On the Inducibility of Stackelberg Equilibrium for Security Games

Qingyu Guo[1], Jiarui Gan[2], Fei Fang[3], Long Tran-Thanh[4], Milind Tambe[5], Bo An[1]
[1]School of Computer Science and Engineering, Nanyang Technological University, {qguo005, boan}@ntu.edu.sg
[2]Department of Computer Science, University of Oxford, Jiarui.gan@cs.ox.ac.uk
[3]School of Computer Science, Carnegie Mellon University, feifang@cmu.edu
[4]Department of Electronics and Computer Science, University of Southampton, ltt08r@ecs.soton.ac.uk
[5]Center for Artificial Intelligence in Society, University of Southern California, tambe@usc.edu

Abstract

Strong Stackelberg equilibrium (SSE) is the standard solution concept of Stackelberg security games. As opposed to the *weak Stackelberg equilibrium* (WSE), the SSE assumes that the follower breaks ties in favor of the leader and this is widely acknowledged and justified by the assertion that the defender can *often* induce the attacker to choose a preferred action by making an infinitesimal adjustment to her strategy. Unfortunately, in security games with resource assignment constraints, the assertion might not be valid; it is possible that the defender cannot induce the desired outcome. As a result, many results claimed in the literature may be overly optimistic. To remedy, we first formally define the utility guarantee of a defender strategy and provide examples to show that the utility of SSE can be higher than its utility guarantee. Second, inspired by the analysis of leader's payoff by Von Stengel and Zamir (2004), we provide the solution concept called the *inducible Stackelberg equilibrium* (ISE), which owns the highest utility guarantee and always exists. Third, we show the conditions when ISE coincides with SSE and the fact that in general case, SSE can be extremely worse with respect to utility guarantee. Moreover, introducing the ISE does not invalidate existing algorithmic results as the problem of computing an ISE polynomially reduces to that of computing an SSE. We also provide an algorithmic implementation for computing ISE, with which our experiments unveil the empirical advantage of the ISE over the SSE.

Introduction

The past few years have witnessed the huge success of game theoretic reasoning in the security domain (Tambe 2011; An 2017). Models based on the Stackelberg security game (SSG) have been deployed to protect high-profile infrastructures, natural resources, large public events, etc. (e.g., (Tsai et al. 2009; Jain et al. 2010; Yin, An, and Jain 2014; Fang et al. 2016; Basilico et al. 2017)). An SSG models the interaction between a defender and an attacker, where the defender commits to a mixed strategy first, and the attacker best responds with knowledge of the defender strategy.

The Stackelberg equilibrium is the standard solution concept for Stackelberg games (Leitmann 1978). In such an equilibrium, no player has the incentive to deviate and the leader assumes that deviations made will result in optimal responses of the follower when evaluating the benefit of deviations. The tie-breaking rules differentiates two forms of Stackelberg equilibria. The *strong* form of the Stackelberg equilibrium, called the *strong Stackelberg equilibrium* (SSE), assumes that the follower always breaks ties by choosing the best action for the defender, whereas its counterpart, the *weak Stackelberg equilibrium* (WSE), assumes that the follower always chooses the worst action. The SSE is commonly adopted as the standard solution concept because the WSE may not exist (Von Stengel and Zamir 2004); and the counter-intuitive tie-breaking rule is justified, implicitly or explicitly in the literature, by the assertion that the defender can often induce the favorable strong equilibrium by selecting a strategy arbitrarily close to the equilibrium.

Unfortunately, the assertion may break, especially in scenarios with various resource assignment constraints, such as scheduling constraints in the Federal Air Marshals Service (FAMS) domain, constraints on patrol paths for protecting ports, and constraints in the form of protection externalities (Tsai et al. 2009; Jain et al. 2010; Shieh et al. 2012; Gan, An, and Vorobeychik 2015). Most existing works failed to realize the potential impossibility to induce SSE in such domains. If the desired SSE cannot be induced, results claimed would questionably be overly optimistic. Such overoptimism is problematic in its own right and may even cause greater risks for the following reasons. First, these results may be used in making security resource acquisition decisions, i.e., what combination of security resources need to be procured (McCarthy et al. 2016); overoptimism of SSE may cause an insufficient number or wrong types of resources to be deployed. Second, statements made based on comparisons between the expected utility of SSE with some heuristic strategies or human-generated solutions to claim superiority of SSE strategies would be in potential jeopardy (Pita et al. 2008; Tsai et al. 2009; Xu et al. 2017). Third, the SSE strategy recommended may not be the optimal one, thus failing in optimizing the use of limited security resources, which is the primary mission of security games.

In this paper, we remedy the inadequacy of the SSE in security games and make the following key contributions. 1) We formalize the notion of overoptimism by defining the utility guarantee of the defender's strategies, and show with a motivating example that the utility claimed to be guaranteed by the SSE is much higher than the actually guar-

anteed utility. 2) Inspired by the notion of inducible strategy (Von Stengel and Zamir 2004), we characterize the solution concept with the highest utility guarantee and call it *inducible Stackelberg equilibrium* (ISE). 3) We compare ISE with SSE and show that for games with certain structures, the two concepts are equivalent, though in general cases the guaranteed utility of SSE can be arbitrarily worse than that of ISE; in addition, introducing the ISE does not invalidate existing algorithmic results as the problem of computing an ISE polynomially reduces to that of computing an SSE. 4) We provide algorithmic implementation for computing the ISE and conduct experiments to evaluate our results; our experiments unveil the significant overoptimism and suboptimality of the SSE, which suggests the practical significance of the ISE solution.

Other Related Works To the best of our knowledge, Okamoto, Hazon, and Sycara (2012) are the only exception who have raised the concern of lack of inducibility in security games, though their model is a very specific type of network security games that cannot be generalized to standard security games, especially games with scheduling constraints. Besides that, the more important question regarding the overoptimism due to the lack of inducibility and the algorithmic remedies needed for such overoptimism were left unanswered (in particular, the solution algorithm proposed by Okamoto, Hazon, and Sycara only converges to a local optimum even only in their setting). These questions are addressed in the affirmative in this paper. The concept of *inducible target* in our paper (Definition 2) is inspired by *inducible strategy* first proposed by von Stengel and Zamir (2004) in their study of general Stackelberg games. However, the focus of their work was solely on characterizing the range of leader's utility in Stackelberg equilibria with the aim of confirming the advantage of commitment (Von Stengel and Zamir 2004; von Stengel and Zamir 2010). Some other works considered potential deviation of the attacker from their optimal responses and proposed solution concepts that were robust to these deviations (Pita et al. 2009; Yang et al. 2014; Nguyen et al. 2013). Our work differs from this line of research in that we consider perfectly rational attackers.

