A Decision Theoretic Approach to Data Leakage Prevention

Janusz Marecki
IBM T.J. Watson Research
1101 Kitchawan Road, Route 134
Yorktown Heights, NY 10598
email: marecki@us.ibm.com

Mudhakar Srivatsa
IBM T.J. Watson Research
1101 Kitchawan Road, Route 134
Yorktown Heights, NY 10598
email: msrivats@us.ibm.com

Pradeep Varakantham
Singapore Management University
Singapore, 207855
email: pradeepv@smu.edu.sg

***Abstract*—In both the commercial and defense sectors a compelling need is emerging for rapid, yet secure, dissemination of information. In this paper we address the threat of information leakage that often accompanies such information flows. We focus on domains with one information source (sender) and many information sinks (recipients) where: (i) sharing is mutually beneficial for the sender and the recipients, (ii) leaking a shared information is beneficial to the recipients but undesirable to the sender, and (iii) information sharing decisions of the sender are determined using imperfect monitoring of the (un)intended information leakage by the recipients. We make two key contributions in this context: First, we formulate data leakage prevention problems as Partially Observable Markov Decision Processes; we show how to encode one sample monitoring mechanism—digital watermarking—into our model. Second, we derive optimal information sharing strategies for the sender and optimal information leakage strategies for a rational-malicious recipient as a function of the efficacy of the monitoring mechanism. We believe that our approach offers a first of a kind solution for addressing complex information sharing problems under uncertainty.**

***Keywords*-data leakage prevention; digital watermarking; partially observable Markov decision processes;**

I. Introduction

In both the commercial and defense sectors a compelling need is emerging for rapid, yet secure, dissemination of information to the concerned actors. For example, in a commercial setting, the ability of multiple partners to come together, share sensitive business information and coordinate activities to rapidly respond to business opportunities is becoming a key driver for success. Similarly, in a military setting, traditional wars between armies of nation-states are being replaced by highly dynamic missions where teams of soldiers, strategists and support staff fight against elusive enemies that easily blend into the civilian population [1]. Securely disseminating mission critical tactical intelligence to the pertinent people in a timely manner will be a critical factor in a mission's success.

Within a single organization, it is possible to allow sharing of information while managing the risk of information disclosure by appropriately labeling (or classifying) information with its secrecy characteristics and performing an in-depth security assessment (including system characterization, threat and vulnerability identification, control analysis, likelihood determination and impact analysis [2]) of its systems and users to create controls necessary to protect information commensurate with its label. However, such an approach may not be viable for information sharing across organizations as one organization will typically not permit another to perform a security assessment of its internal systems, controls and people. In dynamic settings, where systems and processes evolve rapidly and there are transient needs for sharing tactical, time-sensitive information across organizational boundaries, a new approach of securing information flows is required.

In this paper we present a novel decision theoretic approach for securing such information flows by reducing the risk of data leakage. Our approach is designed to make optimal information sharing decisions based on only partial or imperfect monitoring data, while ensuring that the efficacy of our decisions degrades gracefully with that of the monitoring mechanism. We focus on information sharing domains first studied in [3] that involve one information source (sender) and K information sinks (recipients) under the following generalized settings: (i) Information sharing occurs over a fixed period of N decision epochs and is mutually beneficial for the sender and each of the recipients; (ii) In each decision epoch a sender can share only one information object (packet), with a chosen recipient[1]; (iii) Leaking a shared packet results in a positive reward for the recipient and a penalty to the sender; (iv) Sharing a packet is instantaneous and the recipient leaks (or not) a packet immediately upon receiving it; (v) Sender uses a monitoring mechanism to detect an (un)intended packet leakage by the recipients, and finally (vi) Subsequent sender actions (whether to share a packet and with whom) are determined using the imperfect observations made in (v). We remark that if the monitoring mechanism is non-existent or arbitrarily imperfect then the system can have two trivial solutions: (a) share everything if the reward for information sharing is more than the penalty of information leakage; and (b) share nothing otherwise. Hence, we will examine settings wherein the information sharing is encouraged even when the penalty for information leakage is higher than that of information sharing by using a monitoring mechanism with realistic imperfections.

[1]Note that this assumption merely states that the packets to be shared can be arranged in a serial order. By considering multiple copies of a packet one can model situations where a packet is to be shared with multiple recipients.

In arriving at solutions to such planning problems we develop our key contributions: First, we provide a first of a kind formulation of the complex information sharing problems discussed above by combining Partially Observable Markov Decision Processes (POMDPs) with digital watermarking, a monitoring mechanism for data leakage detection. Second, we derive the optimal information sharing strategies for the sender and the optimal information leakage strategies for a rational-malicious recipient as a function of the efficacy of the underlying monitoring mechanism. Finally, we analyze the thresholds on the efficacy of a monitoring system in order to encourage information sharing under imperfect monitoring conditions for various reward models.

II. Background

A. Secure Information Sharing

Recently, new approaches based on risk estimation and economic mechanisms have been proposed for enabling the sharing of information in uncertain environments [4], [5], [6]. These approaches are based on the idea that the sender constantly updates the estimate of the risk of information disclosure when providing information to a receiver based on the secrecy of the information to be divulged and the sender's estimate on the trustworthiness of the recipient. The sender then *charges* the recipient for this estimated risk. The recipient, in turn, can decide which type of information is most useful to him and *pay* (using its line of risk credit) only to access those pieces of information. Under the assumption that the line of risk credit or the risk available for purchase in the market is limited, an entity will be encouraged to be frugal with their amassed risk credits and consequently, reluctant to spend them unnecessarily. Since all information flows are charged against expected losses due to unauthorized disclosure and the amount of risk available is limited, an argument is made that the total information disclosure risk incurred by an organization is controlled. Our work complements past work on risk-based information sharing by considering uncertainty in detecting information disclosure as a first class entity in complex information sharing domains.

