Regret Minimizing Audits:
A Learning-theoretic Basis for Privacy Protection

Jeremiah Blocki
Carnegie Mellon University

Nicolas Christin
Carnegie Mellon University

Anupam Datta
Carnegie Mellon University

Arunesh Sinha
Carnegie Mellon University

***Abstract*—Audit mechanisms are essential for privacy protection in permissive access control regimes, such as in hospitals where denying legitimate access requests can adversely affect patient care. Recognizing this need, we develop the first principled learning-theoretic foundation for audits. Our first contribution is a game-theoretic model that captures the interaction between the defender (e.g., hospital auditors) and the adversary (e.g., hospital employees). The model takes pragmatic considerations into account, in particular, the periodic nature of audits, a budget that constrains the number of actions that the defender can inspect, and a loss function that captures the economic impact of detected and missed violations on the organization. We assume that the adversary is worst-case as is standard in other areas of computer security. We also formulate a desirable property of the audit mechanism in this model based on the concept of regret in learning theory. Our second contribution is an efficient audit mechanism that provably minimizes regret for the defender. This mechanism learns from experience to guide the defender's auditing efforts. The regret bound is significantly better than prior results in the learning literature. The stronger bound is important from a practical standpoint because it implies that the recommendations from the mechanism will converge faster to the best fixed auditing strategy for the defender.**

I. Introduction

Audits complement access control and are essential for enforcing privacy and security policies in many situations. Specifically, audits serve an important function for protecting privacy in organizations that collect, share and use personal information. Health care providers, in particular, use permissive access control policies to grant access to patient records since wrongly denying or delaying access to a patient's record could have adverse consequences on the quality of patient care. Unfortunately, a permissive access control regime opens up the possibility of records being inappropriately accessed. Recent studies have revealed that numerous policy violations occur in the real world as employees access records of celebrities, family members, and neighbors motivated by general curiosity, financial gain, child custody lawsuits and other considerations [1], [2].

Audit mechanisms help detect such violations of policy. In practice, organizations like hospitals conduct *ad hoc* audits in which the audit log, which records accesses and disclosures of personal information, is examined to determine whether personal information was appropriately handled. In contrast to access control, the audit process cannot be completely automated for relevant privacy policies. For example, a recent formal study of privacy regulations [3] shows that a large fraction of clauses in the HIPAA Privacy Rule [4] requires some input from human auditors to enforce. We seek to develop an appropriate mathematical model for studying audit mechanisms involving human auditors. Specifically, the model should capture important characteristics of practical audit mechanisms (e.g., the periodic nature of audits), and economic considerations (e.g., cost of employing human auditors, brand name erosion and other losses from policy violations) that influence the coverage and frequency of audits.

This paper presents the first principled learning-theoretic foundation for audits of this form. Our first contribution is a **repeated game model** that captures the interaction between the defender (e.g., hospital auditors) and the adversary (e.g., hospital employees). The model includes a budget that constrains the number of actions that the defender can inspect thus reflecting the imperfect nature of audit-based enforcement, and a loss function that captures the economic impact of detected and missed violations on the organization. We assume that the adversary is worst-case as is standard in other areas of computer security. We also formulate a desirable property of the audit mechanism in this model based on the concept of *regret* in learning theory [5]. Our second contribution is a novel **audit mechanism** that provably minimizes regret for the defender. The mechanism learns from experience and provides operational guidance to the human auditor about which accesses to inspect and how many of the accesses to inspect. The regret bound is significantly better than prior results in the learning literature.

This work was partially supported by the U.S. Army Research Office contract "Perpetually Available and Secure Information Systems" (DAAD19-02-1-0389) to Carnegie Mellon CyLab, the NSF Science and Technology Center TRUST, the NSF CyberTrust grant "Privacy, Compliance and Information Risk in Complex Organizational Processes," the AFOSR MURI "Collaborative Policies and Assured Information Sharing," and HHS Grant no. HHS 90TR0003/01. Jeremiah Blocki was also partially supported by a NSF Graduate Fellowship. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

The importance of audits has been recognized in computer security and information privacy (see, for example, Lampson [6], Weitzner et al. [7]). However, unlike access control, which has been the subject of a significant body of foundational work, there is comparatively little work on the foundations of audit. Our work aims to fill this gap.

A. *Overview of Results*

Mirroring the periodic nature of audits in practice, we use a repeated game model [8] that proceeds in rounds. A round represents an audit cycle and, depending on the application scenario, could be a day, a week or even a quarter.

Adversary model: In each round, the adversary performs a set of actions (e.g., accesses patient records) of which a subset violates policy. Actions are classified into types. For example, accessing celebrity records could be a different type of action from accessing non-celebrity records. The adversary capabilities are defined by parameters that impose upper bounds on the number of actions of each type that she can perform in any round. We place no additional restrictions on the adversary's behavior. In particular, we do not assume that the adversary violates policy following a fixed probability distribution; nor do we assume that she is rational. Furthermore, we assume that the adversary knows the defender's strategy (audit mechanism) and can adapt her strategy accordingly.

Defender model: In each round, the defender inspects a subset of actions of each type performed by the adversary. The defender has to take two competing factors into account. First, inspections incur cost. The defender has an audit budget that imposes upper bounds on how many actions of each type she can inspect. We assume that the cost of inspection increases linearly with the number of inspections. So, if the defender inspects fewer actions, she incurs lower cost. Note that, because the defender cannot know with certainty whether the actions not inspected were malicious or benign, this is a game of imperfect information [9]. Second, the defender suffers a loss in reputation for detected violations. The loss is higher for violations that are detected externally (e.g., by an Health and Human Services audit, or because information leaked as a result of the violation is publicized by the media) than those that are caught by the defender's audit mechanism, thus incentivizing the defender to inspect more actions.

In addition, the loss incurred from a detected violation depends on the type of violation. For example, inappropriate access of celebrities' patient records might cause higher loss to a hospital than inappropriate access of other patients' records. Also, to account for the evolution of public memory, we assume that violations detected in recent rounds cause greater loss than those detected in rounds farther in the past. The defender's audit mechanism has to take all these considerations into account in prescribing the number of actions of each type that should be inspected in a given round, keeping in mind that the defender is playing against the powerful strategic adversary described earlier.

Note that for adequate privacy protection, the economic and legal structure has to ensure that it is in the best interests of the organization to invest significant effort into auditing. Our abstraction of the reputation loss from policy violations that incentivizes organizations to audit can, in practice, be achieved through penalties imposed by government audits as well as through market forces, such as brand name erosion and lawsuits.

Regret property: We formulate a desirable property for the audit mechanism by adopting the concept of regret from online learning theory. The idea is to compare the loss incurred when the real defender plays according to the strategy prescribed by the audit mechanism to the loss incurred by a hypothetical defender with perfect knowledge of the number of violations of each type in each round. The hypothetical defender is allowed to pick a fixed strategy to play in each round that prescribes how many actions of each type to inspect. The *regret* of the real defender in hindsight is the difference between the loss of the hypothetical defender and the actual loss of the real defender averaged over all rounds of game play. We require that the regret of the audit mechanism quickly converge to a small value and, in particular, that it tends to zero as the number of rounds tends to infinity.

Intuitively, this definition captures the idea that although the defender does not know in advance how to allocate her audit budget to inspect different types of accesses (e.g., celebrity record accesses vs. non-celebrity record accesses), the recommendations from the audit mechanism should have the desirable property that over time the budget allocation comes close to the optimal fixed allocation. For example, if the best strategy is to allocate 40% of the budget to inspect celebrity accesses and 60% to non-celebrity accesses, then the algorithm should quickly converge towards these values.

Audit mechanism: We develop a new audit mechanism that provably minimizes regret for the defender. The algorithm, which we name RMA (for Regret Minimizing Audit) is efficient and can be used in practice. In each round of the game, the algorithm prescribes how many actions of each type the defender should inspect. It does so by maintaining weights for each possible defender action and picking an action with probability proportional to the weight of that action. The weights are updated based on a loss estimation function, which is computed from the observed loss in each round. Intuitively, the algorithm learns the optimal distribution over actions by increasing the weights of actions that yielded better payoff than the expected payoff of the current distribution and decreasing the weight of actions that yielded worse payoff.

Our main technical result (Theorem 1) is that the exact bound on regret for RMA is approximately $2\sqrt{2\frac{\ln N}{T}}$ where

N is the number of possible defender actions and T is the number of rounds (audit cycles). This bound improves the best known bounds of $O\left(\frac{N^{1/3}\log N}{\sqrt[3]{T}}\right)$ for regret minimization over games of imperfect information. The main novelty is in the way we use a loss estimation function and characterize its properties to achieve the significantly better bounds. Specifically, RMA follows the structure of a regret minimization algorithm for perfect information games, but uses the estimated loss instead of the true loss to update the weights in each round. We define two properties of the loss estimation function—*accuracy* (capturing the idea that the expected error in loss estimation in each round is zero) and *independence* (capturing the idea that errors in loss estimation in each round are independent of the errors in other rounds)—and prove that any loss estimation function that satisfies these properties results in regret that is close to the regret from using an actual loss function. Thus, our bounds are of the same order as regret bounds for perfect information games. The better bounds are important from a practical standpoint because they imply that the algorithm converges to the optimal fixed strategy much faster.

The rest of the paper is organized as follows. Section II presents the game model formally. Section III presents the audit mechanism and the theorem showing that the audit mechanism provably minimizes regret. Section IV discusses the implications and limitations of these results. Section V describes in detail the loss estimation function, a core piece of the audit mechanism. Section VI presents the outline of the proof of the main theorem of the paper (Theorem 1) while the complete proofs are presented in the appendices. Section VII provides a detailed comparison with related work, in particular, focusing on technical results on auditing in the computer security literature and regret minimization in the learning theory literature. Finally, Section VIII presents our conclusions and identifies directions for future work.

II. Model

We model the internal audit process as a repeated game played between a defender (organization) and an adversary (employees). In the presentation of the model we will use the following notations:

- Vectors are represented with an arrow on top, e.g., $\vec{v}$ is a vector. The i^{th} component of a vector is given by $\vec{v}\,[i]$. $\vec{v} \leq \vec{a}$ means that both vectors have the same number of components and for any component i, $\vec{v}\,[i] \leq \vec{a}\,[i]$.
- Random variables are represented in boldface, e.g., $\mathbf{x}$ and $\mathbf{X}$ are random variables.

The repeated game we consider is fully defined by the players, the time granularity at which the game is played, the actions the players can take, and the utility the players obtain as a result of the actions they take. We next discuss these different concepts in turn and illustrate them using a running example from an hospital.

Players: The game is played between the organization and its employees. We refer to the organization as $\mathcal{D}$ (*defender*). We subsume all employees into a single player $\mathcal{A}$ (*adversary*). In this paper, we are indeed considering a worst-case adversary, who would be able to control all employees and coerce them into adopting the strategy most damaging to the organization. In our running example, the players are the hospital and all the employees.