Preliminaries

Security Games with Arbitrary Schedules

A security game is a two-player Stackelberg game played between an attacker and a defender. The defender allocates resources R to protect a set of targets T. Let $n = |T|$. A resource $r \in R$ can be assigned to a schedule $s \subseteq T$ which covers multiple targets and is chosen from a known and constrained set $S_r \subseteq 2^T$. The attacker's pure strategy is choosing one target $t \in T$ to attack, and his mixed strategy can be represented as a vector $\mathbf{a} \in \mathcal{A}$ where a_t denotes the probability of attacking $t \in T$. The defender's pure strategy is a joint schedule j which assigns each resource to at most one schedule. Let j be represented as a vector $\mathbf{P}_j = \langle P_{jt} \rangle \in \{0,1\}^n$ where P_{jt} indicates whether target t is covered in joint schedule j. The set of all feasible joint schedules is denoted by J. The defender's mixed strategy $\mathbf{x} \in \mathcal{X}$ is a vector where x_j denotes the probability of playing joint schedule j. Let $\mathbf{c} = \langle c_t \rangle$ be the coverage vector corresponding to $\mathbf{x}$, where $c_t = \sum_{j \in J} P_{jt} x_j$ is the marginal probability of covering t.

The payoffs of players are decided by the target chosen by the attacker and whether the target is protected by the defender. The defender's payoff for an uncovered attack is denoted by $U_d^u(t)$ and for a covered attack $U_d^c(t)$. Similarly, $U_a^u(t)$ and $U_a^c(t)$ are attacker's payoffs respectively. A widely adopted assumption in security games is that $U_d^c(t) > U_d^u(t)$ and $U_a^u(t) > U_a^c(t)$. In other words, covering an attack is beneficial for the defender, while hurts the attacker. Given a strategy profile $\langle \mathbf{x}, \mathbf{a} \rangle$, the expected utilities for both players are

$$U_d(\mathbf{x}, \mathbf{a}) = \sum\nolimits_{t \in T} a_t [c_t U_d^c(t) + (1 - c_t) U_d^u(t)]$$
$$U_a(\mathbf{x}, \mathbf{a}) = \sum\nolimits_{t \in T} a_t [c_t U_a^c(t) + (1 - c_t) U_a^u(t)],$$

where $\mathbf{c}$ is the coverage vector corresponding to $\mathbf{x}$. Let $U_a(\mathbf{x}, t)$ and $U_d(\mathbf{x}, t)$ denote the expected utilities of the attacker and defender respectively when t is attacked. The illustrated security game model has a wide applicability in many security applications (Kiekintveld et al. 2009; Jain et al. 2010; Tsai et al. 2009; Gan, An, and Vorobeychik 2015).

Stackelberg Equilibria and Tie-Breaking Rules

In an SSG, the defender acts first by committing to a mixed strategy and the attacker moves after having observed the defender's commitment. The solution concept of Stackelberg games, called *Stackelberg equilibrium*, captures the outcome in which the defender's strategy is optimal, under the assumption that the attacker will always respond optimally to the strategy the defender plays (Leitmann 1978). A pair of strategies $\langle \mathbf{x}^*, f(\mathbf{x}^*) \rangle$ forms a Stackelberg equilibrium iff:

1. $f : \mathcal{X} \to \mathcal{A}$ is a best response function of the attacker, that satisfies: $U_a(\mathbf{x}, f(\mathbf{x})) \geq U_a(\mathbf{x}, \mathbf{a})$ for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{a} \in \mathcal{A}$;
2. $U_d(\mathbf{x}^*, f(\mathbf{x}^*)) \geq U_d(\mathbf{x}, f(\mathbf{x}))$ for all $\mathbf{x} \in \mathcal{X}$.

A tie represents a situation where multiple best response strategies exist for the attacker. Ties are not rare corner cases, but a fundamentally recurring situation in security games. To achieve maximal usage of defense resources, algorithms avoid allocating too many or too few resources to each target, and in most cases generate a tied solution (Paruchuri et al. 2008; Kiekintveld et al. 2009). Thus, a tie-breaking rule – how the attacker breaks ties – plays a central role in security games and is exploited to design efficient algorithms, such as ORIGAMI (Kiekintveld et al. 2009). Different tie-breaking rules lead to different Stackelberg equilibria. The *strong Stackelberg Equilibrium* (SSE) and the *weak Stackelberg Equilibrium* (WSE) are two prevailing solution concepts, defined respectively with the optimistic and pessimistic assumptions of the attacker's tie-breaking behavior:

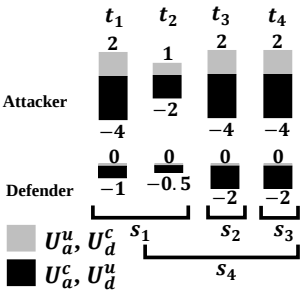
- **SSE**: $f^S(\mathbf{x}) \in \arg\max_{t \in \Gamma(\mathbf{x})} U_d(\mathbf{x}, t)$ for every $\mathbf{x}$;
- **WSE**: $f^W(\mathbf{x}) \in \arg\min_{t \in \Gamma(\mathbf{x})} U_d(\mathbf{x}, t)$ for every $\mathbf{x}$;

where $\Gamma(\mathbf{x}) = \arg\max_{t \in T} U_a(\mathbf{x}, t)$ is the *attack set*, the set of all best response pure strategies (targets) for attacker. In words, the attacker breaks ties in favor of the defender in the SSE, while against the defender in WSE.