As an alternative to economic mechanisms, in order to encourage behavioral conformity in ad-hoc groups one can also employ incentive mechanisms which have received a lot of attention in recent years. To date, the goal of such works has been to either reward "good" behavior [7], [8], [9], or punish "bad" behavior [10], [11]. In [12] for example, entities exchange tokens as a means of charging for/rewarding service usage/provision. Entities which behave correctly and forward packets are rewarded with additional tokens which, in turn, may be spent on forwarding their own packets. However, these approaches also fail to model the uncertainty in detecting good/bad behavior when making appropriate reward/punishment decisions.

Other incentive mechanisms rely on reputation as a means of encouraging entities to behave correctly. Reputation systems, such as [13], [14], aim to encourage good behavior by maintaining a trust/reputation score for some subset of entities in a network. If the reputation value for an entity drops below a predefined threshold, then that entity is deemed to be misbehaving and packets from that entity may be probabilistically dropped until the entity starts to conform [15]. By contrast, punishment mechanisms, such as those found in [16], [17], typically focus on the permanent exclusion of misbehaving entity from the network. Much like reward-based schemes, punishment strategies typically rely on implementing a threshold scheme, where, once a specific (mis)trust value is reached, an entity may instigate a revocation procedure. Our approach differs from these works as it allows to incentivize good behavior even in uncertain domains.

B. Digital Watermarking

In this paper we focus on digital watermarking based monitoring mechanism to detect information leakage. Figures 1 and 2 show how digital watermarking works in a simple spatial domain (2-dimensional image): The main idea is to generate a watermark $W(x, y)$ using a *secret* key chosen by the sender such that $W(x, y)$ is indistinguishable from random noise for any entity that does not know the key (i.e., the recipients). The sender adds the watermark $W(x, y)$ to the information object (image) $I(x, y)$ before sharing it with the recipient(s). It is then hard for any recipient to guess the watermark $W(x, y)$ (and *subtract* it from the transformed image $I'(x, y)$); the sender on the other hand can easily extract and verify a watermark (because it knows the key).

The recipient may attempt to corrupt the information object (e.g., toggle a few bits in the image file) with the goal of erasing the watermark to avoid detection. We note that in a pathological scenario, a recipient may corrupt the entire information object, thereby successfully erasing the watermark completely. Fortunately, corrupting an information object devalues it and thus, in such scenario, the leaked information is worthless. In particular, robustness requirements of digital watermarks mandates that any attempt to remove or destroy the watermark should produce a remarkable degradation in data quality before the watermark is lost [18]. Thus, there is a clear trade-off between the extent of corruption (and the residual value of the corrupted information object) and the false positive/false negative probabilities of the watermark detection algorithm. In this paper we investigate these trade-offs. Specifically, we employ POMDPs to help the sender (information source) characterize strategies of information sharing (what to share with whom?) and understand the optimal corruption strategies for a malign recipient.

C. Partially Observable Markov Decision Processes

Partially Observable Markov Decision Processes (POMDPs) [19] are defined as follows: S is a finite set of discrete states of the process and A is a finite set of agent actions. The process starts in state $s_0 \in S$ and runs for N decision epochs. In particular, if the process is in state $s \in S$ in decision epoch $0 \leq n < N$, the agent controlling the process chooses an action $a \in A$ to be executed next. The agent then receives the immediate reward $R(s, a)$ while the process transitions

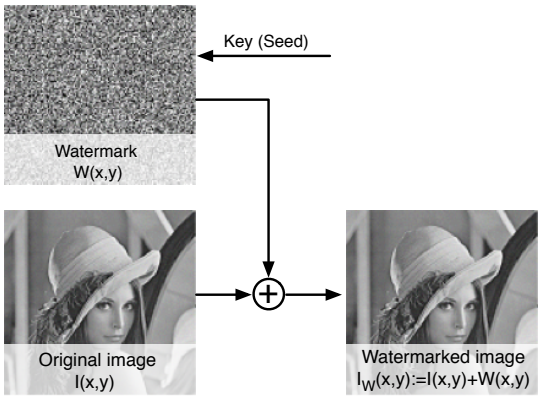

Fig. 1. Creating a watermark

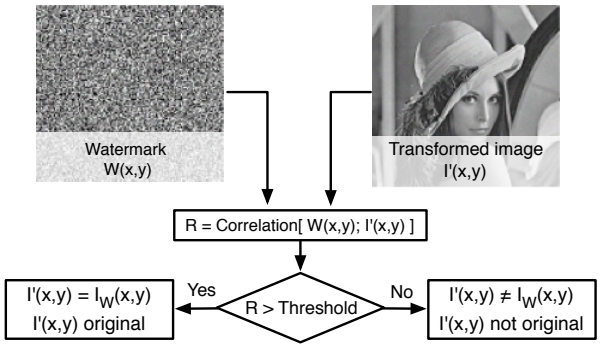

Fig. 2. Verifying a watermark

with probability $P(s'|a,s)$ to state $s' \in S$ and decision epoch $n+1$. Otherwise, if $n = N$, the process terminates. The goal of the agent is to find a policy π that, for each epoch $0 \leq n < N$, maximizes the sum of *expected* immediate rewards earned in epochs $n, n+1, \ldots, N$ when following policy π. What complicates the agent's search for π is that the process is only partially observable to the agent. That is, the agent receives noisy information about the current state $s \in S$ of the process and can therefore only maintain a probability distribution $b(s)$ over states $s \in S$ (referred to as the agent *belief state*). Specifically, when the agent executes an action $a \in A$ and the process transitions to state s', the agent receives with probability $O(z|a,s')$ an observation z from a finite set of observations Z. The agent then uses z to update its current belief state b, as shown in [19].

A policy π of the agent therefore indicates which action $\pi(n,b) \in A$ the agent should execute in decision epoch n in belief state b, for all $0 \leq n < N$ and all belief states b reachable from an initial belief state b_0 after n agent actions. To date, a number of efficient algorithms have been proposed to find a policy π^* that yields the maximum expected reward for the agent [20], [21], [22], [23], [24], [25], [26]. In our experiments we used a POMDP solver based on a point-based incremental pruning technique [21].