Round of play: In practice, audits are usually performed periodically. Thus, we adopt a discrete-time model for this game, where time points are associated with rounds. Each round of play corresponds to an audit cycle. We group together all of the adversary's actions in a given round.

Action space: $\mathcal{A}$ executes tasks, i.e., actions that are permitted as part of their job. We only consider tasks that can later be audited, e.g., through inspection of logs. We can distinguish $\mathcal{A}$'s tasks between legitimate tasks and violations of a specific privacy policy that the organization $\mathcal{D}$ must follow. Different types of violations may have a different impact on the organization. We assume that there are K different types of violations that $\mathcal{A}$ can commit (e.g., unauthorized access to a celebrity's records, unauthorized access to a family member's records, ...). We further assume that the severity of violations, in terms of economic impact on the organization, varies with the types.

In each audit cycle, the adversary $\mathcal{A}$ chooses two quantities for each type k: the number of tasks she performs, and the number of such tasks that are violations. If we denote by U_k the maximum number of type k tasks that $\mathcal{A}$ can perform, then $\mathcal{A}$'s entire action space is given by $A \times A$ with $A = \prod_{i=1}^{K}\{1, \ldots, U_i\}$. In a given audit cycle, an action by $\mathcal{A}$ in the game is given by $\langle \vec{a}, \vec{v} \rangle$, where the components of $\vec{a}$ are the number of tasks of each type $\mathcal{A}$ performs, and the components of $\vec{v}$ are the number of violations of each type. Since violations are a subset of all tasks, we always have $\vec{v} \leq \vec{a}$. In our hospital example, we consider two types of patient medical records: access to celebrity records and access to regular person's record. A typical action may be 250 accesses to celebrity records with 10 of them being violations and 500 accesses to non-celebrity records with 50 of them being violations. Then $\mathcal{A}$'s action is $\langle\langle 250, 500\rangle, \langle 10, 50\rangle\rangle$.

We assume that the defender $\mathcal{D}$ can classify each adversary's task by types. However, $\mathcal{D}$ cannot determine whether a particular task is legitimate or a violation without investigating. $\mathcal{D}$ can choose to *inspect* or *ignore* each of $\mathcal{A}$'s tasks. We assume that inspection is perfect, i.e., if a violation is inspected then it is detected. The number of inspections that $\mathcal{D}$ can conduct is bounded by the number of tasks that $\mathcal{A}$ perform, and thus, $\mathcal{D}$'s actions are defined by a vector $\vec{s} \in A$, with $\vec{s} \leq \vec{a}$. That is, $\mathcal{D}$ chooses a certain number of tasks of each type to be inspected. Further, in each round t, $\mathcal{D}$ has a fixed budget B^t to allot to all inspections. We represent the (assumed fixed) cost of inspection for each

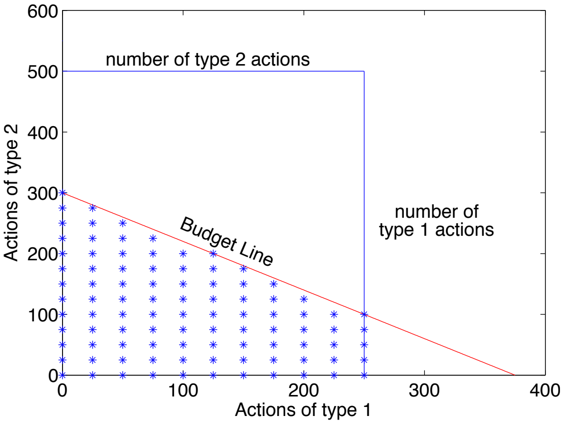

Figure 1. Feasible audit space, represented by the shaded area.

type of violation by $\vec{C}$. The budgetary constraints on $\mathcal{D}$ are thus given by $\vec{C} \cdot \vec{s} \leq B^t$ for all t. Continuing our hospital example, the maximum number of tasks of each type that $\mathcal{D}$ can inspect is $\langle 250, 500 \rangle$. Assuming a budget of 1500 and cost of inspection vector $\langle 4, 5 \rangle$, $\mathcal{D}$'s inspection space is further constrained, and then the feasible inspections are $\{\langle x, y \rangle \mid 4x + 5y \leq 1500, 0 \leq x \leq 250, 0 \leq y \leq 500\}$. The discrete feasible audit points are indicated (not all points are shown) with the asterisks in Figure 1.

Violation detection: Given the budgetary constraints $\mathcal{D}$ faces, $\mathcal{D}$ cannot, in general, inspect all of $\mathcal{A}$'s tasks (i.e., $\vec{C} \cdot \vec{a} > B^t$). Hence, some violations may go undetected internally, but could be detected externally. Governmental audits, whistle-blowing, information leaks are all but examples of situations that could lead to external detection of otherwise unnoticed violations. We assume that there is a fixed exogenous probability p ($0 < p < 1$) of an internally undetected violation getting caught externally.

Formally, we define the outcome of a single audit cycle as the outcome of the internal audit and the number of violations detected externally. Due to the probabilistic nature of all quantities, this outcome is a random variable. Let $\vec{\mathbf{O}}^t$ be the outcome for the t^{th} round. Then $\vec{\mathbf{O}}^t$ is a tuple $\langle \vec{\mathbf{O}}^t_{int}, \vec{\mathbf{O}}^t_{ext} \rangle$ of violations caught internally and externally. By our definitions, the probability mass function for $\vec{\mathbf{O}}^t_{int}$ is parameterized by $\langle \vec{a}, \vec{v} \rangle$ and $\vec{s}$, and the probability mass function for $\vec{\mathbf{O}}^t_{ext}$ conditioned on $\vec{\mathbf{O}}^t_{int}$ is parameterized by p. We make no assumptions about this probability mass function. Observe that, because not all tasks can be inspected, the organization does not get to know the exact number of violations committed by the employees, which makes this a game of imperfect information. In our hospital example, given that $\mathcal{A}$'s action is $\langle \langle 250, 500 \rangle, \langle 10, 50 \rangle \rangle$. In one possible scenario the hospital performs $\langle 125, 200 \rangle$ inspections. These inspections result in $\langle 7, 30 \rangle$ violations detected internally and $\langle 2, 10 \rangle$ violations detected externally.

Utility function: Since we consider a worst-case adversary, $\mathcal{A}$'s payoff function is irrelevant to our model. On the other hand, the utility function of $\mathcal{D}$ influences the organization's strategy. We define $\mathcal{D}$'s utility as the sum of $\mathcal{D}$'s *reputation* and the cost of inspecting $\mathcal{A}$'s actions. In essence, $\mathcal{D}$ has to find the right trade-off between inspecting frequently (which incurs high costs) and letting violations occur (which degrades its reputation, and thus translates to lost revenue).

We assume that the cost of inspection is linear in the number of inspections for each type of action. Hence, if $\mathcal{D}$'s action is $\vec{s}$ then inspection costs are $\vec{C} \cdot \vec{s}$. In our running example of the hospital, this cost is $\langle 4, 5 \rangle . \langle 125, 200 \rangle = 1500$, which is the also the full budget in our example.

We assume that any violation caught (internally, or externally) in a round affects $\mathcal{D}$'s reputation not only in that round, but also in future rounds and that the exact effect in any future round is known. We capture this by defining a function $r^t_k : \{1, \ldots, U_k\} \times \{1, \ldots, U_k\} \times \mathbb{N} \to \mathbb{R}$ for each type k of violation. In round t, r^t_k takes as inputs the number of violations of type k detected internally, the number of violations of type k caught externally, and an integer argument τ. r^t_k outputs the effect of the violations (measured as the loss in reputation) occurring in round t on $\mathcal{D}$'s reputation in round $t + \tau$. We assume that violations of a given type always have the same effect on reputation, that is, r^t_k is actually independent of t, which allows us to use the shorthand notation r_k from here on.

Violations caught far in the past should have a lesser impact on reputation than recently caught violations, thus, r_k should be monotonically decreasing in the argument τ. We further assume violations are forgotten after a finite amount of rounds m, and hence do not affect reputation further. In other words, if $\tau \geq m$ then for any type k, any round t, and any tuple of violations caught $\langle \vec{O}^t_{int}[k], \vec{O}^t_{ext}[k] \rangle$, $r_k(\vec{O}^t_{int}[k], \vec{O}^t_{ext}[k], \tau) = 0$.

Moreover, externally caught violations should have a worse impact on reputation than internally detected violations, otherwise the organization has a trivial incentive never to inspect. Formally, r_k has the following property. If for any two realized outcomes $\vec{O}^l$ and $\vec{O}^j$ at rounds l and j, we have $\vec{O}^l_{int}[k] + \vec{O}^l_{ext}[k] = \vec{O}^j_{int}[k] + \vec{O}^j_{ext}[k]$ (i.e., same number of total violations of type k in rounds j and l) and $\vec{O}^l_{ext}[k] > \vec{O}^j_{ext}[k]$ (i.e., for type k, the number of violations caught externally is more than the number caught internally) then for any τ such that $0 \leq \tau < m$, $r_k(\vec{O}^l[k], \tau) > r_k(\vec{O}^j[k], \tau)$.

We use r_k to define a measure of reputation. Because, by construction, violations only affect at most m rounds of play, we can write the reputation $\mathbf{R}_0$ of the organization at round t as a random variable function of the probabilistic outcomes $\vec{\mathbf{O}}^t, ..., \vec{\mathbf{O}}^{t-m+1}$:

$$\mathbf{R}_0(\vec{\mathbf{O}}^t, ..., \vec{\mathbf{O}}^{t-m+1}) = R - \sum_{k=1}^{K} \sum_{j=t-m+1}^{t} r_k(\vec{\mathbf{O}}^j[k], t-j) \ ,$$

where R is the maximum possible reputation. We assume that at the start of the game the reputation is R, and that r_k's are so that $\mathbf{R}_0$ is always non-negative.

We cannot, however, directly use $\mathbf{R}_0$ in our utility function. Indeed, $\mathbf{R}_0$ is history-dependent, and the repeated game formalism requires that the utility function be independent of past history. Fortunately, a simple construction allows to closely approximate the actual reputation, while at the same time removing dependency on past events. Consider the following function $\mathbf{R}$:

$$\mathbf{R}(\vec{\mathbf{O}}^t) = R - \sum_{k=1}^{K} \sum_{j=0}^{m-1} r_k(\vec{\mathbf{O}}^t[k], j) \ .$$

Rather than viewing reputation as a function of violations that occurred in the past, in round t, the reputation function $\mathbf{R}$ instead immediately accounts for reputation losses that will be incurred in the future (in rounds $t+\tau$, $0 \le \tau < m$) due to violations occurring in round t.