WSE and SSE WSE follows the spirit of maximin solution (Sandholm 2015), which provides the defender a guaranteed value in the sense that if the attacker breaks ties in a different manner, the defender does not gain less. The SSE, however, does not provide such a value guarantee. Despite this, the security game literature has adopted SSE instead of WSE primarily because a WSE may not exist (Conitzer and Sandholm 2006). In addition, the counter-intuitive assumption that the attacker breaks ties in favor of the defender is justified by the assertion that the desired outcome can *often* be induced by playing a strategy arbitrarily close to the SSE strategy. Kiekintveld et al. (2009) were the first who explicitly made such a claim in the security game domain, following the analysis for generic Stackelberg games (Von Stengel and Zamir 2004). Since then, despite a lack of systematic research, the claim has been commonly used to support the SSE in security games of various types, including games with scheduling constraints (e.g., (Jain et al. 2010; Varakantham, Lau, and Yuan 2013; Gan, An, and Vorobeychik 2015)). The idea of SSE is also integrated in real world systems such as the ARMOR deployed at LAX (Pita et al. 2008), and IRIS for the Federal Air Marshal Services (Tsai et al. 2009). To see what can go wrong with the SSE assumption, we provide a concrete example in the next section.

Motivating Example

Consider an instance shown in the following figure where $T = \{t_1, t_2, t_3, t_4\}$. The defender has one resource $R = \{r\}$. We first consider the scenario without resource assignment constraints, which has a unique SSE with coverage $\mathbf{c} = \langle \frac{4}{15}, \frac{1}{5}, \frac{4}{15}, \frac{4}{15} \rangle$. In SSE, the attacker will break the tie $\Gamma(\mathbf{c}) = T$ by attacking t_2. This can be induced by decreasing the coverage on t_2 with infinitesimal amount and increasing the coverage on other targets, making t_2 be strictly preferred.



However, with resource assignment constraints, the defender cannot decrease the coverage on one target arbitrarily while simultaneously not decreasing coverage on all other targets. Suppose joint schedules $J = \{s_1, s_2, s_3, s_4\}$ as shown in the figure. (There is only one resource.) The game still has a unique SSE where the defender plays $\mathbf{x} = \langle \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0 \rangle$ and the attacker is assumed to attack t_2, bringing the defender an expected utility of $-\frac{1}{3}$. Such outcome is explicitly or implicitly considered with previous mentioned infinitesimal strategy deviation in security game literature (Jain et al. 2010). Unfortunately, there exists no strategy arbitrarily close to $\mathbf{x}$ which makes t_2 be strictly preferred by the attacker. If x_1 is decreased, the attacker will prefer t_1 over t_2; otherwise t_3 or t_4 will be attacked. Thus, any infinitesimal strategy deviation will cause the attacker to attack t_1, t_3 or t_4. The best induced outcome for the defender is only approaching $-\frac{2}{3}$, achieved by decreasing x_1 with infinitesimal amount and the attacker is induced to attack t_1.

Can the defender do better than $-\frac{2}{3}$? The answer is yes. Consider the mixed strategy $\langle \frac{1}{2}, 0, 0, \frac{1}{2} \rangle$. The attack set is $\{t_1, t_3, t_4\}$ and the defender can induce the attacker to strictly prefer t_1 by playing $\langle \frac{1}{2} - \delta, 0, 0, \frac{1}{2} + \delta \rangle$ with infinitesimal δ. By doing this, the defender can guarantee an expected utility arbitrarily close to -0.5, better than $-\frac{2}{3}$. In fact, this is the best outcome that the defender can achieve with infinitesimal strategy deviation. Such optimal outcome is captured by the solution concept called *inducible Stackelberg equilibrium* (ISE), proposed in the following section.

Inducible Stackelberg Equilibrium

The above example reveals a failure of the attempt to induce the desired SSE outcome by playing a strategy arbitrarily close to the SSE strategy. It is natural to ask: Given any strategy $\mathbf{x}$, what is the best outcome inducible by playing strategies arbitrarily close to $\mathbf{x}$? Associated with such best outcome, which strategy is optimal? To answer these questions, inspired by the "pessimistic" view of the leader's payoff in Stackelberg games (Von Stengel and Zamir 2004), we define the *utility guarantee* of a defender strategy as the supremum of the worst-case expected utility that can be achieved by playing a strategy arbitrarily close to the measured one.

Definition 1 (Utility Guarantee). *The utility guarantee of a defender strategy* $\mathbf{x}$ *is defined as*

$$U^I(\mathbf{x}) = \limsup_{\mathbf{x}' \to \mathbf{x}} \min_{t \in \Gamma(\mathbf{x}')} U_d(\mathbf{x}', t) \tag{1}$$

The utility guarantee is well-defined since the limit superior always exists. It measures the inducibility of a defender strategy: $U^I(\mathbf{x})$ is the optimal outcome at $\mathbf{x}$ that is inducible via infinitesimal strategy deviation. The aforementioned assumption widely acknowledged in security games falsely claim that any SSE strategy $\mathbf{x}$ provides utility guarantee $U_d(\mathbf{x}, f^S(\mathbf{x}))$, i.e., $U^I(\mathbf{x}) = U_d(\mathbf{x}, f^S(\mathbf{x}))$. Therefore, we need to find the optimal strategy with respect to the utility guarantee. We notice that the optimal utility guarantee coincides with the "pessimistic" leader's payoff (Von Stengel and Zamir 2004) as follows:

$$\max_{\mathbf{x} \in \mathcal{X}} U^I(\mathbf{x}) = \sup_{\mathbf{x} \in \mathcal{X}} \min_{t \in \Gamma(\mathbf{x})} U_d(\mathbf{x}, t) \tag{2}$$

Inspired by the analysis of "pessimistic" leader's payoff (Von Stengel and Zamir 2004), we introduce several useful notions for defining ISE. The first is *inducible target*.

Definition 2 (Inducible Target). *A target* t *is inducible iff there exists at least one defender mixed strategy* $\mathbf{x} \in \mathcal{X}$ *such that* $\Gamma(\mathbf{x}) = \{t\}$.

Inducible target offers the defender a lower bound on the utility guarantee as $U^I(\mathbf{x}) \geq U_d(\mathbf{x}, t)$ holds for any inducible target t in $\Gamma(\mathbf{x})$. The intuition is as follows. Since

t is inducible, there exists $\mathbf{x}'$ against which t is the unique best response for attacker. Thus, from $\mathbf{x}$, we can always play $(1-\alpha)\mathbf{x}+\alpha\mathbf{x}'$ with $\alpha \to 0$ which always makes t a unique best response as long as $\alpha > 0$. Then it is easy to verify that the supremum in (1) is always at least $U_d((1-\alpha)\mathbf{x}+\alpha\mathbf{x}', t)$, of which the limit is $U_d(\mathbf{x}, t)$.