III. Data Leakage Prevention using POMDPs

As we now demonstrate, a systematic study of data leakage prevention domains of increased complexity allows to employ POMDPs to characterize optimal information sharing strategies for the sender and optimal watermark corruption strategies for a maligned recipient. We begin this study with a domain with a single, deterministic recipient (who either leaks out all the packets it receives or none of them). Next, we relax the assumption that the recipient is deterministic by considering a fuzzy recipient who leaks $f\%$ of the packets it receives. We finally generalize our models to domains where the sender shares information with multiple fuzzy recipients (each leaking a different percentage of packets it receives).

A. One Deterministic Recipient

The first type of data leakage prevention domains that we study involves a single information recipient (i.e., $K = 1$) who acts in a deterministic way (leaks either 0% or 100% of all the packets it receives). We model such domain using POMDPs as follows: The set of states is $S = \{s_0, s_{100}\}$ where s_0 denotes a state where the recipient leaks 0% of the packets it receives whereas s_{100} denotes a state where the recipient leaks 100% of the packets it receives. The set of sender actions is $A = \{a_{noShare}, a_{Share}\}$ where action $a_{noShare}$ results in the sender not sharing a packet with the recipient and a_{Share} in sharing exactly one packet with the recipient, in some decision epoch. We assume that the recipients never change the percentage of packets they leak out, and thus, the transition function is given by $P(s_0|a_{noShare}, s_0) = P(s_0|a_{Share}, s_0) = P(s_{100}|a_{noShare}, s_{100}) = P(s_{100}|a_{Share}, s_{100}) = 1$. The set of sender observations is $Z = \{z_{Leak}, z_{noLeak}, z_\emptyset\}$ where, according to z_{Leak}, the last-shared packet has been leaked and, according to z_{noLeak}, the last-shared packet has not been leaked. The sender receives an *empty* observation $z_\emptyset$ when it does not share a packet with the recipient. (Note, that because $z_\emptyset$ carries no information about the status of shared packets, it also does not affect the current sender estimate of the trustworthiness of the recipient. Also, because of the false positive/false negative observations, we may have $O(z_{Leak}|a_{Share}, s_0) > 0$ and $O(z_{Leak}|a_{Share}, s_{100}) < 1$.) Finally, we have $R(s_0, a_{noShare}) = R(s_{100}, a_{noShare}) = 0$ (not sharing a packet provides the sender with no reward/penalty) and $R(s_{100}, a_{Share}) < 0 < R(s_0, a_{Share})$ (sharing a packet is beneficial to the sender only if the packet is not leaked).

To illustrate a domain with a deterministic recipient on an example assume $N = 10$ decision epochs, rewards $R(s_0, a_{Share}) = 2, R(s_{100}, a_{Share}) = -1$, observation function $O(z_{Leak}|a_{Share}, s_0) = 10\%$, $O(z_{noLeak}|a_{Share}, s_0) = 1 - O(z_{Leak}|a_{Share}, s_0) = 90\%$, $O(z_{noLeak}|a_{Share}, s_{100}) = 30\%$, $O(z_{Leak}|a_{Share}, s_{100}) = 1 - O(z_{noLeak}|a_{Share}, s_{100}) = 70\%$, $O(z_\emptyset|a_{noShare}, s_0) = O(z_\emptyset|a_{noShare}, s_{100}) = 100\%$ and initial sender belief about the trustworthiness of the recipient $b_0(s_0) = b_0(s_{100}) = 50\%$. In such setting, the optimal policy of the sender yields the expected reward of 2.81. In Table I we show this policy for 5 selected action-observation scenarios and 4 initial decision epochs.

TABLE I
OPTIMAL SENDER POLICY FOR A DOMAIN WITH ONE DETERMINISTIC RECIPIENT

	Decision epoch 1		Decision epoch 2		Decision epoch 3		Decision epoch 4
	Action	Observation	Action	Observation	Action	Observation	Action
scenario 1	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}
scenario 2	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}
scenario 3	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}	z_{noLeak}	a_{Share}
scenario 4	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}	z_{Leak}	$a_{noShare}$
scenario 5	a_{Share}	z_{Leak}	$a_{noShare}$	$z_{\emptyset}$	$a_{noShare}$	$z_{\emptyset}$	$a_{noShare}$

The policy in Table I confirms some of the early intuitions about the domain with just one deterministic recipient: First *(scenario 1)*, if the sender does not observe any leaked packets, it keeps sharing the packets with the recipient. Second *(scenario 2)*, if the sender does not observe any leaked packets in the $1st$ and $2nd$ decision epochs, it builds enough confidence of the trustworthiness of the recipient so that, when a packet is observed to be leaked out in $3rd$ decision epoch, the sender attributes this leakage to its imperfect observations and resumes sharing the packets with the recipient in the $4th$ decision epoch. Third, *(scenario 4)* if the sender does not observe any leaked packets in the $1st$ decision epoch, but observes two consecutive leaked packets in the $2nd$ and $3rd$ decision epochs, the sender confidence of the trustworthiness of the recipient drops below a threshold where the sender decides to stop sharing further packets with the recipient. Finally, *(scenario 5)* if the sender observes a leaked packet in the $1st$ decision epoch, it attributes this leakage to a malevolent recipient (rather than to an imperfect observation) and never attempts to resume sharing packets with the recipient. (Note that not sharing any packets with the recipient provides no further observations to the sender and thus, the sender confidence of the trustworthiness of the recipient will not change. However, by considering $P(s|a,s) < 1$ for some $a \in A$, $s \in S$ one can model a sender who is forgiving towards the recipient. The optimal policy of a forgiving sender might then include a series of $a_{noShare}$ actions preceding a a_{Share} action, so that the impact of the old observations on the current belief state is less significant.)

B. One Fuzzy Recipient

We now move on to study more complex domains where the recipient can be fuzzy, i.e. leaks $f\%$ of the packets it receives thus appearing (to the sender) to be benevolent in some decision epochs and malevolent in other decision epochs. In modeling a fuzzy recipient we must address a key issue: The *recipient fuzziness* f is never known to the sender, and can only be estimated by the sender, using the observations it receives. It is then required that the sender maintains a probability distribution over all possible recipient fuzziness levels, i.e., *a probability distribution over the probabilities* with which the recipient can leak the packets. Because the number of all possible recipient fuzziness levels is infinite ($f \in [0,1]$), one cannot use POMDPs to model a fuzzy recipient exactly (due to an infinite POMDP state-space and the corresponding infinite transition/observation/reward functions).