While $\mathbf{R}$ and $\mathbf{R}_0$ are different reputation functions, when we compute the difference of their averages over T rounds, denoting by $\vec{v}_{\max}$ the maximum possible number of violations, we obtain:

$$\frac{1}{T}\sum_{\tau=t}^{t+T} |\mathbf{R}(\vec{\mathbf{O}}^\tau) - \mathbf{R}_0(\vec{\mathbf{O}}^\tau, \ldots, \vec{\mathbf{O}}^{\tau-m})| \le \frac{1}{T}\sum_{k=1}^{K}\sum_{j=1}^{m-1} j \cdot r_k(\vec{v}_{max}(k), j) \ .$$

The right-hand side of the above inequality goes to zero as T grows large. Hence, using $\mathbf{R}$ to model reputation instead of $\mathbf{R}_0$ does not significantly impact the utility function of the defender. We define the utility function at round t in the repeated game by the random variable

$$\mathbf{L}^t(\langle \vec{a}^t, \vec{v}^t \rangle, \vec{s}^t) = \mathbf{R}(\vec{\mathbf{O}}^t) - \vec{C} \cdot \vec{s}^t \ .$$

Since utility gains are only realized through loss reduction, we will equivalently refer to $\mathbf{L}$ as a loss function from here on.

An example of the loss of reputation function r_k is $r_k(O, t) = c_k(O_{int} + 2 \times O_{ext})\delta^t$ for $0 \le t < m$ and $r_k(O,t) = 0$ for $t \ge m$, where $\delta \in (0,1)$. Observe that r_k decreases with t and puts more weight on external violations. Also for the same number of violations and same value for argument t r_k has different values for different types of violations due to c_k that varies with the types. Then, considering only one type of violation, the loss function can be written as

$$\mathbf{L}^t(\langle \vec{a}^t, \vec{v}^t \rangle, \vec{s}^t) = R - c_1 \sum_{j=0}^{m-1} (\mathbf{O}^t_{int} + 2 \times \mathbf{O}^t_{ext})\delta^j - \vec{C} \cdot \vec{s}^t \ .$$

Observe that we can expand the summation in the above equation to get $c_1(1 + \delta ... + \delta^{m-1})\mathbf{O}^t_{int} + 2c_1(1 + \delta ... + \delta^{m-1})\mathbf{O}^t_{ext}$. Then let $R_{int} = c_1(1 + \delta ... + \delta^{m-1})$ and similarly let $R_{ext} = 2c_1(1 + \delta ... + \delta^{m-1})$. We can rewrite the loss equation above as

$$\mathbf{L}^t(\langle \vec{a}^t, \vec{v}^t \rangle, \vec{s}^t) = R - R_{int} \cdot \mathbf{O}^t_{int} - R_{ext} \cdot \mathbf{O}^t_{ext} - \vec{C} \cdot \vec{s}^t \ .$$

III. Audit Mechanism and Property

In this section, we present our audit mechanism RMA and the main theorem that characterizes its property. RMA prescribes the number of tasks of each type that the defender should inspect in each round of the repeated game. The property compares the loss incurred by the defender when she follows RMA to the loss of a hypothetical defender who has perfect knowledge of how many violations of each type occurred in each round, but must select one fixed action $\vec{s}$ to play in every round. In particular, we obtain exact bounds on the defender's regret and demonstrate that the average regret across all rounds converges to a small value relatively quickly.

A. Audit Mechanism

Our Regret Minimizing Audit (RMA) mechanism is presented as Algorithm 1. In each round of the game, RMA prescribes how many tasks of each type the defender should inspect. It does so by maintaining weights for each possible defender action (referred to as "experts" following standard terminology in the learning literature) and picking an action with probability proportional to the weight of that action. For example, in a hospital audit, with two types of tasks—celebrity record access and regular record access—the possible defender actions $\vec{s}$ are of the form $\langle k_1, k_2 \rangle$ meaning that k_1 celebrity record accesses and k_2 regular record accesses are inspected. The weights are updated based on an estimated loss function, which is computed from the observed loss in each round. γ is a learning parameter for RMA. Its value is less than but close to 1. We show how to choose γ in sub-section III-B.

RMA is fast and could be run in practice. Specifically, the running time of RMA is $O(N)$ per round where N is the number of experts.

In more detail, RMA maintains weights $w^t_{\vec{s}}$ for all experts [10]. $w^t_{\vec{s}}$ is the weight of the expert before round t has been played. Initially, all experts are equally weighted. In each round, an action is probabilistically selected for the defender. As discussed in the Section II there are two factors that constrain the set of actions available to the defender: the number of tasks performed by the adversary and the budget available for audits. In our hospital example we had the feasible audit space as $\{\langle x, y \rangle \mid 4x + 5y \le 1500, 0 \le x \le 250, 0 \le y \le 500\}$. These considerations motivate the definition of the set AWAKE^t of experts that are awake in round t (Step 1). Next, from this set of awake experts, one is chosen with probability $p^t_{\vec{s}}$ proportional to the weight of that expert (Steps 2, 3, 4). Continuing our hospital example, 250 celebrity record accesses and 500 regular record accesses

Algorithm 1 RMA

```
• Initialize: Set $w^0_{\vec{s}} = 1$ for each expert.
• Select Move: On round $t$ let $\langle \vec{a}^t, \vec{v}^t \rangle$ denote the action of the adversary.
    1) Set
         $\mathsf{AWAKE}^t = \{\vec{s} : \vec{s} \leq \vec{a}^t \wedge \vec{C} \cdot \vec{s} \leq B^t\ \}$ .
    2) Set
         $W^t = \sum_{\vec{s} \in \mathsf{AWAKE}^t} w^t_{\vec{s}}$ .
    3) Set
         $p^t_{\vec{s}} = \frac{w^t_{\vec{s}}}{W^t}$ ,
       for $\vec{s} \in \mathsf{AWAKE}$. Otherwise set $p^t_{\vec{s}} = 0$.
    4) Play $\vec{s}$ with probability $p_{\vec{s}}$ (randomly select one expert to follow).
• Estimate loss function: Set $\tilde{\mathbf{L}} = \mathsf{est}\left(\vec{\mathbf{O}}^t, \vec{s}^t\right)$.
• Update Weights: For each $\vec{s} \in \mathsf{AWAKE}^t$ update
         $w^{t+1}_{\vec{s}} = w^t_{\vec{s}} \gamma^{\tilde{\mathbf{L}}^t(\vec{s}) - \gamma \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})}$ ,
  where $\tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA}) = \sum_{\vec{s}} p^t_{\vec{s}} \tilde{\mathbf{L}}^t(\vec{s})$, is the expected (estimated) loss of the algorithm.
```

will be inspected with probability 0.3 in a round if the expert $\langle 250, 500 \rangle$ is awake in that round and its weight divided by the total weight of all the experts who are awake is 0.3. Technically, this setting is close to the setting of sleeping experts in the regret minimization literature [5], [11].

However, we also have to deal with imperfect information. Since only one action (say $\vec{s}^t$) is actually played by the defender in a round, she observes an outcome $\vec{\mathbf{O}}^t$ for that action. For example, the $\langle 250, 500 \rangle$ inspection might have identified 5 celebrity record access violations and 3 regular record access violations internally; the same number of violations may have been detected externally. Based on this observation, RMA uses an algorithm est to compute an estimated loss function $\tilde{\mathbf{L}}$ for *all* experts (not just the one she played). We describe properties of the loss function for which this estimation is accurate in subsection V. We also provide an example of a natural loss function that satisfies these properties. Finally, the estimated loss function is used to update the weights for all the experts who are awake. Intuitively, the multiplicative weight update ensures that the weights of experts who performed better than their current distribution increase and the weights for those who performed worse decrease. In RMA the weight for expert $\vec{s}$ increases when $\tilde{\mathbf{L}}^t(\vec{s}) - \gamma \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})$ is negative, i.e., $\tilde{\mathbf{L}}^t(\vec{s}) < \gamma \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})$, and since γ is close to 1, the loss of expert $\vec{s}$ is less than the loss of RMA, i.e., the expert $\vec{s}$ performed better than RMA.

B. Property

The RMA mechanism provides the guarantee that the defender's *regret* is minimal. *Regret* is a standard notion from the online learning literature. Intuitively, regret quantifies the difference between the loss incurred by the defender when she follows RMA and the loss of a hypothetical defender who has perfect knowledge of how many violations of each type occurred in each round, but must select one fixed action (or expert) to play in every round. Our main theorem establishes exact bounds on the defender's regret.

Let T denote the total number of rounds played, $I(t)$ be a *time selection function* whose output is either 0 or 1, $\mathbf{L}^t(\vec{s}) = \mathbf{L}^t\left(\langle \vec{a}^t, \vec{v}^t \rangle, \vec{s}\right)$ be the loss function at time t after the adversary has played $\langle \vec{a}^t, \vec{v}^t \rangle$, and $p^t_{\vec{s}}$ be the probability of choosing the action $\vec{s}$ in round t while following RMA. We define the total loss of RMA as follows:

$$\mathbf{Loss}\left(\mathsf{RMA}, I\right) = \sum_{t=1}^{T} \sum_{\vec{s}} I(t) p^t_{\vec{s}} \mathbf{L}^t(\vec{s}) \ .$$

Similarly for each fixed expert $\vec{s}$ we set the following loss function

$$\mathbf{Loss}\left(\vec{s}, I\right) = \sum_{t=1}^{T} I(t) \mathbf{L}^t(\vec{s}) \ .$$

We use $\mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right)$ to denote our regret in hindsight of not playing the fixed action $\vec{s}$ when it was available. Formally,

$$\mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right) = \mathbf{Loss}\left(\mathsf{RMA}, I_{\vec{s}}\right) - \mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) .$$

Here, $I_{\vec{s}}(t)$ is the time selection function that selects only the times t that action $\vec{s}$ is available.

As before, we use N to denote the total number of fixed actions available to the defender. If T is known in advance we can obtain the bound in Theorem 1 below by setting γ to be $1 - \sqrt{\frac{2 \ln N}{T}}$. Otherwise, if T is not known in advance we can dynamically tune γ to obtain similar bounds. See Remark 7 for more details on dynamic tuning.