The concept of inducible target is insufficient to fully characterize the utility guarantee of a strategy because a pair of targets might be indistinguishable from the attacker's perspective as they always bring the attacker the same utility irrespective of the strategy the defender plays. Such targets are called *identical targets*.

Definition 3 (**Identical Target**). *A pair of targets t and t' are identical iff $U_a(\mathbf{x}, t) = U_a(\mathbf{x}, t')$ for any $\mathbf{x} \in \mathcal{X}$.*

Identical targets are non-inducible by Definition 2. However, it is possible that the optimal utility guarantee in (1) is achieved via infinitesimal strategy deviation that induces a group of identical targets to be "unique" best responses. Thus, a generalized notion, *inducible element*, is defined to capture this special case. We begin with defining an *element*.

Definition 4 (**Element**). *An element is a set of targets in which: i) every pair of targets are identical, and ii) no target is identical to any target not in it.*

The reason that we call it an element is as follows. First, from the attacker's perspective, the *element* is the generalization on *target* as it characterizes the extend to which the attacker can distinguish from the perspective of payoffs. Second, with mild assumption which often holds true in practice, one can easily verify that two targets are identical iff they have same payoffs for attacker and they are covered by the same set of schedules. Thus it is easy to enumerate all possible elements. Let $\{t\}$ be a singleton element if no target in T is identical to t. The inducible element extends the concept of inducible targets as follows.

Definition 5 (**Inducible Element**). *An element e is inducible iff there exists at least one defender mixed strategy $\mathbf{x} \in \mathcal{X}$ such that $\Gamma(\mathbf{x}) = e$.*

The observation that inducible target offers a lower bound to utility guarantee extends to inducible element. To show this, we first define the utility function in an element-based manner. For a defender strategy $\mathbf{x}$, we define $\tilde{U}_d(\mathbf{x}, e)$ and $\tilde{U}_a(\mathbf{x}, e)$ as follows

$$\begin{aligned} \tilde{U}_d(\mathbf{x}, e) &= \min_{t \in e} U_d(\mathbf{x}, t) \\ \tilde{U}_a(\mathbf{x}, e) &= U_a(\mathbf{x}, t) \quad \forall t \in e. \end{aligned} \tag{3}$$

One key observation here is that, if e is inducible, $\tilde{U}_d(\mathbf{x}, e)$ lower bounds $U^I(\mathbf{x})$. This follows the similar explanation with inducible targets. Since e is inducible, there exists $\mathbf{x}'$ such that $\Gamma(\mathbf{x}') = e$ by definition and we can "perturb" $\mathbf{x}$ towards $\mathbf{x}'$ with infinitesimal amount and the attack set becomes exactly e. The observation follows as we notice that $\tilde{U}_d(\mathbf{x}, e)$ is a smooth function and thus the change on it with infinitesimal deviation on $\mathbf{x}$ is bounded.

With singleton element defined, the target set T is partitioned into a disjoint element set $\mathcal{E}$. It is easy to see that, for any defender strategy $\mathbf{x}$, the attack set $\Gamma(\mathbf{x})$ is always a union of some elements in $\mathcal{E}$. Thus, we define $\tilde{\Gamma}(\mathbf{x}) = \{e \in \mathcal{E} \mid e \subseteq \Gamma(\mathbf{x})\}$, and one can always verify that $\Gamma(\mathbf{x}) = \bigcup_{e \in \tilde{\Gamma}(\mathbf{x})} e$. $\tilde{\Gamma}(\mathbf{x})$ can be interpreted as an "attack set" consisting of elements, instead of targets. Let $\mathcal{E}^I \subseteq \mathcal{E}$ denote the set of inducible elements. The utility guarantee is actually decided by the inducible elements as presented in the following equation.

$$U^I(\mathbf{x}) = \max_{e \in \tilde{\Gamma}(\mathbf{x}) \cap \mathcal{E}^I} \tilde{U}_d(\mathbf{x}, e) \tag{4}$$

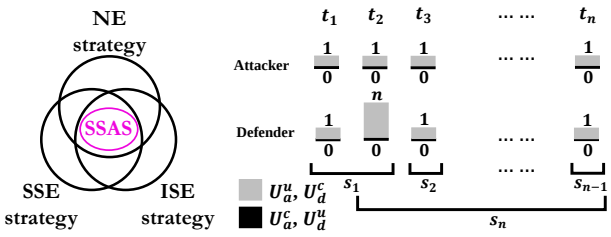
The correctness of this equation formally follows the analysis of "pessimistic" leader's payoff by Von Stengel and Zamir (2004), and here we provide an intuitive explanation. Since the players' utility functions are smooth, with infinitesimal strategy deviation, $U_d(\mathbf{x}, t)$ and $U_a(\mathbf{x}, t)$ can be regarded as unchanged for any t. Besides, with infinitesimal strategy deviation, it is only possible for defender to "remove" some target out of the attack set, while it is unable for the defender to add a new target into the attack set given the non-zero gap between attacker's utilities between targets inside and outside the attack set respectively. Thus, since the inducible outcome $U^I(\mathbf{x})$ is defined on the worst tie-breaking rule, i.e., $\min_{t \in \Gamma(\mathbf{x}')} U_d(\mathbf{x}', t)$ in (1), and infinitesimal strategy deviation won't change the defender's utility on any target, the defender always has an intention to reduce the attack set via infinitesimal strategy deviation. It is then noticed that, any inducible element can be the attack set itself with infinitesimal strategy deviation as we shown before, while any element, that is not inducible, cannot be the unique best response element. Besides, the definition of element determines that if one target from element e is in the attack set, so as all targets from e. Thus, the defender can only get $\min_{t \in e} U_d(\mathbf{x}, t)$ under worst-case tie-breaking rule, when $e \in \tilde{\Gamma}(\mathbf{x}) \cap \mathcal{E}^I$ becomes the unique best response element with infinitesimal strategy deviation.

To this end, we characterize the inducible outcome with well-defined concept of inducible elements, and we are ready to define the concept of inducible Stackelberg equilibrium, which straightforwardly follows the previous analysis

Definition 6 (**ISE**). *A pair of strategies $\langle \mathbf{x}^*, f^I(\mathbf{x}^*) \rangle$ forms an ISE if the following holds:*

1. $\mathbf{x}^* \in \arg\max_{\mathbf{x} \in \mathcal{X}} U^I(\mathbf{x})$;
2. $f^I(\mathbf{x}) \in \arg\min_{t \in e(\mathbf{x})} U_d(\mathbf{x}, t)$ *where* $e(\mathbf{x}) \in \arg\max_{e \in \tilde{\Gamma}(\mathbf{x}) \cap \mathcal{E}^I} \tilde{U}_d(\mathbf{x}, e)$.