We circumvent the problem of having to consider an infinite number of possible recipient fuzziness levels by approximating the actual (unknown) recipient fuzziness level f within some error ϵ with only a finite set M of chosen fuzziness levels. Precisely, we choose M to contain $\lceil \frac{1}{2\epsilon} + 1 \rceil$ uniformly distributed fuzziness levels so that for any $f \in [0,1]$ there always exists some $m \in M$ where $|f - m| < \epsilon$. The set of POMDP states is then $S = \{s_m\}_{m \in M}$ where s_m is a state wherein the recipient leaks $m\%$ of the packets it receives. The set of sender actions and observations, $A = \{a_{noShare}, a_{Share}\}$ and $Z = \{z_{Leak}, z_{noLeak}, z_{\emptyset}\}$ respectively, are the same as for a deterministic recipient. Similarly, (assuming that the recipient never changes the percentage of packets it leaks) the transition function is defined as $P(s_m|a_{Share}, s_m) = P(s_m|a_{noShare}, s_m) = 1$ for all $s_m \in S$. In defining the sender observation and reward functions, one needs to use the extreme values of these functions for a deterministic recipient case (when the recipient leaks 0% and 100% of packets it receives). Specifically, if the process is in state $s_m \in S$ and the sender executes action a_{Share}, there is $m\%$ chance that the packet will be leaked and $(100-m)\%$ chance that the packet will not be leaked and thus, $R(s_m, a_{Share}) = \frac{m}{100} R(s_{100}, a_{Share}) + \frac{100-m}{100} R(s_0, a_{Share})$. Similarly, (recall that the sender detects a leak if the leak really occurred with probability $O(z_{Leak}|a_{Share}, s_{100})$ and, if the leak did not occur, with probability $O(z_{Leak}|a_{Share}, s_0)$) if the process is in state $s_m \in S$ and the sender executes action a_{Share}, it will observe a leak with probability $O(z_{Leak}|a_{Share}, s_m) = \frac{m}{100} O(z_{Leak}|a_{Share}, s_{100}) + \frac{100-m}{100} O(z_{Leak}|a_{Share}, s_0)$.

To illustrate a domain with a fuzzy recipient on an example assume that the recipient fuzziness f is approximated with a set of fuzziness levels $M = \{0\%, 33\%, 66\%, 100\%\}$. Also, let $N = 10$, $R(s_0, a_{Share}) = 2, R(s_{100}, a_{Share}) = -1$, $O(z_{Leak}|a_{Share}, s_0) = 10\%$, $O(z_{Leak}|a_{Share}, s_{100}) = 70\%$—exactly as in the deterministic recipient case. Similarly, let the initial belief state of the sender be uniform, i.e., $b_0(s_m) = 0.25$ for all $m \in M$. In such setting, the optimal policy of the sender yields the expected reward of 2.23. In Table II we show this policy for 7 selected action-observation scenarios and 4 initial decision epochs.

As can be seen, the optimal policy of the sender when facing a fuzzy recipient (Table II) differs from the optimal policy of the sender when the recipient is deterministic (Table I). Specifically, the sender is more tolerant of packet leaks (compare

TABLE II
OPTIMAL SENDER POLICY FOR A DOMAIN WITH ONE FUZZY RECIPIENT

	Decision epoch 1		Decision epoch 2		Decision epoch 3		Decision epoch 4
	Action	Observation	Action	Observation	Action	Observation	Action
scenario 1	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}
scenario 2	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}
scenario 3	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}	z_{noLeak}	a_{Share}
scenario 4	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	a_{Share}	z_{Leak}	$a_{noShare}$
scenario 5	a_{Share}	z_{Leak}	a_{Share}	z_{noLeak}	a_{Share}	z_{noLeak}	a_{Share}
scenario 6	a_{Share}	z_{Leak}	a_{Share}	z_{noLeak}	a_{Share}	z_{Leak}	$a_{noShare}$
scenario 7	a_{Share}	z_{Leak}	a_{Share}	z_{Leak}	$a_{noShare}$	$z_{\emptyset}$	$a_{noShare}$

scenarios 5 in Table I with *scenarios 5,6,7* in Table II): Even if a packet shared in the $1st$ decision epoch is *observed* to be leaked, the sender decides to share another packet in the $2nd$ decision epoch. This phenomenon occurs because, for a leak detection probability $O(z_{Leak}|a_{Share}, s_{100})$ approaching 100%, whenever the sender detects a leak of a deterministic recipient, the sender considers the recipient to be almost 100% non-trustworthy; In contrast, if the leak is caused by a fuzzy recipient (of fuzziness f), the sender knows that there is still a $(100 - f)\%$ chance that the recipient will not leak further packets. Increased sender tolerance of leaked packets has an impact on the expected reward of its optimal policy; it amounts to only 2.23 as opposed to 2.81 if the recipient is deterministic.

C. Multiple Recipients

We finally move on to investigate the most complex data leakage prevention domains wherein the sender shares packets with multiple recipients, each potentially leaking a different percentage of packets it receives. That is, we now consider situations where a sender can choose which recipient (if any) should receive a packet in each decision epoch. In modeling such domains involving $K > 1$ recipients we must first choose the accuracy with which we approximate the actual (unknown) fuzziness values of each of the K recipients. Specifically, we assume a set M_k of fuzziness levels that approximate the (unknown) fuzziness of recipient k for each recipient $k \in K$. (As shown below, sets M_k need not to be equal as the sender might desire higher accuracy in approximating the fuzziness of more important recipients.)

