associate Editor for the *IEEE Transactions on Control of Network Systems*.

Wenke Lee (F'21) is a Professor of Computer Science in the College of Computing at Georgia Institute of Technology. He also serves as the Director of the Georgia Tech Information Security Center (GTISC). He received his Ph.D. in Computer Science from Columbia University in 1999. Dr. Lee works in systems and network security. His current research projects are in the areas of botnet detection and attribution, malware analysis, virtual machine monitoring, mobile phone security, and detection and mitigation of information manipulation on the Internet, with funding from NSF, DHS, DoD, and the industry. Dr. Lee has published over 100 articles with more than 40 of them cited more than 100 times. In 2006, Dr. Lee co-founded Damballa, Inc., a spin-off from his lab that focuses on botnet detection and mitigation.

Radha Poovendran (F'15) is a Professor in the Department of Electrical and Computer Engineering at the University of Washington (UW) - Seattle. He served as the Chair of the Electrical and Computer Engineering Department at UW for five years starting January 2015. He is the Director of the Network Security Lab (NSL) at UW. He is the Associate Director of Research of the UW Center for Excellence in Information Assurance Research and Education. He received the B.S. degree in Electrical Engineering and the M.S. degree in Electrical and Computer Engineering from the Indian Institute of Technology- Bombay and University of Michigan - Ann Arbor in 1988 and 1992, respectively. He received the Ph.D. degree in Electrical and Computer Engineering from the University of Maryland - College Park in 1999. His research interests are in the areas of wireless and sensor network security, control and security of cyber-physical systems, adversarial modeling, smart connected communities, control-security, games-security, and information theoretic security in the context of wireless mobile networks. He is a Fellow of the IEEE for his contributions to security in cyber-physical systems. He is a recipient of the NSA LUCITE Rising Star Award (1999), National Science Foundation CAREER (2001), ARO YIP (2002), ONR YIP (2004), and PECASE (2005) for his research contributions to multi-user wireless security. He is also a recipient of the Outstanding Teaching Award and Outstanding Research Advisor Award from UW EE (2002), Graduate Mentor Award from Office of the Chancellor at University of California - San Diego (2006), and the University of Maryland ECE Distinguished Alumni Award (2016). He was co-author of award-winning papers including IEEE/IFIP William C. Carter Award Paper (2010) and WiOpt Best Paper Award (2012).