Tie-breaking rule f^I partially shares the property of f^S as the attacker breaks the ties of elements in favor of the defender. Meanwhile, it behaves as f^W when the attacker breaks the ties of targets from the same element. Notice that ISE successfully addresses the inducibility issue of SSE, and always exists by its definition. In the next section, we conduct extensive analysis to compare ISE with SSE.



(a) SSE, ISE and NE (b) SSE can be arbitrarily worse

Figure 1: ISE vs. SSE.

ISE vs. SSE

In this section, we show that when *Subsets of Schedules Are Schedules* (SSAS) (Korzhyk et al. 2011) is satisfied, ISE and SSE are equivalent under mild assumption. However, in general cases the utility guarantee of SSE can be much worse than that of ISE, and we present one such example.

Formally, SSAS states that $2^s \subseteq S$ for all $s \in S$. This can happen, for example, when the defender can choose to bypass arbitrary targets on their patrol route.

Theorem 7. *If SSAS is satisfied, every SSE $\langle \mathbf{x}^*, t^* \rangle$ such that $c_{t^*} > 0$ is also an ISE.*

Proof. It is easy to see that when SSAS is satisfied, no pair of targets are identical, as every target t covered by s is uniquely covered by $s' = \{t\}$. Therefore, $\mathcal{E} = \{\{t\} \mid \forall t \in T\}$. We then show that each singleton element $\{t\}$ is inducible. In other words, t is an inducible target. If $\Gamma(\mathbf{x}^*)$ contains only t^*, then t^* is inducible. If $\Gamma(\mathbf{x}^*)$ also contains other targets, since SSAS is satisfied, we can construct a defender strategy $\mathbf{x}'$ such that $x'_j = x^*_j$ for all supporting strategies j that does not contain t^*, and $x'_{j \setminus \{s\}} = x^*_j$ for all other j that contains s. Thus, the corresponding coverage c'_{t^*} strictly decreases while the coverage of other targets remain the same. As a result, the attacker will strictly prefers to attack t^*, so t^* is inducible. □

Under SSAS, the set of SSE strategies is also a subset of NE strategies (Korzhyk et al. 2011). This suggests the relationship between SSE, ISE and NE strategies illustrated in Figure 1(a). Notice that security games without schedules can be seen as ones with singleton schedules $S = T$, so that SSAS is satisfied trivially. Although SSAS is valid in many real scenarios, it is risky to regard it as being ubiquitous. For example, in the presence of protection externalities (Gan, An, and Vorobeychik 2015; Gan et al. 2017), the effect that a defense resource might protect a set of targets within a certain radius can hardly be confined to a specific subset; in FAMS tasks (Tsai et al. 2009; Jain et al. 2010), when air marshals are allocated to a row of connected flights, it is unrealistic to make them "jump" over only a subset of the schedule. Our example below shows that in general security games, SSE can be arbitrarily worse than ISE in terms of the utility guarantee.

Example 1. The example is shown in Figure 1(b). The defender has only one resource. One can verify, the SSE strategy uniformly allocates this resource on $s_1, ..., s_{n-1}$ in order to make t_2 be in the attack set. Unfortunately, t_2 is not inducible since t_2 is weakly dominated by t_1 for the attacker. Therefore, $U_d^I(\mathbf{x}) = \frac{1}{n-1}$. On the other hand, the ISE strategy uniformly assigns the resource on s_1 and s_n which together cover all the targets and $U_d^I(\mathbf{x}') = \frac{1}{2}$.

Example 2. Notice that, in previous examples shown in the paper, a lot of targets have equal payoffs. However, this is only for the convenience of exposition. It is possible that ISE is not SSE when all targets have unequal payoffs. For example, there are 4 targets t_1, t_2, t_3, and t_4. Payoffs for the attacker on a successful attack are 1, 2, 3, and 4, respectively, and on an unsuccessful attack are -1, -2, -3, and -8 respectively. Payoffs for the defender on preventing an attack are 1, 100, 2, and 30 respectively and for failing to cover the attacked target are -1, 0, -2, and -3 respectively. There are two schedules $s_1 = \{t_1, t_2, t_3\}$ and $s_2 = \{t_4\}$, and one resource available to the defender. We can easily verify that SSE strategy is $\mathbf{x}^{SSE} = \langle 0.5, 0.5 \rangle$ and attacker is assumed to attack t_2. However t_2 is not inducible, and ISE strategy is $\mathbf{x}^{ISE} = \langle 9/14, 5/14 \rangle$ and attacker is induced to attack t_4.

Computing an ISE

We have shown that ISE mitigates the inducibility risk of SSE which can cause extremely worse performance in utility guarantee. In this section, we show that, from a computational perspective, ISE does not complicate existing solution concepts as the problem of computing an ISE polynomially reduces to that of computing an SSE on the same class of schedules. Besides the theoretical result, a practical approach is also presented to compute an ISE.

A Polynomial-time Reduction to Computing an SSE

We start by defining the *feasibility* of a target. We say a target is *feasible* if there exists $\mathbf{x} \in \mathcal{X}$ such that $t \in \Gamma(\mathbf{x})$. We will refer to the problem of deciding if a target is feasible or not, the *feasibility problem*; and of deciding if a target is inducible or not, the *inducibility problem*. The feasibility and inducibility of an element follow the similar definitions. We first restrict the investigation to games without identical targets. The reduction is presented in Theorem 9, where a series of feasibility checks are incorporated as sub-procedures.