Theorem 1. *For all $\epsilon > 0$,*

$$\Pr\left[\begin{array}{cc} \exists \vec{s}, & \frac{\mathbf{Regret}(\mathsf{RMA}, \vec{s})}{T} \geq 2\sqrt{2\frac{\ln N}{T}} + \\ & 2\sqrt{\frac{2\ln\left(\frac{4N}{\epsilon}\right)}{T}} + \frac{2}{T}\ln N \end{array}\right] \leq \epsilon \ .$$

Remark 1. *To prove this theorem we need to make several reasonable assumptions about the accuracy of our loss function estimator* est*. We discuss these assumptions in section V. We also assume that losses have been scaled so that $\mathbf{L}^t(x) \in [0, 1]$.*

Remark 2. *This bound is a worst case regret bound. The guarantee holds against* any *attacker.* $\mathbf{Regret}$ *may typically be lower than this, e.g. when hospital employees do not behave adversarially.*

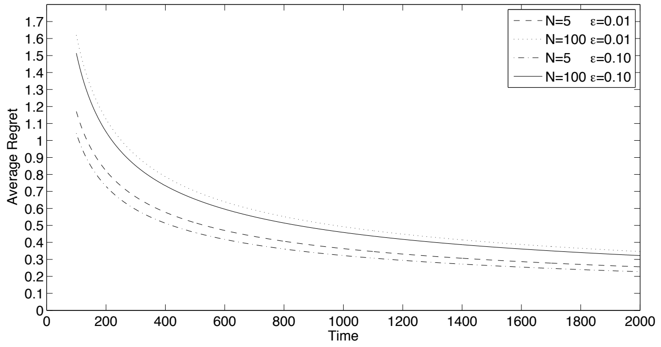

Figure 2. Worst case Average Regret vs Time for different values of N and ϵ

In order to understand what this bound means, consider the following example scenario. Suppose that an employee at a hospital can access two types of medical records—celebrity or regular. The defender can choose to inspect accesses of a certain type *lightly, moderately,* or *heavily*. In this case, the defender has $N = 9$ possible pairs of actions in each round. If the hospital performs daily audits (which some hospitals currently do for celebrity record accesses) over a 5 year period, then $T = 365 \times 5 = 1825$. For simplicity, assume that each action $\vec{s}$ is available every day. In this case, the theorem guarantees that except with probability $\epsilon = \frac{1}{100}$, the average regret of RMA does not exceed 29%:

$$\frac{\textbf{Regret}\left(\mathsf{RMA}, \vec{s}\right)}{T} < 0.29 \ .$$

Note that there are several existing algorithms for regret minimization in games with imperfect information [9], [12]–[14]. These algorithms do guarantee that as $T \to \infty$ the average regret will tend to 0, but the convergence rate is unacceptably slow for our audit model (see Section VII-B for a more detailed comparison). The convergence rate of RMA is significantly faster. Also, in contrast to prior work, we focus on *exact* (not *asymptotic*) regret bounds for our algorithm. This is important because in practice we care about the value of the bound for a fixed value of T (as in the example above), not merely that it tends to 0 as $T \to \infty$.

IV. Discussion

A few characteristics of the model and algorithms described above may not necessarily be evident from the technical presentation given above, and warrant further discussion.

Figure 2 shows the variation of average regret with time for different values of N and ϵ. As can be seen, the RMA algorithm produces smaller average regret bounds for higher values of time T and lower values of N. In other words, and quite intuitively, a high audit frequency ensures low regret. Some medical centers carry out audits every week; RMA is particularly appropriate for such high frequency audits. Lower values of N means that RMA's performance is compared to fewer fixed strategies and hence yields lower regret. One situation in which N could be low is when the fixed strategies correspond to discrete levels of audits coverage used by the organization. Also, higher values of ϵ yield smaller average regret bounds. Indeed, ϵ is a measure of uncertainty on the stated bound. Thus, when higher values of ϵ are tolerable, we obtain tighter regret bounds, but at the expense of greater uncertainty on whether those bounds are met.

We have already noted that all actions may not be available at all times. The result in Theorem 1 bounds the average regret taken over all rounds of the game. It is easy to modify the proof of Theorem 1 to obtain average regret bounds for each expert such that the average is taken over the time for which that expert is awake. The bound thus obtained is of the same *order* as the bound in Theorem 1, but all instance of T in Theorem 1 are replaced by $T_{\vec{s}}$, where $T_{\vec{s}}$ is the time for which expert $\vec{s}$ is awake. Similar result for the traditional sleeping experts setting can be found in Blum et al. [11]. The modified bound equation exhibits the fact that the average regret bound for a given inspection vector (average taken over the time for which that inspection was available) depends on how often this inspection vector is available to the defender. If a given inspection vector is only available for a few audit cycles, the average regret bound may be relatively high. The situation is analogous to whitewashing attacks [15], where the adversary behaves in a compliant manner for many rounds to build up reputation, attacks only once, and immediately leaves the game after the attack. For instance, a spy infiltrates an organization, becomes a trusted member by behaving as expected, and then suddenly steals sensitive data. However, we argue that, rather than being an auditing issue, whitewashing attacks can be handled by a different class of mechanisms, e.g., that prevent the adversary from vanishing once she has attacked.

Furthermore, RMA guarantees low *average* regret compared to playing a fixed action (i.e., inspection vector) in the audit cycle in which that action was available; it does not guarantee violations will not happen. In particular, if a certain type k of violation results in catastrophic losses for the organization (e.g., losses that threaten the viability of the organization itself), tasks of type k should always be fully inspected.

While the discussion surrounding the RMA algorithm focused only on one kind of experts (which recommend how many tasks of each type to inspect in any round), RMA applies more generally to any set of experts. For example, we could include an expert who recommends a low inspection probability when observed violations are below a certain threshold and a higher inspection probability when observed violations are above that threshold. Over time, the RMA algorithm will perform as well as any such expert. Note.

however, that if we make the size (N) of the set of experts too large, the regret bounds from Theorem 1 will be worse. Thus, in any particular application, the RMA algorithm will be effective if appropriate experts are chosen without making the set of experts too large.

Finally, we note that non-compliance with external privacy regulations may not only cause a loss of reputation for the organization, but can also result in fines being levied against the organization. For instance, a hospital found in violation of HIPAA provisions [4] in the United States will likely face financial penalties in addition to damaging its reputation. We can readily extend our model to account for such cases, by forcing the defender (organization) to perform some minimum level of audit (inspections) to meet the requirements stipulated in the external regulations. For example, we can constrain the action space available to the defender by removing strategies such as "never inspect." As long as the budgetary constraints allow the organization to perform inspections in addition to the minimal level of audit required by law, the guarantees provided by RMA still hold. Indeed, Theorem 1 holds as long as there is at least one awake expert in each round.

V. Estimating Losses

RMA uses a function $\mathsf{est}\left(\vec{\mathbf{O}}^t, \vec{s}^t\right)$ to estimate the loss function $\tilde{\mathbf{L}}^t$. In this section, we formally define two properties—*accuracy* and *independence*; the regret bound in Theorem 1 holds for any estimator function that satisfies these two properties. We also provide an example of a loss function estimator algorithm that provably satisfies these properties, thus demonstrating that such estimator functions can in fact be implemented. The use of an estimator function and the characterization of its properties is a novel contribution of this paper that allows us to achieve significantly better bounds than prior work in the regret minimization literature for repeated games of imperfect information (see Section VII for a detailed comparison).

Estimator Properties: The function $\tilde{\mathbf{L}}^t = \mathsf{est}\left(\vec{\mathbf{O}}^t, \vec{s}^t\right)$ should be efficiently computable for practical applications. Note that the loss estimation at time t depends on the outcome $\vec{\mathbf{O}}^t$ (violations of each type detected internally and externally) and the defender's action $\vec{s}^t$ at time t. Intuitively, the function outputs an estimate of the loss function by estimating the number of violations of each type based on the detected violations of that type and the probability of inspecting each action of that type following the defender's action.

For each defender action (expert) $\vec{s}$, we define the random variable

$$\mathbf{X}_{\vec{s}}^t = \tilde{\mathbf{L}}^t(\vec{s}) - \mathbf{L}^t(\vec{s}).$$

Intuitively, $\mathbf{X}_{\vec{s}}^t$ is a random variable representing our estimation error at time t after the actions $\langle \vec{v}^t, \vec{a}^t \rangle$ and $\vec{s}^t$ have been fixed by the adversary and the defender respectively.

Because we have assumed that our loss functions are scaled so that $\tilde{\mathbf{L}}^t(\vec{s}), \mathbf{L}^t(\vec{s}) \in [0,1]$ we have $\mathbf{X}_{\vec{s}}^t \in [-1,1]$. This property of $\mathbf{X}_{\vec{s}}^t$ is useful in bounding the regret as we discuss later.

Formally, we assume the following properties about est:

1) **Accuracy:** $E\left[\mathbf{X}_{\vec{s}}^j\right] = 0$ for $0 \leq j \leq T$.
2) **Independence:** $\forall \vec{s}$, $\mathbf{X}_{\vec{s}}^1, \ldots, \mathbf{X}_{\vec{s}}^T$ are all independent random variables.

Any estimation scheme est that satisfies both properties can be plugged into RMA yielding the regret bound in Theorem 1. Informally, *accuracy* captures the idea that the estimate is accurate in an expected sense while *independence* captures the idea that the error in the estimate in each round is independent of the error in all other rounds. We motivate these properties by way of an example.

Remark 3. *In fact if our estimation scheme only satisfied δ-accuracy, i.e., $\left|E\left[\mathbf{X}_{\vec{s}}^j\right]\right| < \delta$, then we could still guarantee that the average regret bounds from Theorem 1 still hold with an extra additive term δ. Formally, the following property holds: for all $\epsilon \in (0,1)$*

$$\Pr\left[\begin{array}{cc} \exists \vec{s}, & \begin{array}{c} \frac{\mathbf{Regret}(\mathsf{RMA},\vec{s})}{T} \geq \delta + 2\sqrt{2\frac{\ln N}{T}} + \\ 2\sqrt{\frac{2\ln\left(\frac{4N}{\epsilon}\right)}{T}} + \frac{2}{T}\ln N \end{array} \end{array}\right] \leq \epsilon \ .$$

Example Loss Function: We return to our running example of the hospital. We use the example reputation (loss) function from the previous section:

$$\mathbf{L}^t(\vec{s}) = R - \left(\vec{\mathbf{O}}_{int}^t \cdot \vec{R}_{int} + \vec{\mathbf{O}}_{ext}^t \cdot \vec{R}_{ext} + \vec{C} \cdot \vec{s}\right) \ .$$

To simplify our presentation we assume that there is only one type of violation. It is easy to generalize the loss function that we present in this example to include multiple types of violations.

$$\mathbf{L}^t(s) = R - \left(\mathbf{O}_{int}^t \times R_{int} + \mathbf{O}_{ext}^t \times R_{ext} + C \times s\right) \ .$$

Here $\mathbf{O}_{int}^t$ represents the number of violations caught internally after the actions $\langle v^t, a^t \rangle$ and s^t are played by the adversary and the defender respectively, R_{int} (resp. R_{ext}) captures the damage to the hospital's reputation when a violation is caught internally (resp. externally), and C is the cost of performing one inspection. Notice that

$$E\left[\mathbf{O}_{ext}^t \times R_{ext}\right] = p\left(v^t - E\left[\mathbf{O}_{int}^t\right]\right) \times R_{ext} \ ,$$

where p is the probability that an undetected violation gets caught externally. Therefore,

$$\begin{aligned} E\left[\mathbf{L}^t(s)\right] \quad &= \quad R - \left(E\left[\mathbf{O}_{int}^t\right]\left(R_{int} - p \times R_{ext}\right)\right. \\ &\qquad \left. + p \times v^t \times R_{ext} + C \times s\right) \ . \end{aligned}$$

We can set $R' = \left(R_{int} - p \times R_{ext}\right)$ and then

$$E\left[\mathbf{L}^t(s)\right] = R - \left(E\left[\mathbf{O}_{int}^t\right] \times R' + p \times v^t \times R_{ext} + C \times s\right) \ .$$

In our loss model, we allow the defender to use any recommendation algorithm REC that sorts all a^t actions at time t and probabilistically recommends s^t actions to inspect. We let $p_d \leq 1$ denote the probability that the d^{th} inspection results in a detected violation, where this probability is over the coin flips of the recommendation algorithm REC. Because this probability is taken over the coin flips of REC the outcome $\mathbf{O}^t_{int}$ is independent of previous outcomes once $\langle \vec{a}^t, \vec{v}^t \rangle, \vec{s}^t$ have been fixed.