A POMDP for a domain with multiple recipients is then defined as follows: Let $m = (m_1, ..., m_K)$ be a vector such that $m_k \in M_k$ is the chance that recipient k leaks a packet it receives, for $k \in K$. The set of states is then $S = \{s_m\}_{m \in M_1 \times ... \times M_K}$. Because in each decision epoch the sender can share a packet with at most one recipient, the set of actions is $A = \{a_{noShare}, a_{Share(1)}, ..., a_{Share(K)}\}$ where $a_{Share(k)}$ is an action that the sender executes to share a packet with recipient k. When the process is in state s_m and the sender executes action $a_{Share(k)}$, the process transitions to the same state s_m (recipients' fuzziness values never change) with probability 1. The sender then gets reward $\mathcal{R}(s_m, a_{Share(k)}) \equiv R(s_{m_k}, a_{Share})$ where the latter term is the sender reward in a single recipient case, as defined earlier[2]. Finally, the set of observations $Z = \{z_{Leak}, z_{noLeak}, z_{\emptyset}\}$ is the same as in the one recipient case, because the last executed action uniquely identifies the recipient who affects the sender last observation. As such, the observation function only depends on the fuzziness of the recipient that the packet was sent to, and thus, $\mathcal{O}(z_{Leak}|a_{Share(k)}, s_m) \equiv O(z_{Leak}|a_{Share}, s_{m_k})$ where the latter term is the sender observation function in a single recipient case, as defined earlier.

To illustrate a domain with multiple recipients on an example assume $K = 2$ recipients whose fuzziness values are approximated with different accuracy, i.e., $M_1 = \{0\%, 100\%\}$ and $M_2 = \{0\%, 33\%, 66\%, 100\%\}$. Also, let $N = 10$, $R(s_0, a_{Share}) = 2$, $R(s_{100}, a_{Share}) = -1$, $O(z_{Leak}|a_{Share}, s_0) = 10\%$, $O(z_{Leak}|a_{Share}, s_{100}) = 70\%$—exactly as in a single recipient case. Similarly, let the initial belief state of the sender be uniform, i.e., $b_0(s_m) = 0.5 \cdot 0.25 = 0.125$ for all $m \in M_1 \times M_2$. In such setting, the optimal policy of the sender yields the expected reward of 3.27. In Table III we show this policy for 7 selected action-observation scenarios and 4 initial decision epochs.

As can be seen (*scenario 1*), the sender always starts its optimal policy by sharing a packet with recipient 1, because recipient 1 appears to the sender to be more predictable (its fuzziness is approximated with fewer fuzziness levels) and consequently, it is easier for the sender to identify the trustworthiness of recipient 1 than to identify the trustworthiness of recipient 2. Next (*scenario 2*), if the sender observes no leaks while sharing the packets with recipient 1 in the $1st$ and $2nd$ decision epoch, it builds enough confidence about the trustworthiness of recipient 1 so that, even if a leak is observed after sharing a packet with recipient 1 in the $3rd$ decision epoch, the sender attributes this leak to its imperfect observations and decides to resume sharing packets with recipient 1 in the $4th$ decision epoch. However, (*scenarios 3,4*) if the sender observes no leak while sharing a packet with recipient 1 in the $1st$ decision epoch, but observes a leak while sharing a packet with recipient 1 in the $2nd$ decision epoch, sender confidence about the trustworthiness of recipient 1 is too low and the sender decides to switch to sharing the packets

[2]The sender could vary the importance of sharing the packets with different recipients by assuming that different recipients offer different rewards for received packets.

TABLE III
OPTIMAL SENDER POLICY FOR A DOMAIN WITH MULTIPLE RECIPIENTS

	Decision epoch 1		Decision epoch 2		Decision epoch 3		Decision epoch 4
	Action	Observation	Action	Observation	Action	Observation	Action
scenario 1	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$
scenario 2	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{Leak}	$a_{Share(1)}$
scenario 3	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{Leak}	$a_{Share(2)}$	z_{noLeak}	$a_{Share(2)}$
scenario 4	$a_{Share(1)}$	z_{noLeak}	$a_{Share(1)}$	z_{Leak}	$a_{Share(2)}$	z_{Leak}	$a_{noShare}$
scenario 5	$a_{Share(1)}$	z_{Leak}	$a_{Share(2)}$	z_{noLeak}	$a_{Share(2)}$	z_{noLeak}	$a_{Share(2)}$
scenario 6	$a_{Share(1)}$	z_{Leak}	$a_{Share(2)}$	z_{noLeak}	$a_{Share(2)}$	z_{Leak}	$a_{Share(2)}$
scenario 7	$a_{Share(1)}$	z_{Leak}	$a_{Share(2)}$	z_{Leak}	$a_{noShare}$	$z_{\emptyset}$	$a_{noShare}$

with recipient 2. In particular, (*scenarios 4,7*) if recipient 2 is also observed to be leaking the packets, the sender decides to stop sharing the packets with the recipients. Note that the sender ability to choose a recipient to share a packet with results in an increased expected reward of its optimal policy (equal to 3.27 as opposed to 2.81 and 2.23 when $K = 1$).

D. Recipient Strategy

Our methods for computing the sender policy assume that the number of decision epochs and the sender observation function (the accuracy of the mechanism that examines a watermark to determine if a packet is leaked or not) are fixed and known to both parties. Yet, there may be situations where the recipient can try to remove the watermarks from the packets, in an attempt to disguise the packets it leaks. In these situations, recipient's tampering with the watermark has a direct impact on the sender observation function. While this may seem to complicate the sender decision making, we show in the following that this is not the case: If both the sender and the recipient are rational and if they both know the domain parameters, the recipient strategy (how much it tampers with watermarks to obfuscate sender observations) is predictable, allowing the sender to compute its optimal policy when facing such a recipient. Note that it is of clear interest to the recipient to tamper with the watermarks. If the recipient leaves the watermarks intact, each time it leaks a packet, the leak will be detected with 100% accuracy by the sender (who may consequently stop sharing the packets with the recipient). On the other hand, if the recipient completely prevents the sender from detecting a leak, the sender may have little incentive to even begin sharing the packets with the recipient. Exactly how much to corrupt the watermarks therefore constitutes a decision problem in itself that every rational recipient has to face.