Lemma 8. *For any target t in security games, the inducibility problem reduces to the feasibility problem on games with the same class of schedules in polynomial time.*

Proof sketch. The intuition behind Lemma 8 is the observation that whenever $U_a(\mathbf{c}, t) > U_a(\mathbf{c}, t')$ for all $t' \neq t$, there is a lower bound δ of the gap, such that $U_a(\mathbf{c}, t) - U_a(\mathbf{c}, t') \geq \delta$ for all $t' \neq t$, and $\log \frac{1}{\delta}$ is bounded by a polynomial in the input size. Blending δ into the payoffs, we construct a new game such that t is inducible in original game if and only if t is feasible in the new constructed game. □

Theorem 9. *The problem of computing an ISE reduces to the problem of computing an SSE of games with the same class of schedules in polynomial time.*

Proof. An ISE can be computed in the following way:

1. Check inducibility of every targets and obtain T^I.

2. For each $t \in T^I$, solve $\max_{\mathbf{x}:t\in\Gamma(\mathbf{x})} U_d(\mathbf{x}, t)$, which yields the defender's optimal strategy under the constraint that t is an optimal response of the attacker.
3. Among all the solutions obtained above, find out the one with the highest defender utility. The corresponding target, t^* say, and the optimal defender strategy corresponding to t^* forms an ISE.

Specifically, in Step 1, the inducibility problem reduces to the feasibility problem by Lemma 8. The feasibility of t can further be decided by computing the SSE of a game in which defender's payoff parameters are modified to: $U_d^u(t') = 1$ and $U_d^c(t') = 2$ for all $t' \neq t$; and $U_d^u(t) = 3$ and $U_d^c(t) = 4$ (the attacker's payoffs remain the same as in the feasibility problem). In this game, even the penalty on t is strictly higher than the rewards on all the other targets, so the defender strictly prefers the attacker to choose t, irrespective of the coverage of the targets. Therefore, t is feasible if t is in the attacker's attack set in every SSE, so we can check whether this is true to decide the feasibility of t.

In Step 2, each of the optimizations can be solved, again, by computing the SSE of a game in which defender's payoff parameters are modified to: $U_d^u(t') = 1$ and $U_d^c(t') = 2$ for all $t' \neq t$; and $U_d^u(t) = 3$ and $U_d^c(t) = 4$ (the attacker's payoffs remain the same as in the *original game*). t is inducible and hence feasible in the original game, and the feasibility remains in modified game as the attacker's payoffs are the same. For the same reason above, an SSE must incorporate t in the attack set, so that $U_a(\mathbf{x}, t) \geq U_a(\mathbf{x}, t')\ \forall\, t' \neq t$ is satisfied. In addition, c_t is maximized in the solution, so the SSE is exactly a solution to the optimization in Step 2.

Therefore, an ISE is obtained via polynomially many calls to the computation of an SSE. This completes the proof. □

Dealing with Identical Targets

In the presence of identical targets, it is assumed that, for every inducible element, the target worst for the defender is to be chosen by the attacker. We keep Step 1 of the procedure in the proof of Theorem 9 by treating identical targets as one target, so that a target is inducible if at least one target identical to it is inducible (even though this target might not actually be induced). However, when we actually compute the defender's optimal strategy conditioned on a particular inducible target being attacked as in Step 2, we need additional constraints that require this target is worst for the defender among all targets that is identical to it, i.e.,

$$U_d(c_t, t) \leq U_d(c_{t'}, t'), \quad \forall\, t' \text{ identical to } t.$$

We convert the constraints to equivalent ones in the form of $U_a(\mathbf{x}, t) \geq U_a(\mathbf{x}, t')$ (as in Step 2) to finish the reduction.

Observe that $c_t = c_{t'}$ since t' and t are identical, so the above constraints are equivalent to

$$U_d(c_t, t) \leq U_d(c_t, t'), \quad \forall\, t' \text{ identical to } t,$$

which only involve a single variable c_t. Thus, the constraints effectively reduce to an inequality of the form $\alpha \leq c_t \leq \beta$, with two constants α and β. For $c_t \geq \alpha$, given the objective of the problem as maximizing $U_d(\mathbf{x}, t)$ (which increases with c_t), this part can be ignored: if the solution does not satisfy $c_t \geq \alpha$, that means there is no feasible solution satisfying $c_t \geq \alpha$; we simply skip t in Step 3. The second half, $c_t \leq \beta$, can be captured by modifying the attacker's payoff parameters of an arbitrary target t', that is identical to t, to $U_a^c(t') = U_a^u(t') = U_a(\beta, t)$, so that the constraint $U_a(c_t, t) \geq U_a(c_{t'}, t')$ is now equivalent to $c_t \leq \beta$ (this constraint is useless before the modification when it always holds that $U_a(c_t, t) = U_a(c_{t'}, t')$).

Algorithmic Implementation

As a theoretical result, the above reduction involves repeated calls of computing an SSE and therefore falls short on practical performance. We introduce a more concise practical approach to compute an ISE. We first limit our scope to the games without identical target. The extension to include identical targets into consideration is fairly straightforward.

First, inducibility of a target t can be decided using the following program: t is inducible iff the optimum $u^* > 0$.

$$\begin{aligned} \max_{\mathbf{x},u} \quad & u \\ \text{s.t.} \quad & U_a(\mathbf{x}, t) \geq U_a(\mathbf{x}, t') + u \quad \forall t' \neq t \\ & \sum\nolimits_{j\in J} x_j = 1 \end{aligned} \tag{5}$$

Solving the above program for each $t \in T$, we obtain the inducible target set T^I. By Proposition 10, the computation of an ISE further converts to computing an SSE of a game restricted to targets in T^I. There is a large body of research on designing algorithms for computing an SSE of security games with various types of schedules, such as ASPEN (Jain et al. 2010) and CLASPE (Gan, An, and Vorobeychik 2015). These algorithms can be applied directly.

Implied by Proposition 10 to compute the "pessimistic" leader's payoff (von Stengel and Zamir 2010), we can directly compute an SSE in a restricted game g' whose target set is the set of inducible targets in targeted game g, and map this SSE to ISE of game g.

Proposition 10. *For a security game $g = \langle T, R, S\rangle$, an SSE defender strategy of the game $g' = \langle T^I, R, S^I\rangle$ is an ISE strategy in g, where T^I is the inducible target set of g, and $S^I = \{s \cap T^I | s \in S\}$.*

When identical targets exist, we first enumerate all inducible elements $\mathcal{E}^I$ by solving optimization (5) with slight modification, by replacing the target t and utility function $U_\dagger(\mathbf{x}, t)$ with the element e and element-based utility function $\tilde{U}_\dagger(\mathbf{x}, e)$ defined in (3), for $\dagger \in \{a, d\}$. An ISE can be computed with the multi-LP approach (Conitzer and Sandholm 2006), where each LP corresponds to an inducible element $e \in \mathcal{E}^I$ as follows

$$\begin{aligned} \max_{\mathbf{x},u} \quad & u \\ \text{s.t.} \quad & \tilde{U}_a(\mathbf{x}, e) \geq \tilde{U}_a(\mathbf{x}, e') \qquad \forall e' \in \mathcal{E} \\ & u \leq U_d(\mathbf{x}, t) \qquad\qquad\quad \forall t \in e \\ & \sum\nolimits_{j\in J} x_j = 1 \end{aligned} \tag{6}$$

The solution with the highest objective among multiple LPs is an ISE. It can be easily verified that the large body of designing algorithms, especially those based on strategy generation techniques, can adapt to solve (6) with little effort.