For example, a naive recommendation algorithm REC might just select a few actions uniformly at random and recommend that the defender inspect these actions. In this case $p_j = \frac{v^t}{a^t}$ for each j. (Remember that in this example we consider only one type of violation, so v^t and a^t are scalars). If REC is more clever, then we will have $p_1 > \frac{v^t}{a^t}$. In this case the p_j's will also satisfy diminishing returns ($p_j > p_{j+1}$).

We assume that inspection is perfect, i.e., if we inspect a violation it will be caught with probability 1. Thus, if we inspect all a^t actions we would catch all v^t violations, i.e.,

$$\sum_{j=1}^{a^t} p_j = v^t .$$

Set $p_j = v^t \left(\frac{1-\beta}{1-\beta^{a^t}} \right) \beta^{j-1}$, where the parameter β could be any value in $(0, 1)$. Notice that

$$\sum_{j=1}^{a} p_j = v^t \left(\frac{1-\beta}{1-\beta^{a^t}} \right) \sum_{j=0}^{a^t-1} \beta^j = v^t \ ,$$

and $p_j > p_{j+1}$ so our model does satisfy diminishing returns. Furthermore, if $\beta = \max\{1 - \frac{1}{a^t}, \frac{1}{2}\}$ then we have $p_j \leq 1$ for each j. We can express $E\left[\mathbf{O}_{int}\right] = \sum_{i=1}^{s^t} p_j$.

$$E\left[\mathbf{L}^t(s)\right] = R - \left(R' \sum_{i=1}^{s^t} p_j + p \times v^t \times R_{ext} + C \times s \right) \ .$$

Example Loss Estimator: Our loss function estimator $\mathsf{est}\left(\mathbf{O}^t_{int}, s^t\right)$ is given in Algorithm 2.

Algorithm 2 Example: $\mathsf{est}\left(\mathbf{O}^t_{int}, s^t\right)$

```
• Input: $\mathbf{O}^t_{int}, s^t$.
• Estimate $v^t$: Set $\tilde{\mathbf{v}}^t := \frac{1-\beta^{a^t}}{1-\beta^{s^t}} \mathbf{O}^t_{int}$.
• Compute $\tilde{\mathbf{L}}$: Set
    $\tilde{\mathbf{L}}(x) := R - \left( \begin{array}{c} R' \times \tilde{\mathbf{v}}^t \times \sum_{j=1}^{x} \left( \frac{1-\beta}{1-\beta^a} \beta^{j-1} \right) \\ +p \times \tilde{\mathbf{v}}^t \times R_{ext} + C \times x \end{array} \right) .$
• Output: $\tilde{\mathbf{L}}^t$.
```

Assuming that the defender understands the accuracy of his recommendation algorithm REC[1] β is a known quantity so that this computation is feasible and can be performed quickly. Independence of the random variables $\mathbf{X}^t_s$ follows from the independence of $\mathbf{O}^t_{int}$. Now we verify that our estimator satisfies our accuracy condition.

Claim 1. *When $\tilde{\mathbf{L}}^t = \mathsf{est}\left(\mathbf{O}^t_{int}, s^t\right)$ from Algorithm 2 $E\left[\mathbf{X}^t_s\right] = 0$.*

The proof can be found in Appendix A. The main insight here is that because of the way the probabilities p_j's are scaled, the expectation of the estimated number of violations is equal to the number of actual violations, i.e. $E\left[\tilde{\mathbf{v}}^t\right] = v^t$. Consequently, the expected value of the error turns out to be 0, thus satisfying the accuracy property.

Remark 4. *If there are multiple types of actions then we can estimate $\tilde{\mathbf{v}}^t_{\mathbf{k}}$, the number of violations of type k at time t, separately for each k. To do this* est *should substitute $\vec{\mathbf{O}}^t_{int}[k]$ and $\vec{s}^t[k]$ for $\mathbf{O}^t_{int}$ and v^t and set*

$$\tilde{\mathbf{v}}^t_{\mathbf{k}} := \left(\frac{1-\beta^{a^t}}{1-\beta^{s^t}} \right) \mathbf{O}^t_{int} \ .$$

Then we have

$$\tilde{\mathbf{L}}^t(x) = R - \left(\begin{array}{c} \vec{R}' \cdot \langle \tilde{\mathbf{v}}^t_{\mathbf{k}} \rangle_k \times \frac{1-\beta^{x[k]}}{1-\beta^a} \\ +p \times \langle \tilde{\mathbf{v}}^t_{\mathbf{k}} \rangle_k \cdot \vec{R}_{ext} + \vec{C} \cdot \vec{x} \end{array} \right) \ .$$

VI. Proof Outline

In this section, we present the outline of the proof of our main theorem (Theorem 1) which establishes high probability regret bounds for RMA. The complete proof is in the appendices. The proof proceeds in two steps:

1) We first prove that RMA achieves low regret with respect to the estimated loss function using standard results from the literature on regret minimization [5], [11].
2) We then prove that with high probability the difference between regret with respect to the actual loss function and regret with respect to the estimated loss function is small. This step makes use of the two properties—accuracy and independence—of the estimated loss function est presented in the previous section and is the novel part of the proof. The key technique used in this step are the Hoeffding inequalities [16].

Let T denote the total number of rounds played, $T_{\vec{s}}$ denote the total number of rounds that the action $\vec{s}$ was awake and I as before is a time selector function. We define

$$\widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, I\right) = \sum_{t=1}^{T} \sum_{\vec{s}} I(t) p^t_{\vec{s}} \tilde{\mathbf{L}}^t(\vec{s}) \ ,$$

to be our total estimated loss. Notice that $\widetilde{\mathbf{Loss}}$ is the same as $\mathbf{Loss}$ except that we replaced the actual loss function $\mathbf{L}^t$ with the estimated loss function $\tilde{\mathbf{L}}^t$. We define $\widetilde{\mathbf{Loss}}\left(\vec{s}, I\right)$ and $\widetilde{\mathbf{Regret}}\left(\mathsf{RMA}, \vec{s}\right)$ in a similar manner by using the estimated loss function.

[1] If the algorithm recommends actions uniformly at random, then the accuracy is certainly understood.

Lemma 1 and Lemma 2 below bound the regret with respect to the estimated loss function. They are based on standard results from the literature on regret minimization [5], [11]. We provide full proofs of these results in Appendix C.

Lemma 1. *For each expert $\vec{s}$ we have*

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \leq \frac{1}{L-1}T + 2L\ln N \ ,$$

where N is the total number of experts and γ, our learning parameter has been set to $\gamma = 1 - \frac{1}{L}$

Remark 5. *We would like to bound our average regret:*

$$\frac{\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s})}{T_{\vec{s}}} \ .$$

There is a trade-off here in the choice of L. If L is too large, then $L\ln N$ will be large. If L is too small, then $\frac{1}{L-1}T$ will be large. If we know T in advance, then we can tune our learning parameter γ to obtain the best bound by setting

$$L = \sqrt{\frac{T}{2\ln N}} + 1 \ .$$

After substituting this value for L in Lemma 1, we immediately obtain the following result:

Lemma 2. *For each expert $\vec{s}$ we have*

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \quad \leq \quad 2\sqrt{2T\ln N} + 2\ln N \ ,$$

where N is the total number of experts and learning parameter $\gamma = 1 - \sqrt{\frac{2\ln N}{T}}$.

Remark 6. *This shows that* RMA *can achieve low regret with respect to the estimated loss functions $\tilde{\mathbf{L}}^t$. This completes step* 1 *of the proof. We now move on to step 2.*

Notice that we can write our actual loss function $(\mathbf{Loss}\,(\mathsf{RMA}, I))$ in terms of our estimated loss function $\left(\widetilde{\mathbf{Loss}}\,(\mathsf{RMA}, I)\right)$ and $\mathbf{X}^t_{\vec{s}}$.

Fact 1.

$$\mathbf{Loss}\,(\mathsf{RMA}, I) = \widetilde{\mathbf{Loss}}\,(\mathsf{RMA}, I) + \sum_{t=1}^{T} I(t)\mathbf{X}^t_{\vec{s}} \ .$$

We know that $E\left[\mathbf{X}^t_{\vec{s}}\right] = 0$ (from the accuracy property of the loss estimation function) and that $\mathbf{X}^1_{\vec{s}}, \ldots \mathbf{X}^T_{\vec{s}}$ are independent (from the independence property), so we can apply the Hoeffding inequalities to bound $\sum_t \mathbf{X}^t_{\vec{s}}$ obtaining Lemma 3. Appendix B contains a description of the inequalities and the full proof of the following lemma.

Lemma 3.

$$\Pr\left[\exists\vec{s}, \mathbf{Regret}\,(\mathsf{RMA}, \vec{s}) - \widetilde{\mathbf{Regret}}\,(\mathsf{RMA}, \vec{s}) \geq 2K\right] \leq \epsilon \ ,$$

where $K = \sqrt{2T\ln\left(\frac{4N}{\epsilon}\right)}$.

After straightforward algebraic substitution we can obtain our main result in Theorem 1 by combining Lemma 3 with Lemma 2 (see Appendix D).

Observe that the optimal value of γ is dependent on T. But, it is conceivable that the time T for which the game is played is not known in advance. The following remark shows that we can overcome this problem by choosing a dynamic value for γ. This makes RMA usable in the real world.