To illustrate this decision problem on an example, recall the domain with a deterministic recipient introduced earlier. Refer to Figure 3. Each bar in the figure represents the expected reward of the optimal sender policy for a given number of decision epochs N, a chance of leak detection $O(z_{Leak}|a_{Share}, s_{100})$ and an initial belief about recipient trustworthiness $b_0(s_0) = b_0(s_{100}) = 50\%$. (For explanation purposes we assume no false negative observations: $O(z_{Leak}|a_{Share}, s_0) = 0$.) As can be seen, the expected reward of an optimal sender policy can be either greater

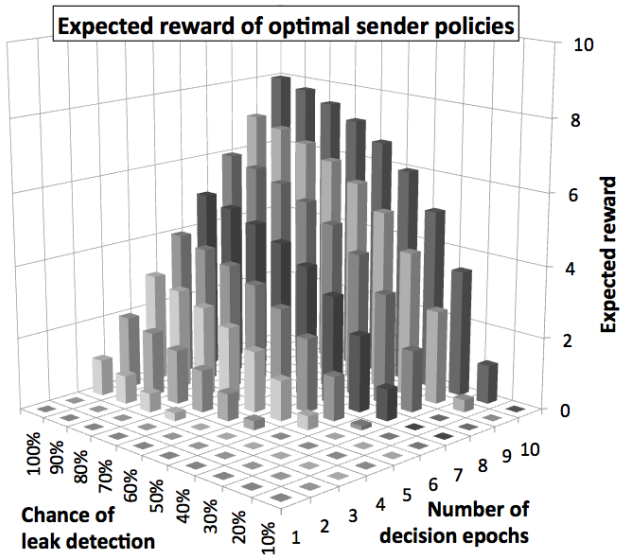

Fig. 3. Expected rewards of optimal sender policies

than zero, if the sender decides to share the packets with the recipient, or equal to zero, if the sender decides to not to share any packets with the recipient. Under these circumstances, the optimal strategy of a rational recipient will be to act in such a way that the chance of leak detection encourages the sender to share its packets, but provides the sender with as inaccurate information as possible about packet leaks. For example, if the number of decision epochs is 3, a rational recipient will allow the sender to detect a leak with 70% chance because that guarantees that the sender will start sharing the packets with the recipient but also ensures that sender observations will allow it to learn as little as possible about the packet leaks. A complete recipient strategy (Table IV) is hence a mapping from the number of decision epochs to the chance with which the recipient allows the sender to detect a leak. If both the sender and the recipient are rational and if they both know the domain parameters, their information sharing and watermark corruption strategies form a Nash-Equlibrium that both players will adhere to.

Finally, we note that to properly account for the strategic iterative decision making of the recipient (who may only possess partial information about the reward structure of

the sender), one would have to employ Partially Observable Stochastic Games [27] or Interactive POMDPs [28]. However, due to the N-EXP completeness of the exact algorithms for solving problems modeled in these frameworks [29], such an approach would be computationally prohibitive.

IV. Experiments

We have adopted the state-of-the-art POMDP solver [21] to conduct sensitivity and scalability analysis of our method applied to data leakage prevention. In our first experiment we investigated the sensitivity of the optimal sender policies to the changes in the chance of leak detection (Figure 4) in a deterministic recipient domain. We assumed $N = 10$ decision epochs, reward $R(s_0, a_{Share}) = 2$ for sharing a packet that is not leaked and an initial sender belief about the recipient trustworthiness $b_0(s_0) = b_0(s_{100}) = 50\%$. We then recorded the expected reward of optimal sender policies (y-axis) considering various leak costs $R(s_{100}, a_{Share}) = 0, -2, -4, -6$ and various chances $O(z_{Leak}|a_{Share}, s_{100})$ of leak detection (x-axis). Our results revealed that if the leak cost is 0, the chance of leak detection has no impact on the expected reward ($= 10$) of the optimal sender policy. This is because when the sender shares the packets with only one recipient and there is no penalty for leaked packets, sender optimal policy is to share the packets in all the decision epochs, regardless of the trustworthiness of the recipient—thus, regardless of its observations and the chance of leak detection. However, when the leak cost is other than 0, smaller chances of leak detection translate into higher chances of the sender deciding not to share the packets with the recipient and consequently, smaller expected rewards of the optimal sender policies. Furthermore, increase in the absolute value of the leak cost appears to amplify this phenomenon. For example, a decrease of the chance of leak detection from 100% to 50% corresponds to 12% decrease (from 9 to 8) of the expected reward if leak cost is -2, 25% decrease (from 8 to 6) if leak cost is -4 and as much as 43% decrease (from 7 to 4) is the leak cost is -6. We hence conclude that the greater the absolute value of the leak cost, the greater the sensibility of sender policy to the chance of leak detection.

The effect that the initial belief b_0 has on the expected reward of sender policies is orthogonal to that of the chance of leak detection, as revealed in our second experiment (Figure 5). Specifically, for a fixed chance $O(z_{Leak}|a_{Share}, s_{100}) = 80\%$ of leak detection, the greater the initial sender belief $b_0(s_0)$ about the trustworthiness of the recipient (x-axis), the higher the chance that the sender will share the packets with the recipient and consequently, the bigger the expected reward of the sender policy. Not surprisingly, if $b_0(s_0)$ drops below a certain threshold (e.g., 30% for the leak cost -4 and -6), the expected reward becomes 0 as it is not profitable for the sender to even start sharing packets with the recipient. On the other hand, the expected reward peaks at $b_0(s_0) = 100\%$ where it is invariant of the cost of leak and derived from $N \cdot R(s_0, a_{Share}) = 20$ (for $P(s_0|a_{Share}, s_0) = 1$). We hence

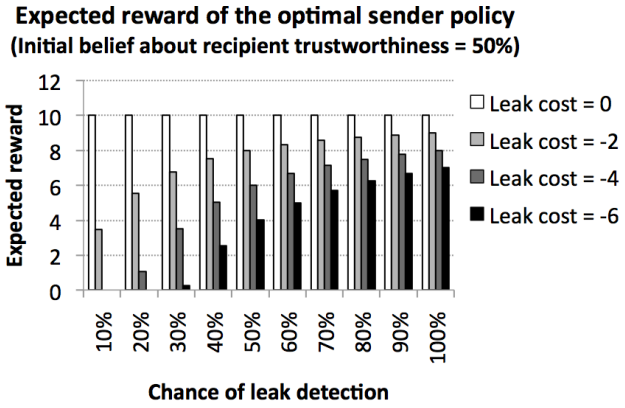

Fig. 4. Sensitivity of the expected reward of the optimal sender policies to different chances of leak detection