Experimental Evaluation

We evaluate our solution concept and proposed algorithmic implementation with extensive experiments. All results are obtained on a platform with a 2.60 GHz dual-core CPU and 8.0 GB memory. All linear programs are solved using the existing solver CPLEX (version 12.4). The random instances are generated as follows: rewards and penalties are all integers randomly drawn from $[0, 5]$ and $[-5, 0]$ respectively. Each schedule is randomly generated covering a fixed number l of targets and each target is ensured to be covered by at least one schedule. The resources are all homogeneous, i.e., $S_r = S$ for any $r \in R$. Unless otherwise specified, all results are averaged on 100 randomly generated instances.

For the purpose of comparison, we define the *overoptimism* and *sub-optimality* of SSE w.r.t. the utility guarantee.

Definition 11 (Overoptimism and sub-optimality). *Let* $\mathbf{x}$ *be an SSE strategy.*

- $\mathbf{x}$ *is overoptimistic if* $U_d(\mathbf{x}, f^S(\mathbf{x})) > U_d^I(\mathbf{x})$*;*
- $\mathbf{x}$ *is sub-optimal if* $U_d^I(\mathbf{x}) < \max_{\mathbf{x}' \in \mathcal{X}} U_d^I(\mathbf{x}')$.

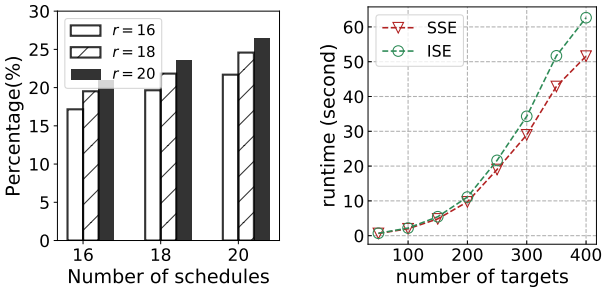


Figure 2: Inducibility (left) and scalability (right).

Inducibility We depict the percentage of inducible targets on instances with 100 targets and 1 resource on the left of Figure 2. The results show that with more schedules and more targets per schedule, the game has more inducible targets. That is because the defender can cover the high valued targets with enough resources so that the low valued targets can be induced to become unique best responses. The important observation here is that the percentage is neither too high nor too low (within $[15\%, 30\%]$), which indicates that the inducibility is not a trivial property.

Scalability We evaluate the scalability of our algorithmic implementation for computing an ISE. The result is shown on the right of Figure 2. The game instances are randomly generated with $l = 5$, $|R| = 5$, $|T|$ ranges from 50 to 400 with step size of 50, and $|S| = |T|/2$. We adopt the column generation approach with heuristic bounds to solve the large scale LPs (Gan, An, and Vorobeychik 2015). As a comparison, the scalability of computing SSE with the same algorithmic framework is also depicted. The result shows that, it takes almost the same computational costs to compute an ISE and an SSE. The algorithmic implementation can compute ISE for large-scale instances. Thus, ISE successfully mitigates the inducibility issue of SSE without sacrificing the benefit of scalable algorithms for computng SSE.

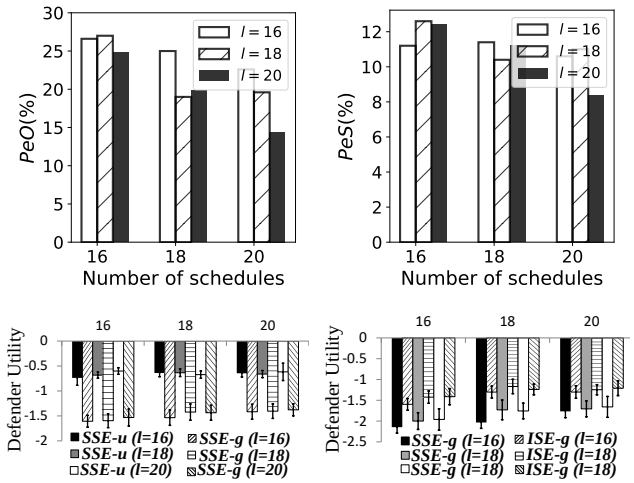


Figure 3: Overoptimism (left) and sub-optimality (right) of SSE.

Overoptimism and Sub-optimality of SSE We examine the overoptimism and sub-optimality of SSE. 500 instances are randomly generated with 200 targets, 1 resource, $|S| \in \{16, 18, 20\}$ and $l \in \{16, 18, 20\}$. This setting fits many realistic security domains, such as the port protection (Shieh et al. 2012), where the Coast Guard has few resources (patrol boats) and limited schedules due to complex geographic and efficiency constraints, and each schedule corresponds with one patrolling path visiting several targets. The results are shown in Figure 3, where *PeO* and *PeS* denote the percentages of instances with overoptimistic and sub-optimal SSE respectively. Moreover, Figure 3 also shows the comparisons between the expected utility of SSE ("SSE-u") with the utility guarantee of SSE ("SSE-g") averaged on instances where SSE is overoptimistic, and similarly the comparisons between average utility guarantees of SSE and ISE (tagged with "SSE-g" and "ISE-g" respectively) on instances, where SSE is suboptimal. The 95% confidence interval is depicted. The results show that SSE suffers from significant overoptimism and sub-optimality, which is highly problematic as we explained in the introduction. We also conduct simulations in a large number of different parameter settings with 3 and more resources. Here we list the results on ten settings in the table on the right. For each of these settings, we randomly generate 50 instances. Significant numbers of cases with overoptimistic and suboptimal SSE are observed for almost every setting. Thus, the aforementioned risk of applying SSE in practice can be a general issue for many security domains and applications, and we argue that ISE should be considered as a "safer" alternative.