Remark 7. *Even if we don't know T in advance we can tune γ dynamically using a technique from [17]. We set*

$$\gamma_t = \frac{1}{1-\alpha_t} \ ,$$

where

$$\alpha_t = \sqrt{2\frac{\ln N}{L^t - 1}} \ .$$

where L^t is the minimum loss of any expert till time t (calculated using the estimated loss). Before playing round t we recompute the weights $w^t_{\vec{s}}$, pretending that our learning parameter γ had been set to γ_t from the beginning i.e.

$$w^t_{\vec{s}} = \gamma_t^{\sum_{i=1}^{t} I_{\vec{s}}(t)\left(\tilde{\mathbf{L}}^t(\vec{s}) - \gamma_t\tilde{\mathbf{L}}^t(\mathsf{RMA})\right)} \ .$$

In this case our final guarantee (similar to Theorem 1) would be that:

$$\Pr\left[\begin{array}{cc} \exists\vec{s}, & \begin{array}{c} \frac{\mathbf{Regret}(\mathsf{RMA},\vec{s})}{T} \geq 2\sqrt{2\frac{\ln N}{T}} + \\ 2\sqrt{\frac{2\ln\left(\frac{4N}{\epsilon}\right)}{T}} + \frac{10}{T}\ln N + \\ \frac{4}{T}\left(\ln N\right)\left(\ln(1+T)\right) \end{array} \end{array}\right] \leq \epsilon \ .$$

VII. Related Work

A. Auditing in Computer Security

A line of work in computer security uses evidence recorded in audit logs to understand why access was granted and to revise access control policies if unintended accesses are detected [6], [18], [19]. In contrast, we use audits to detect violations of policies, such as those restricting information use to specified purposes, that cannot be enforced using access control mechanisms.

Cederquist et al. [20] present logical methods for enforcing a class of policies, which cannot be enforced using preventive access control mechanisms, based on evidence recorded on audit logs. The evidence demonstrating policy compliance is presented to the auditor in the form of a proof in a logic and can be checked mechanically. In contrast, our focus is on policies that cannot be mechanically enforced in their entirety, but require involvement of human auditors. In addition, the new challenges in our setting arises from the imperfect and repeated nature of audits.

Zhao et al. [21] recognize that rigid access control can cause loss in productivity in certain types of organizations. They propose an access control regime that allows all access requests, but marks accesses not permitted by the policy for posthoc audit coupled with punishments for violating policy. They assume that the utility function for the organization and the employees are known and use a single shot game to analyze the optimal behavior of the players. Our approach of using a permissive access control policy coupled with audits is a similar idea. However, we consider a worst-case adversary (employee) because we believe that it is difficult to identify the exact incentives of the employee. We further recognize that the repeated nature of interaction in audits is naturally modeled as a repeated game rather than a one-shot game. Finally, we restrict the amount of audit inspections because of budgetary constraints. Thus, our game model is significantly more realistic than the model of Zhao et al. [21].

Cheng et al. [22], [23] also start from the observation that rigid access control is not desirable in many contexts. They propose a risk-based access control approach. Specifically, they allocate a risk budget to each agent, estimate the risk of allowing an access request, and permit an agent to access a resource if she can pay for the estimated risk of access from her budget. Further, they use metaheuristics such as genetic programming to dynamically change the security policy, i.e. change the risk associated with accesses dynamically. We believe that the above mechanism mitigates the problem of rigid access control in settings such as IT security risk management, but is not directly applicable for privacy protection in settings such as hospitals where denying access based on privacy risks could have negative consequences on the quality of care. Our approach to the problem is fundamentally different: we use a form of risk-based auditing instead of risk-based access control. Also, genetic programming is a metaheuristic, which is known to perform well empirically, but does not have theoretical guarantees [24]. In contrast, we provide mechanisms with provable guarantees. Indeed an interesting topic for future work is to investigate the use of learning-theoretic techniques to dynamically adjust the risk associated with accesses in a principled manner.

Guts et al. [25] consider an orthogonal problem in this space. They provide a characterization of auditable properties and describe how to type-check protocols to guarantee that they log sufficient evidence to convince a judge that an auditable property was satisfied on a protocol run.

Garg et al. [26] present an algorithm that mechanically enforces objective parts of privacy policies like HIPAA based on evidence recorded in audit logs and outputs subjective predicates (such as beliefs) that have to be checked by human auditors. Combining the their algorithm with ours provides an end-to-end enforcement mechanism for policies of this form.

B. Regret Minimization

A regret minimization algorithm ALG is a randomized algorithm for playing in a repeated game. Our algorithm RMA is based on the weighted majority algorithm [10] for regret minimization. The weighted majority maintains weights $w_{\vec{s}}$ for each of the N fixed actions of the defender. $w_{\vec{s}}^t$ is the weight of the expert before round t has been played. The weights determine a probability distribution over actions, $p_{\vec{s}}^t$ denotes the probability of playing $\vec{s}$ at time t. In any given round the algorithm attempts to learn the optimal distribution over actions by increasing the weights of experts that performed better than its current distribution and decreasing the weights of experts that performed worse.

1) Sleeping Experts: In the setting of [10] all of the actions are available all of the time. However, we are working in the sleeping experts model where actions may not be available every round due to budget constraints. Informally, in the sleeping experts setting the regret of RMA with respect to a fixed action $\vec{s}$ in hindsight is the expected decrease in our total loss had we played $\vec{s}$ in each of the $T_{\vec{s}}$ rounds when $\vec{s}$ was available.

There are variations of the weighted majority algorithm that achieve low regret in the sleeping experts setting [5], [11]. These algorithms achieve average regret bounds:

$$\forall \vec{s}, \frac{\mathbf{Regret}\,(\mathsf{Alg}, \tilde{\mathsf{s}})}{T_{\vec{s}}} = O\left(\frac{\sqrt{T \log N}}{T_{\vec{s}}}\right) .$$

In fact RMA is very similar to these algorithms. However, we are interested in finding exact (not asymptotic) bounds. We also have to deal with the imperfect information in our game.

2) Imperfect Information: In order to update its weight after round t, the weighted majority algorithm needs to know the loss of ever available defender action $\vec{s}$. Formally, the algorithm needs to know $\mathbf{L}^t(\vec{s})$ for each $\vec{s} \in \mathsf{AWAKE}^t$. However, we only observe an outcome $\vec{\mathbf{O}}^t$, which allows us to compute

$$\mathbf{L}^t(\vec{s}^t) = \mathbf{R}(\vec{\mathbf{O}}^t) - \vec{C} \cdot \vec{s}^t,$$

the loss for the particular action $\vec{s}^t$ played by the defender at time t. There are several existing algorithms for regret minimization in games with imperfect information [9], [12]–[14]. For example, [9] provides an average regret bound of

$$\forall \vec{s}, \frac{\mathbf{Regret}(\mathsf{Alg}, \vec{s})}{T} = O\left(\frac{N^{1/3} \log N}{\sqrt[3]{T}}\right) .$$

It is acceptable to have $\log N$ in the numerator, but the $N^{1/3}$ term will make the algorithm impractical in our setting. The average regret still does tend to 0 as $T \to \infty$, but the rate of convergence is much slower compared to the case when only $\log N$ in present in the numerator. Other algorithms [12]–[14] improve this bound slightly, but we still have the $N^{1/3}$ term in the numerator. Furthermore, [9] assumes that each action $\vec{s}$ is available in every round. There are algorithms that deal with sleeping experts in repeated games with imperfect information, but the convergence bounds get even worse.

Regret minimization techniques have previously been applied in computer security by Barth et al. [27]. However, that paper addresses a different problem. They show that reactive security is not worse than proactive security in the long run. They propose a regret minimizing algorithm (reactive security) for allocation of budget in each round so that the attacker's "return on attack" does not differ much from the case when a fixed allocation (proactive security) is chosen. Their algorithm is not suitable for our audit setting due to imperfect information and sleeping experts. In their work, the defender learns the attack path played by the adversary after each round, and by extension has perfect knowledge of the loss function for that round. By contrast, RMA must work in the imperfect information setting (see section VII-B2). Also, their model considers unknown attack paths that get discovered over time. This is a special subcase of the sleeping experts setting, where an expert is awake in every round after she wakes up. They extend the multiplicative weight update algorithm [10] to handle the special case. In our setting experts may be available in one round and unavailable in next. RMA was designed to work in this more general setting.

VIII. Conclusion and Future Work

We presented a principled approach to audits in organizations, such as hospitals, with permissive access control regimes. We modeled the interaction between the defender (e.g., hospital auditors) and the adversary (e.g., hospital employees) as a repeated game. The model takes pragmatic considerations into account, and considers a powerful worst-case adversary. We formulate a desirable property of the audit mechanism in this model based on the concept of regret in learning theory and present an efficient audit mechanism that provably minimizes regret for the defender. This mechanism learns from experience to guide the defender's auditing efforts. The regret bound is significantly better than prior results in the learning literature. The stronger bound is important from a practical standpoint because it implies that the recommendations from the mechanism will converge faster to the best fixed auditing strategy for the defender.

There are several directions for future work. We plan to develop similar results with a weaker (but still realistic) adversary model. Specifically, the regret bounds guaranteed by our algorithm hold even if an adversary controls the actions of all the employees in a hospital. It is reasonable to believe that not all employees behave adversarially. We plan to consider an alternative model in which some employees are adversarial, some are selfish and others are well-behaved (cf. [28]). Such a model could enable us to develop audit mechanisms that provide better bounds on the organization's regret. We also plan to implement such an audit mechanism and evaluate its performance over real hospital audit logs.

References

[1] G. Hulme, "Steady Bleed: State of HealthCare Data Breaches," September 2010, InformationWeek.

[2] *HIPPA Enforcement*, 2010 (accessed November 19,2010). [Online]. Available: http://www.hhs.gov/ocr/privacy/hipaa/enforcement/index.html

[3] H. DeYoung, D. Garg, L. Jia, D. Kaynar, and A. Datta, "Experiences in the logical specification of the HIPAA and GLBA privacy laws," in *Proceedings of the 9th annual ACM Workshop on Privacy in the Electronic Society (WPES)*, 2010.

[4] US Congress, "Health Insurance Portability and Accountability Act of 1996, Privacy Rule," 45 CFR 164, 2002, available at http://www.access.gpo.gov/nara/cfr/waisidx_07/45cfr164_07.html.

[5] A. Blum and Y. Mansour, "Learning, regret minimization, and equilibria," *Algorithmic Game Theory*, pp. 79–102, 2007.

[6] B. W. Lampson, "Computer security in the real world," *IEEE Computer*, vol. 37, no. 6, pp. 37–46, 2004.

[7] D. J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigenbaum, J. A. Hendler, and G. J. Sussman, "Information accountability," *Commun. ACM*, vol. 51, no. 6, pp. 82–87, 2008.

[8] D. Fudenberg and J. Tirole, *Game theory.* MIT Press, 1991.

[9] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2003.

[10] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Inf. Comput.*, vol. 108, no. 2, pp. 212–261, 1994.

[11] A. Blum and Y. Mansour, "From external to internal regret," in *COLT*, 2005, pp. 621–636.