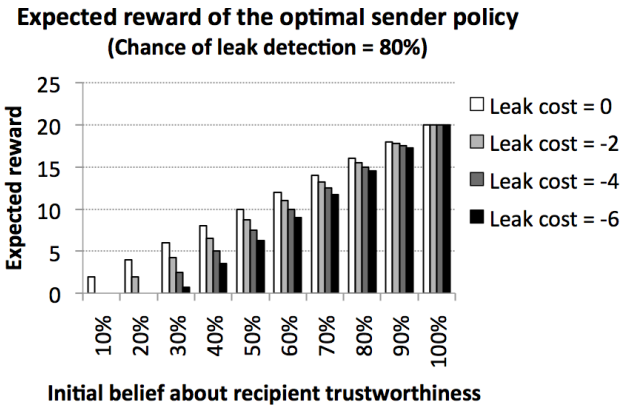

Fig. 5. Sensitivity of the expected reward of the optimal sender policies to different initial beliefs b_0

conclude that the greater the initial belief $b_0(s_0)$, the smaller the sensibility of the optimal sender policy to the leak cost.

In the final part of our experimental evaluation (refer to Table V) we recorded the runtimes of the POMDP solver employed to find the optimal sender policies (The solver was run on a 2.4 GHz machine with 2GB of RAM.) As expected, the runtime increases for higher number of epochs, higher number of recipients, and higher number of fuzziness levels (used to approximate a fuzzy recipient). The less dramatic increase seems to be related to the number of decision epochs, as the algorithm does not suffer severely from adding more epochs. However, increasing either the number of recipients or the number of fuzziness level results in a running time higher by almost an order of magnitude. Even though it is undeniable that some scalability issues arise here, the extent to which this might be a real concern in a *practical application* is unclear. As far as the number of recipients is concerned, we note that a group of recipients can always be modeled as a single one at the expense of losing some accuracy in our decisions (e.g., either we share with all of them or with none). In this way, applications where the number of recipients is high could be approached with an initial classification stage wherein recipients with similar characteristics are grouped

TABLE IV
OPTIMAL RECIPIENT STRATEGY (THE CHANCE WITH WHICH THE RECIPIENT ALLOWS THE SENDER TO DETECT A LEAK)

Number of decision epochs	1	2	3	4	5	6	7	8	9	10
Chance of leak detection	0%	0%	70%	50%	40%	30%	30%	30%	20%	20%

TABLE V
POMDP SOLVER RUNTIMES (IN MILLISECONDS)

	Number of decision epochs				
	2	4	6	8	10
	One recipient				
Fuzziness = 2	37	21	78	57	82
Fuzziness = 3	86	49	293	94	38
Fuzziness = 4	75	40	41	71	45
	Two recipients				
Fuzziness = 2	61	71	89	99	149
Fuzziness = 3	345	393	478	561	577
Fuzziness = 4	959	1045	1214	1397	1623
	Three recipients				
Fuzziness = 2	371	456	860	1064	1365
Fuzziness = 3	3851	4965	6215	8487	11060
Fuzziness = 4	20815	25996	32748	48473	64447
	Four recipients				
Fuzziness = 2	2083	3230	6532	11429	16316
Fuzziness = 3	45560	69745	93792	156302	249792
	Five recipients				
Fuzziness = 2	11744	19982	47398	96674	156288
Fuzziness = 3	631660	1156397	1638620	3180314	4587189

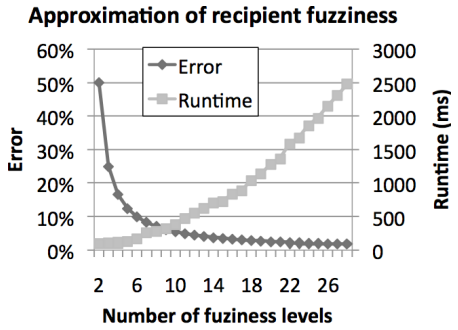

Fig. 6. Efficiency of the approximation of recipient fuzziness

into a single "class". (Note that this same philosophy has been applied in access control schemes for decades, where similar users are given the same security clearance.) As for the number of fuzziness levels, the trade-off here is clearly one of how much computational overhead we can afford against how accurately we want to approximate the recipient. Figure 6 illustrates graphically such a trade-off for a domain with one recipient and 10 decision epochs. For instance, if we are modeling a single recipient and the maximum amount of time we can afford is 500 ms, the best parameterization consists of using 12 levels, resulting in a 5% approximating error.

An interesting consequence of varying the number of decision epochs (in scenarios where this is possible) is that it affects not only performance (more epochs = higher expected reward, as seen in Figure 3), but also the receiver's optimal strategy. Previously we discussed how a rational recipient would see increased his possibilities of tampering with the watermarks (see Figure 3 and Table 4) depending on the number of epochs. It is easy to see that reducing the number of epochs increases the rate at which a recipient will start to leak out information. Nevertheless, we foresee that in many real applications the sender will not be at liberty of manipulating this parameter.

V. CONCLUSIONS

In both business and military applications an increase in demand is seen for solutions that allow for rapid yet secure sharing of information. Of particular need are solutions that view the information sharing as a sequential process where the trustworthiness of the information recipients is constantly monitored using data leakage detection mechanisms. Towards addressing this need, this paper has shown (i) how to formulate information sharing decisions using Partially Observable Markov Decision Processes combined with a digital watermarking leakage detection mechanism and (ii) how to derive optimal information sharing strategies for the sender and optimal information leakage strategies for a rational-malicious recipient as a function of the efficacy of the underlying monitoring mechanism. We have experimentally shown that the efficacy of our system degrades gracefully with the efficacy of the underlying monitoring mechanism and that the proposed system scales-up to realistic information sharing domains.