$\|T\|$	$\|R\|$	$\|S\|$	l	PeO	PeS
400	3	24	24	22%	6%
400	5	26	26	18%	8%
400	5	26	24	26%	14%
400	5	24	28	18%	6%
600	3	20	40	32%	10%
600	3	22	42	24%	6%
600	3	18	38	38%	4%
800	3	30	40	24%	6%
800	3	28	38	26%	10%
800	3	28	40	26%	6%

Conclusion

This paper reveals the significant potential risk of overoptimism of SSE in security games. We propose a new solution concept, ISE, by exploiting the inducible targets. Our theoretical analysis proves the existence of ISE and its optimality in utility guarantee, and our formal comparisons between ISE and SSE emphasize that ISE is a more suitable solution concept in security games. Extensive evaluation shows that SSE is significantly overoptimistic and ISE achieves significantly higher utility guarantee than SSE. We will investigate the inducibility issues in generic games and Bayesian games in future work.

Acknowledgments

This research was supported by Singapore NRF, MOE, NTU, and MURI Grant W911NF-11-1-0332. Jiarui Gan is supported by the EPSRC International Doctoral Scholars Grant EP/N509711/1. Tran-Thanh Long was supported by the EPSRC funded project EP/N02026X/.

References

An, B. 2017. Game theoretic analysis of security and sustainability. In *IJCAI*, 5111–5115.

Basilico, N.; Celli, A.; Nittis, G. D.; and Gatti, N. 2017. Coordinating multiple defensive resources in patrolling games with alarm systems. In *AAMAS*, 678–686.

Conitzer, V., and Sandholm, T. 2006. Computing the optimal strategy to commit to. In *EC*, 82–90.

Fang, F.; Nguyen, T. H.; Pickles, R.; Lam, W. Y.; Clements, G. R.; An, B.; Singh, A.; Tambe, M.; and Lemieux, A. 2016. Deploying PAWS: Field optimization of the protection assistant for wildlife security. In *IAAI*, 3966–3973.

Gan, J.; An, B.; and Vorobeychik, Y. 2015. Security games with protection externalities. In *AAAI*, 914–920.

Gan, J.; An, B.; Vorobeychik, Y.; and Gauch, B. 2017. Security games on a plane. In *AAAI*, 530–536.

Hohn, F. E. 2013. *Elementary matrix algebra*. Courier Corporation.

Jain, M.; Kardes, E.; Kiekintveld, C.; Ordóñez, F.; and Tambe, M. 2010. Security games with arbitrary schedules: A branch and price approach. In *AAAI*, 792–797.

Kiekintveld, C.; Jain, M.; Tsai, J.; Pita, J.; Ordóñez, F.; and Tambe, M. 2009. Computing optimal randomized resource allocations for massive security games. In *AAMAS*, 689–696.

Korzhyk, D.; Yin, Z.; Kiekintveld, C.; Conitzer, V.; and Tambe, M. 2011. Stackelberg vs. Nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *J. Artif. Intell. Res.* 41:297–327.

Leitmann, G. 1978. On generalized Stackelberg strategies. *Journal of Optimization Theory and Applications* 26(4):637–643.

McCarthy, S. M.; Tambe, M.; Kiekintveld, C.; Gore, M. L.; and Killion, A. 2016. Preventing illegal logging: Simultaneous optimization of resource teams and tactics for security. In *AAAI*, 3880–3886.

Nguyen, T. H.; Yang, R.; Azaria, A.; Kraus, S.; and Tambe, M. 2013. Analyzing the effectiveness of adversary modeling in security games. In *AAAI*, 718–724.

Okamoto, S.; Hazon, N.; and Sycara, K. P. 2012. Solving non-zero sum multiagent network flow security games with attack costs. In *AAMAS*, 879–888.

Paruchuri, P.; Pearce, J. P.; Marecki, J.; Tambe, M.; Ordóñez, F.; and Kraus, S. 2008. Playing games for security: An efficient exact algorithm for solving Bayesian Stackelberg games. In *AAMAS*, 895–902.

Pita, J.; Jain, M.; Marecki, J.; Ordóñez, F.; Portway, C.; Tambe, M.; Western, C.; Paruchuri, P.; and Kraus, S. 2008. Deployed ARMOR protection: the application of a game theoretic model for security at the Los Angeles international airport. In *AAMAS*, 125–132.

Pita, J.; Jain, M.; Ordóñez, F.; Tambe, M.; Kraus, S.; and Magori-Cohen, R. 2009. Effective solutions for real-world stackelberg games: when agents must deal with human uncertainties. In *AAMAS*, 369–376.

Sandholm, T. 2015. Solving imperfect-information games. *Science* 347(6218):122–123.

Shieh, E.; An, B.; Yang, R.; Tambe, M.; Baldwin, C.; DiRenzo, J.; Maule, B.; and Meyer, G. 2012. PROTECT: A deployed game theoretic system to protect the ports of the United States. In *AAMAS*, 13–20.

Tambe, M. 2011. *Security and Game Theory - Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press.

Tsai, J.; Kiekintveld, C.; Ordonez, F.; Tambe, M.; and Rathi, S. 2009. IRIS-A tool for strategic security allocation in transportation networks. In *AAMAS*, 37–44.

Varakantham, P.; Lau, H. C.; and Yuan, Z. 2013. Scalable randomized patrolling for securing rapid transit networks. In *IAAI*, 1563–1568.

Von Stengel, B., and Zamir, S. 2004. Leadership with commitment to mixed strategies. *Technical Report LSE-CDAM-2004-01, CDAM Research Report.*

von Stengel, B., and Zamir, S. 2010. Leadership games with convex strategy sets. *Games and Economic Behavior* 69(2):446–457.

Xu, H.; Ford, B. J.; Fang, F.; Dilkina, B.; Plumptre, A. J.; Tambe, M.; Driciru, M.; Wanyama, F.; Rwetsiba, A.; Nsubaga, M.; and Mabonga, J. 2017. Optimal patrol planning for green security games with black-box attackers. In *GameSec*, 458–477.

Yang, R.; Ford, B. J.; Tambe, M.; and Lemieux, A. 2014. Adaptive resource allocation for wildlife protection against illegal poachers. In *AAMAS*, 453–460.

Yin, Y.; An, B.; and Jain, M. 2014. Game-theoretic resource allocation for protecting large public events. In *AAAI*, 826–834.