[12] V. Dani and T. Hayes, "Robbing the bandit: Less regret in online geometric optimization against an adaptive adversary," in *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*. ACM, 2006, p. 943.

[13] M. Zinkevich, M. Johanson, M. Bowling, and C. Piccione, "Regret minimization in games with incomplete information," *Advances in Neural Information Processing Systems*, vol. 20, pp. 1729–1736, 2008.

[14] B. Awerbuch and R. Kleinberg, "Online linear optimization and adaptive routing," *Journal of Computer and System Sciences*, vol. 74, no. 1, pp. 97–114, 2008.

[15] M. Feldman, C. H. Papadimitriou, J. Chuang, and I. Stoica, "Free-riding and whitewashing in peer-to-peer systems," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 1010–1019, 2006.

[16] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 13–30, 1963.

[17] P. Auer, N. Cesa-Bianchi, and C. Gentile, “Adaptive and self-confident on-line learning algorithms,” *Journal of Computer and System Sciences*, vol. 64, no. 1, pp. 48–75, 2002.

[18] J. A. Vaughan, L. Jia, K. Mazurak, and S. Zdancewic, “Evidence-based audit,” in *CSF*, 2008, pp. 177–191.

[19] L. Bauer, S. Garriss, and M. K. Reiter, “Detecting and resolving policy misconfigurations in access-control systems,” in *SACMAT*, 2008, pp. 185–194.

[20] J. G. Cederquist, R. Corin, M. A. C. Dekker, S. Etalle, J. I. den Hartog, and G. Lenzini, “Audit-based compliance control,” *Int. J. Inf. Sec.*, vol. 6, no. 2-3, pp. 133–151, 2007.

[21] X. Zhao and M. E. Johnson, “Access governance: Flexibility with escalation and audit,” in *HICSS*, 2010, pp. 1–13.

[22] P.-C. Cheng and P. Rohatgi, “IT Security as Risk Management: A Research Perspective,” *IBM Research Report*, vol. RC24529, April 2008.

[23] P.-C. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger, “Fuzzy Multi-Level Security : An Experiment on Quantified Risk-Adaptive Access Control,” in *Proceedings of the IEEE Symposium on Security and Privacy*, 2007.

[24] M. D. Vose, A. H. Wright, and J. E. Rowe, “Implicit parallelism,” in *IN GECCO (2003)*, 2003, pp. 1505–1517.

[25] N. Guts, C. Fournet, and F. Z. Nardelli, “Reliable evidence: Auditability by typing,” in *ESORICS*, 2009, pp. 168–183.

[26] D. Garg, L. Jia, and A. Datta, “Policy Monitoring over Evolving Audit Logs: A Logical Method for Privacy Policy Enforcement,” CMU, Tech. Rep. CMU-CyLab-11-002, January 2011.

[27] A. Barth, B. Rubinstein, M. Sundararajan, J. Mitchell, D. Song, and P. Bartlett, “A Learning-Based Approach to Reactive Security,” *Financial Cryptography and Data Security*, pp. 192–206, 2010.

[28] A. S. Aiyer, L. Alvisi, A. Clement, M. Dahlin, J.-P. Martin, and C. Porth, “Bar fault tolerance for cooperative services,” *SIGOPS Oper. Syst. Rev.*, vol. 39, pp. 45–58, October 2005. [Online]. Available: http://doi.acm.org/10.1145/1095809.1095816

Appendix

Our main goal is to prove our main theorem from section III-B, showing that our audit mechanism (RMA) achieves low regret with high probability.

First, we prove an orthogonal result in appendix A. We prove that our example estimator function from section V is accurate for the example loss function that we provided.

In appendix B we prove that Lemma 3 holds for *any* estimator function est satisfies the accuracy and independence properties outlined in section V. Therefore, with high probability the defender's actual regret will be close to his estimated regret.

In appendix C we review standard regret bounds from the literature on regret minimization [5], [11]. We prove that our algorithm achieves low regret with respect to our estimated loss function.

Finally, in appendix D we combine our results to prove our main theorem in section III-B. This theorem shows that, except with probability ϵ, RMA will achieve low regret.

A. Estimating Losses

Recall that our regret bounds for algorithm 1 depended on the *accuracy* of the loss function estimator est. We prove that our example estimator (Algorithm 2) from section V is accurate.

Reminder of Claim 1. *When* $\tilde{\mathbf{L}}^t = \mathsf{est}\left(\mathbf{O}_{int}^t, s^t\right)$ *from algorithm 2*

$$E\left[\mathbf{X}_s^t\right] = 0 \ .$$

Proof: First observe that

$$\begin{aligned} E\left[\tilde{\mathbf{v}}^t\right] &= \frac{1-\beta^{a^t}}{1-\beta^{s^t}} E\left[\mathbf{O}_{int}^t\right] \\ &= \frac{1-\beta^{a^t}}{1-\beta^{s^t}} \sum_{i=1}^{s^t} p_j \\ &= \frac{1-\beta^{a^t}}{1-\beta^{s^t}} \sum_{i=1}^{s^t} v^t \frac{1-\beta}{1-\beta^{a^t}} \beta^{j-1} \\ &= \frac{1-\beta}{1-\beta^{s^t}} v^t \sum_{i=1}^{s^t} \beta^{j-1} \\ &= v^t \ . \end{aligned}$$

From which it follows that

$$\begin{aligned}
E\left[\mathbf{X}_s^t\right] &= E\left[\tilde{\mathbf{L}}^t(\vec{s})\right] - E\left[\mathbf{L}^t(\vec{s})\right] \\
&= R - R'\sum_{i=1}^{x} p_j - p \times v^t \times R_{ext} \\
&\quad -C \times s - E\left[\mathbf{L}^t(\vec{s})\right] \\
&= R - R'\sum_{i=1}^{x} p_j - p \times v^t \times R_{ext} - C \times s \\
&\quad -R + R' \times E\left[\tilde{\mathbf{v}}^t\right] \times \sum_{j=1}^{x}\left(\frac{1-\beta}{1-\beta^{a}}\beta^{j-1}\right) \\
&\quad +p \times E\left[\tilde{\mathbf{v}}^t\right] \times R_{ext} + C \times s \\
&= R' \times E\left[\tilde{\mathbf{v}}^t\right] \times \sum_{j=1}^{x}\left(\frac{1-\beta}{1-\beta^{a^t}}\beta^{j-1}\right) - R'\sum_{i=1}^{x} p_j \\
&= \left(R' \times \sum_{j=1}^{x}\left(\frac{1-\beta}{1-\beta^{a^t}}\beta^{j-1}v\right)\right) - R'\sum_{i=1}^{x} p_j \\
&= \left(R' \times \sum_{j=1}^{x} p_j\right) - R'\sum_{i=1}^{x} p_j \\
&= 0\ .
\end{aligned}$$

■

B. Hoeffding Bounds

Hoeffding Bound [16] bounds the probability of the deviation of a sum of independent random variables from the mean of the sum of random variables. The statement of Hoeffding Bound is as follows: if $X_1, X_2, ..., X_n$ are independent real valued random variables and $a_i \le X_i \le b_i$, then for $t > 0$,

$$Pr\left[\left|\sum_{i=1}^{n} X_i - E\left[\sum_{i=1}^{n} X_i\right]\right| > t\right] \le 2\exp\left(\frac{-2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

Claim 2. *For every* $\vec{s}$

$$\Pr\left[\left|\mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right)\right| \ge K\right] \le \frac{\epsilon}{2N}\ ,$$

where $K = \sqrt{2T \ln \frac{4N}{\epsilon}}$.

Proof: Notice that we can rewrite

$$\mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right) = \sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{X}_{\vec{s}}^t\ .$$

By the independence property of of loss estimator est, the random variables $\mathbf{X}_{\vec{s}}^t$ are independent. By the accuracy property of our loss function estimator est we have

$$E\left[\sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{X}_{\vec{s}}^t\right] = 0\ .$$

By definition there are exactly $T_{\vec{s}}$ times when $I_{\vec{s}}(t) = 1$ so the sum contains $T_{\vec{s}}$ independent random variables. We also have $-1 \le \mathbf{X}_{\vec{s}}^t \le 1$ so we can apply Hoeffding Bounds directly to obtain.

$$\Pr\left[\left|\sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{X}_{\vec{s}}^t\right| \ge K\right] \le 2\exp\left(\frac{-2K^2}{2^2 \times T_{\vec{s}}}\right)\ .$$

Plugging in for K and using the fact that $T_{\vec{s}} \le T$,

$$\begin{aligned}
\Pr\left[\left|\sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{X}_{\vec{s}}^t\right| \ge K\right] &\le 2\exp\left(\frac{-T\ln\frac{4N}{\epsilon}}{T_{\vec{s}}}\right) \\
&\le 2\exp\left(-\ln\frac{4N}{\epsilon}\right) \\
&= \frac{\epsilon}{2N}\ .
\end{aligned}$$

■

Claim 3. *For every* $\vec{s}$

$$\Pr\left[\left|\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right)\right| \ge K\right] \le \frac{\epsilon}{2N}\ ,$$

where $K = \sqrt{2T \ln \frac{4N}{\epsilon}}$.

Proof: Notice that we can rewrite

$$\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right) = \sum_{t=1}^{T}\sum_{\vec{s}} I_{\vec{s}}(t)p_{\vec{s}}^t\mathbf{X}_{\vec{s}}^t\ .$$

Set $\mathbf{Y}^t = \sum_{\vec{s}} p_{\vec{s}}^t \mathbf{X}_{\vec{s}}^t$, and observe that the random variables $\mathbf{Y}^t$ are independent and that $\mathbf{Y}^t \in [-1, 1]$. Substituting we get

$$\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right) = \sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{Y}^t\ .$$

Applying Hoeffding Bounds we have

$$\Pr\left[\left|\sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{Y}^t\right| > K\right] \le 2\exp\left(\frac{-2K^2}{2^2 T_{\vec{s}}}\right)\ .$$

Set $K = \sqrt{2T \ln \frac{4N}{\epsilon}}$. Then,

$$\begin{aligned}
&\Pr\left[\left|\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right)\right| > K\right] \\
= &\Pr\left[\left|\sum_{t=1}^{T} I_{\vec{s}}(t)\mathbf{Y}^t\right| > K\right] \\
\le &2\exp\left(\frac{-2K^2}{2^2 T_{\vec{s}}}\right) \\
\le &2\exp\left(-\ln\frac{4N}{\epsilon}\right) \\
\le &\frac{\epsilon}{2N}\ .
\end{aligned}$$

■

Lemma 4. *Except with probability ϵ, for all $\vec{s}$ we have*

$$\left|\mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right)\right| < K \ ,$$

and

$$\left|\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right)\right| < K \ ,$$

where $K = \sqrt{2T \ln \frac{4N}{\epsilon}}$.