ACKNOWLEDGMENTS

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defense and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defense or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] D. Roberts, G. Lock, and D. Verma, "Holistan: A Futuristic Scenario for International Coalition Operations," in *In Proceedings of Fourth International Conference on Knowledge Systems for Coalition Operations (KSCO)*, 2007.

[2] G. Stoneburner, A. Goguen, and A. Feringa, "Risk Management Guide for Information Technology Systems," NIST, Tech. Rep. 800-300, 2002.

[3] M. Srivatsa, P. Rohatgi, S. Balfe, and S. Reidt, "Securing information flows: A metadata framework," in *Proceedings of 1st IEEE Workshop on Quality of Information for Sensor Networks (QoISN)*, 2008.

[4] P.-C. Cheng, P. Rohatgi, C. Keser, P. Karger, G. Wagner, and A. Reninger, "Fuzzy Multi-Level Security: An Experiment on Quantified Risk-Adaptive Access Control," in *Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP 2007)*, 2007, pp. 222–230.

[5] J. P. Office, "HORIZONTAL INTEGRATION: Broader Access Models for Realizing Information Dominance," MITRE Corporation, Special Report JSR-04-13, 2004.

[6] M. Srivatsa, S. Balfe, K. Paterson, and P. Rohatgi, "Trust Management for Secure Information Flows," in *Proceedings of 15th ACM Conference on Computer and Communication Security (CCS)*, 2008.

[7] G. Athanasiou, L. Tassiulas, and G. S. Yovanof, "Overcoming Misbehaviour in Mobile Ad Hoc Networks: An Overview," *Crossroads The ACM Student Magazine*, no. 114, pp. 23–30, 2005.

[8] M. Conti, E. Gregori, and G. Maselli, "Cooperation Issues in Mobile Ad Hoc Networks," in *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW 2004)*. IEEE Computer Society, 2004, pp. 803–808.

[9] S. Reidt, M. Srivatsa, and S. Balfe, "The Fable of the Bees: Incentivizing Robust Revocation Decision Making in Ad-Hoc Networks," in *Proceedings of 16th ACM Conference on Computer and Communication Security (CCS)*, 2009.

[10] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks," in *Proceedings of the 2003 IEEE Symposium on Security and Privacy (S&P 2003)*. IEEE Computer Society, May 2003, pp. 197–213.

[11] L. Eschenauer and V. Gligor, "A Key-Management Scheme for Distributed Sensor Networks," in *Proceedings of the 9th ACM conference on Computer and communications security (CCS 2002)*, 2002.

[12] L. Buttyán and J.-P. Hubaux, "Enforcing Service Availability in Mobile Ad-Hoc WANs," in *Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc 2000)*. IEEE Press, 2000, pp. 87–96.

[13] S. Buchegger and J.-Y. L. Boudec, "Self-Policing Mobile Ad Hoc Networks by Reputation Systems," *Communications Magazine, IEEE*, vol. 43, no. 7, pp. 101–107, 2005.

[14] P. Michiardi and R. Molva, "CORE: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks," in *IFIP TC6/TC11 6th Joint Working Conference on Communications and Multimedia Security*, ser. IFIP Conference Proceedings, vol. 228. Kluwer Academic, 2002, pp. 107–121.

[15] Q. He, D. Wu, and P. Khosla, "SORI: A Secure and Objective Reputation-Based Incentive Scheme for Ad-Hoc Networks," in *Proceedings of the 3rd IEEE Wireless Communications and Networking Conference, (WCNC 2004)*. IEEE Press, 2004, pp. 825–830.

[16] K. Hoeper and G. Gong, "Bootstrapping Security in Mobile Ad Hoc Networks Using Identity-Based Schemes with Key Revocation," Centre for Applied Cryptographic Research (CACR) at the University of Waterloo, Canada, Tech. Rep. CACR 2006-04, 2006.

[17] R. A. T. Moore, J. Clulow and S. Nagaraja, "New Strategies for Revocation in Ad-Hoc Networks," in *Proceedings of the 4th European Workshop on Security and Privacy in Ad Hoc and Sensor Networks (ESAS 2007)*. Springer, 2007, pp. 232–246.

[18] M. Barni, F. Bartolini, V. Cappellini, and A. Piva, "Image watermarking for secure transmission over public networks," 1995.

[19] E. J. Sondik, "The optimal control of partially observable Markov processes," in *Ph.D Thesis*. Stanford University, 1971.

[20] M. Hauskrecht, "Value-function approximations for POMDPs," *JAIR*, vol. 13, pp. 33–94, 2000.

[21] J. Pineau, G. Gordon, and S. Thrun, "PBVI: An anytime algorithm for POMDPs," in *IJCAI*, 2003, pp. 335–344.

[22] C. Poupart, P.; Boutilier, "VDCBPI: An approximate scalable algorithm for large scale POMDPs," in *NIPS*, vol. 17, 2004, pp. 1081–1088.

[23] Z. Feng and S. Zilberstein, "Region-based incremental pruning for POMDPs," in *UAI*, 2004, pp. 146–15.

[24] T. Smith and R. Simmons, "Point-based pomdp algorithms: Improved analysis and implementation," in *UAI*, 2005.

[25] M. T. J. Spaan and N. Vlassis, "Perseus: Randomized point-based value iteration for POMDPs," *JAIR*, vol. 24, pp. 195–220, 2005.

[26] P. Varakantham, R. Maheswaran, G. T., and M. Tambe, "Towards efficient computation of error bounded solutions in POMDPs: Expected value approximation and dynamic disjunctive beliefs," in *IJCAI*, 2007.

[27] R. Emery-Montemerlo, G. Gordon, J. Schneider, and S. Thrun, "Approximate solutions for partially observable stochastic games with common payoffs," in *AAMAS*, 2004.

[28] P. Gmytrasiewicz and P. Doshi, "A framework for sequential planning in multiagent settings," *Journal of Artificial Intelligence Research*, vol. 24, pp. 49–79, 2005.

[29] D. S. Bernstein, S. Zilberstein, and N. Immerman, "The complexity of decentralized control of MDPs," in *UAI*, 2000.