Proof: There are N fixed actions $\vec{s}$ and thus $2N$ total events. Applying the union bound to claims 2 and 3 yields the desired result immediately. ■

Claim 4. *Suppose that for all $\vec{s}$ we have*

$$\left|\mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right)\right| < K \ ,$$

and

$$\left|\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right)\right| < K \ ,$$

where $K = \sqrt{2T \ln \frac{4N}{\epsilon}}$. then for every $\vec{s}$ we have

$$\mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Regret}}\left(\mathsf{RMA}, \vec{s}\right) \leq 2K \ .$$

Proof: We use the definition of $\mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right)$ and $\widetilde{\mathbf{Regret}}$, and then apply Lemma 4.

$$\begin{aligned}
& \mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Regret}}\left(\mathsf{RMA}, \vec{s}\right) \\
= \ & \left(\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right)\right) \\
& - \left(\widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right)\right) \\
\leq \ & \left|\mathbf{Loss}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \vec{s}\right)\right| \\
& + \left|\mathbf{Loss}\left(\vec{s}, I_{\vec{s}}\right) - \widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right)\right| \\
\leq \ & 2K \ .
\end{aligned}$$

■

We are now ready to prove Lemma 3 from section VI.

Reminder of Lemma 3.

$$\Pr\left[\exists \vec{s}, \mathbf{Regret}\left(\mathsf{RMA}, \vec{s}\right) - \widetilde{\mathbf{Regret}}\left(\mathsf{RMA}, \vec{s}\right) \geq 2K\right] \leq \epsilon \ ,$$

where $K = \sqrt{2T \ln\left(\frac{4N}{\epsilon}\right)}$.

Proof: We combine claim 4 and Lemma 4. ■

C. Standard Regret Bounds

We prove upper bounds on our estimated $\widetilde{\mathbf{Regret}}$. The proof techniques used in this section are standard [5], [11]. We include them to be thorough. The following claims will be useful in our proofs.

Claim 5.

$$\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \tilde{\mathbf{L}}^t(\vec{s}) = \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \tilde{\mathbf{L}}^t(\mathsf{RMA}) \ .$$

Proof: We plug in the definition of $\tilde{\mathbf{L}}^t(\mathsf{RMA})$:

$$\begin{aligned}
\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \tilde{\mathbf{L}}^t(\mathsf{RMA}) &= \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \sum_{\vec{\sigma}} \tilde{\mathbf{L}}^t\left(\vec{\sigma}\right) \\
&= \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \sum_{\vec{\sigma}} p_{\vec{\sigma}}^t \tilde{\mathbf{L}}^t\left(\vec{\sigma}\right) \\
&= \sum_{\vec{\sigma}} \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t p_{\vec{\sigma}}^t \tilde{\mathbf{L}}^t\left(\vec{\sigma}\right) \\
&= \sum_{\vec{\sigma}} \tilde{\mathbf{L}}^t\left(\vec{\sigma}\right) \left(\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t p_{\vec{\sigma}}^t\right) \\
&= \sum_{\vec{\sigma}} \tilde{\mathbf{L}}^t\left(\vec{\sigma}\right) \left(w_{\vec{\sigma}}^t\right) \\
& \quad \text{Relabel } \vec{\sigma} \\
&= \sum_{\vec{s}} \tilde{\mathbf{L}}^t\left(\vec{s}\right) \left(w_{\vec{s}}^t\right) \ .
\end{aligned}$$

■

Claim 6. *For all times t,*

$$\sum_{\vec{s}} w_{\vec{s}}^t \leq N \ .$$

Proof: Initially, $w_{\vec{s}}^0 = 1$ so initially the claim holds,

$$\sum_{\vec{s}} w_{\vec{s}}^0 = N \ .$$

The sum of weights can only decrease. At time t we only update the weights for those experts $\vec{s} \in \mathsf{AWAKE}^t$.

$$\begin{aligned}
\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^{t+1} &= \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \gamma^{\tilde{\mathbf{L}}^t(\vec{s}) - \gamma \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})} \\
&= \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \gamma^{\tilde{\mathbf{L}}^t(\vec{s})} \gamma^{-\gamma \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})} \\
&\leq \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \left(1 - (1 - \gamma) \tilde{\mathbf{L}}^t(\vec{s})\right) \\
& \quad \left(1 + (1 - \gamma) \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})\right)
\end{aligned}$$

$$\begin{aligned}
\leq \ & \left(\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t\right) \\
& -(1 - \gamma) \left(\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \tilde{\mathbf{L}}^t(\vec{s})\right) \\
& +(1 - \gamma) \left(\sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \tilde{\mathbf{L}}^{\mathbf{t}}(\mathsf{RMA})\right) \\
& \text{Apply Claim 5} \\
\leq \ & \sum_{\vec{s} \in \mathsf{AWAKE}^t} w_{\vec{s}}^t \ ,
\end{aligned}$$

where we used the following two facts

Fact 2.

$$\forall \gamma, y \in [0,1], \gamma^{y} \leq 1-(1-\gamma)y \ ,$$

and

Fact 3.

$$\forall \gamma, y \in [0,1], \gamma^{-y} \leq 1+(1-\gamma)\frac{y}{\gamma} \ .$$

Therefore,

$$\begin{aligned}
\sum_{\vec{s}} w_{\vec{s}}^{t+1} &= \sum_{\vec{s}\notin \mathsf{AWAKE}^t} w_{\vec{s}}^{t+1} + \sum_{\vec{s}\in \mathsf{AWAKE}^t} w_{\vec{s}}^{t+1} \\
&= \sum_{\vec{s}\notin \mathsf{AWAKE}^t} w_{\vec{s}}^{t} + \sum_{\vec{s}\in \mathsf{AWAKE}^t} w_{\vec{s}}^{t+1} \\
&\leq \sum_{\vec{s}\notin \mathsf{AWAKE}^t} w_{\vec{s}}^{t} + \sum_{\vec{s}\in \mathsf{AWAKE}^t} w_{\vec{s}}^{t} \\
&\leq N \ .
\end{aligned}$$

■

Claim 7. *For each expert $\vec{s}$ we have*

$$\widetilde{\mathbf{Loss}}(\mathsf{RMA}, I_{\vec{s}}) \leq \frac{\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{\ln N}{\ln \frac{1}{\gamma}}}{\gamma} \ ,$$

where N is the total number of experts.

Proof: By assumption, $\tilde{\mathbf{L}}^t(\vec{s})$ is independent of $\vec{s}^t$ so we can think of $\tilde{\mathbf{L}}^t(\vec{s})$ as being fixed before the defender selects its action $\vec{s}^t$. Notice that for all times j we have

$$N \geq w_{\vec{s}}^{j} = \gamma^{\sum_{t=1}^{j} I_{\vec{s}}(t)\left(\tilde{\mathbf{L}}^t(\vec{s}) - \gamma \tilde{\mathbf{L}}^t(\mathsf{RMA})\right)} \ .$$

Taking $\log_{\frac{1}{\gamma}}$ we obtain:

$$\begin{aligned}
\log_{\frac{1}{\gamma}} N &= \frac{\ln N}{\ln \frac{1}{\gamma}} \\
&\geq -\sum_{t=1}^{j} \left(I_{\vec{s}}(t)\tilde{\mathbf{L}}^t\left(\vec{s}\right) - \gamma \tilde{\mathbf{L}}^t\left(\mathsf{RMA}\right)\right) \\
&= -\widetilde{\mathbf{Loss}}\left(\vec{s}, I_{\vec{s}}\right) + \gamma \widetilde{\mathbf{Loss}}\left(\mathsf{RMA}, \mathbf{I}_{\tilde{\mathbf{s}}}\right) \ .
\end{aligned}$$

Hence,

$$\widetilde{\mathbf{Loss}}(\mathsf{RMA}, I_{\vec{s}}) \leq \frac{\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{\ln N}{\ln \frac{1}{\gamma}}}{\gamma} \ .$$

■

We are now ready to prove Lemma 1.

Reminder of Lemma 1. *For each expert $\vec{s}$ we have*

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \leq \frac{1}{L-1}T + 2L \ln N \ .$$

where N is the total number of experts and γ, our learning parameter has been set to $\gamma = 1 - \frac{1}{L}$.

Proof: Set $\gamma = 1 - \frac{1}{L}$ and apply claim 7. We have

$$\widetilde{\mathbf{Loss}}(\mathsf{RMA}, I_{\vec{s}}) \leq \frac{L}{L-1}\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{L}{L-1}\frac{\ln N}{\ln \frac{L}{L-1}} \ .$$

Using the definition of $\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s})$ we get

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \leq \frac{1}{L-1}\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{L}{L-1}\frac{\ln N}{\ln \frac{L}{L-1}} \ .$$

We will use the following fact

Fact 4.

$$\frac{1}{L-1} < 2\ln\left(\frac{L}{L-1}\right) \ ,$$

to get

$$\begin{aligned}
\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) &\leq \frac{1}{L-1}\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{L}{L-1}\frac{\log N}{\log \frac{L}{L-1}} \\
&\leq \frac{1}{L-1}\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + \frac{L}{L-1}\frac{\log N}{\frac{1}{2L-2}} \\
&\leq \frac{1}{L-1}\widetilde{\mathbf{Loss}}(\vec{s}, I_{\vec{s}}) + 2L \log N \ .
\end{aligned}$$

■

Lemma 2 follows immediately from 1.

Reminder of Lemma 2. *For each expert $\vec{s}$ we have*

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \quad \leq \quad 2\sqrt{2T \ln N} + 2\ln N \ ,$$

where N is the total number of experts and where our learning parameter has been set to

$$\gamma = 1 - \sqrt{\frac{2\ln N}{T}} \ .$$

D. Main Theorem

We are finally ready to prove our main theorem from section III-B.

Reminder of Theorem 1. *For all $\epsilon > 0$,*

$$\Pr\left[\begin{array}{cc} \exists \vec{s}, & \begin{array}{c} \frac{\mathbf{Regret}(\mathsf{RMA}, \vec{s})}{T} \geq 2\sqrt{2\frac{\ln N}{T}} + \\ 2\sqrt{\frac{2\ln\left(\frac{4N}{\epsilon}\right)}{T}} + \frac{2}{T}\ln N \end{array} \end{array} \right] \leq \epsilon \ .$$

Proof: Lemma 2 tells us that for each expert $\vec{s}$ we have

$$\widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \leq 2\sqrt{2T \ln N} + 2\ln N \ .$$

and Lemma 3 tells us that except with probability ϵ, for all actions $\vec{s}$ we have

$$\mathbf{Regret}(\mathsf{RMA}, \vec{s}) - \widetilde{\mathbf{Regret}}(\mathsf{RMA}, \vec{s}) \leq 2\sqrt{2T \ln\left(\frac{4N}{\epsilon}\right)} \ .$$

Combining Lemma 2 with Lemma 3 we obtain the desired result. ■