A Stochastic Game Model with Imperfect Information in Cyber Security

Harkeerat Bedi
Department of Computer Science, University of Memphis
hsbedi@memphis.edu
Dr. Sajjan Shiva and Dr. Dipankar Dasgupta

Introduction

Game theory provides huge potential to address the cyber security problem. The interaction between the attacker and the defender (system administrator) can be considered as a **game.** One of the techniques proposed by prior researchers used **stochastic game models** to emulate network security games.

Background:

Stochastic game: Each player performs an action at a given state and receives a payoff. Based on the previous state & actions performed, the game moves to a new state.

Perfect Information: The present state of the game is always known to each player.

Imperfect Information: The present state of the game is **not** always known.

Problem:

Prior researchers determined the Nash Equilibrium (NE) strategy for the defender considering the possible attack actions. However, they assumed that the players have perfect information about the current state of the game, which generally does not hold in reality.

Our contribution:

We compute Nash Equilibrium (NE) strategy for a zero sum stochastic game with imperfect information. In particular,

- We present a static analysis and compute the best strategy of the system administrator in realistic scenarios.
- Our analysis and simulation experiments illustrate that the system administrator will be better off if he/she takes our strategy compared to the scenario when he/she executes the strategy prescribed by the perfect information models.

Related Work and Motivation

Related work	Stochastic game?	Perfect Information?	Zero-sum / General-sum game	Type of analysis
(Lye 2005)	Yes	Perfect	General-sum	Static
(Alpcan 2003)	No(static game)	Imperfect	General-sum	Static
(Alpcan 2004)	No(repeated game)	Imperfect	General-sum	Dynamic
(Alpcan 2006)	Yes	Imperfect	Zero-Sum	Only Numerical Examples
(Nguyen 2009)	No(repeated game)	Imperfect	General-Sum	Dynamic
Our work	**Yes**	**Imperfect**	**Zero-sum**	**Static**

Prior stochastic game models for network security (Lye 2005) assume that the players have perfect information about the current state of the game,

- which implies that the defender is always able to detect an attack and the attacker is always aware of the employed defense mechanism.

In real systems, a player uses a sensor (e.g., Intrusion Detection System (IDS)) to observe the current status of the system to decide the strategy.

It is widely believed that no real sensor can perfectly read the environment, i.e., usually there is a non-zero <u>error probability</u>. Therefore, in most cases, the above assumption about perfect information does not hold in real life.

We compute the best strategy and the payoff considering such <u>error probabilities</u>.

The Game Model

Our model is an extension of the prior model (Lye 2005) and considers that a player k $(k = 1, 2)$ observes the game's true state at a particular moment by an imperfect sensor device. That means, player k can view the true state, ξ_j as any state in the information set $I^k_{\xi_j}$ with some probability where $I^k_{\xi_j} = \{\xi_{j_1}, \xi_{j_2}, \dots, \xi_{j_p}\}$ with ξ_j being an element of $I^k_{\xi_j}$.

Compared to the perfect information model, player k's action space may become wider, i.e., player k may take an action which is allowed at a state $\xi_{j_i} \neq \xi_j$ belonging to the information set, $I^k_{\xi_j}$.

Let $B^k_{\xi_j}$ denote the set of possible actions of player k when his/her information set is $I^k_{\xi_j}$. Then $B^k_{\xi_j} = \bigcup_{\xi_i \in I^k_{\xi_j}} A^k_{\xi_i}$ where $A^k_{\xi_j}$ denotes the action set of player k when he/she is sure that the true current state is ξ_i.

Below we formally define the outcome of player k's extended action set $B^k_{\xi_j}$, compared to $A^k_{\xi_j}$ in the previous model, when the true state is ξ_j. If player k takes an action $\alpha^k \in B^k_{\xi_j}$ when the true state is ξ_j but α^k is not in $A^k_{\xi_i}$, then in terms of the influence on state transition probability, α^k is equivalent to player k taking no action at state ξ_j. However, regarding the influence on player k's payoff α^k may not be equivalent to player k taking no action at state ξ_j depending upon the cost of the execution of α^k.

Formally, our model is represented by a tuple, $(S, I^1, I^2, E^1, E^2, A^1, A^2, B^1, B^2, Q, R^1, R^2, \beta)$ whose elements are defined below.

- $S = \{\xi_1, \xi_2, \dots, \xi_N\}$ is the set of states.
- $I^k = \{I^k{}_{\xi_1}, I^k{}_{\xi_2}, \dots, I^k{}_{\xi_N}\}$, $k = 1, 2$ where $I^k_{\xi_j}$ represents the information set of player k when the true state is ξ_j, i.e., $I^k{}_{\xi_j} = \{\xi_{j_1}, \xi_{j_2}, \dots, \xi_{j_p}\}$ (where p is an arbitrary positive integer) with the condition that $\xi_j \in I^k{}_{\xi_j}$.
- $E^k = \{E^k{}_{\xi_1}, E^k{}_{\xi_2}, \dots, E^k{}_{\xi_N}\}$, $k = 1, 2$ where the *j-th* set $E^k{}_{\xi_j}$ represents the error probabilities of *k-th* player's sensor at the true state ξ_j over the corresponding information set, $I^k{}_{\xi_j}$
- $A^k = \{A^k{}_{\xi_1}, A^k{}_{\xi_2}, \dots, A^k{}_{\xi_N}\}$, $k = 1, 2$ where $A^k{}_{\xi_j} = \{\alpha^k_{j_1}, \alpha^k_{j_2}, \dots, \alpha^k_{j_{M^k}}\}$ is the action set of player k at state ξ_j.
- $B^k = \{B^k{}_{\xi_1}, B^k{}_{\xi_2}, \dots, B^k{}_{\xi_N}\}$, $k = 1,2$ where $B^k{}_{\xi_j}$ represents the extended action set of player k at $I^k{}_{\xi_j}$. That means, $B^k{}_{\xi_j} = \bigcup_{\xi_i \in I^k_{\xi_j}} A^k_{\xi_i}$. By introducing identical actions we can make $|B^k{}_{\xi_j}|$ same for all $1 \leq j \leq N$. Let $T^k = |B^k{}_{\xi_j}|$.
- The state transition probabilities are represented by the function $Q: S \times B^1 \times B^2 \times S \to [0\ 1]$ which maps a pair of states and a pair of actions to a real number between 0 and 1.
- The reward of player k is determined by the function $R^k: S \times B^1 \times B^2 \to \mathbb{R}$ which maps a state and a pair of actions to a real number.
- $\beta, 0 < \beta < 1$ is a discount factor for discounting future rewards in this infinite horizon game.

We redefine the strategy function π^k of the perfect information model for this imperfect information model as $\pi^k: S \to \Omega^{T_k}$ where $\pi^k(s) = [\pi^k(s, \alpha_1), \pi^k(s, \alpha_2), \dots, \pi^k(s, \alpha_{T_k})]$

The definition of the payoff vector of player k $(v^k{}_{\pi^1, \pi^2})$ and the Nash equilibrium, (π^1_*, π^2_*) are similarly extended.

One major difference of this model from the perfect information game is as follows: As player k's sensor is not perfect, when his/her strategy π^k is executed in the true sense, his/her observed strategy (referred to as **apparent strategy**), $\pi^{k'}$ is different from π^k.

Static Analysis for a Game with Two States

To make the analysis simpler, we consider a game with only two states as illustrated below in Figure 1.

<u>Defender's sensor error probabilities:</u>

γ_1: false positive

γ_2: false negative

The system is either in *NormalState* (s_1) or in *HackedState* (s_2).

The defender's sensor is imperfect and the error probability at state: s_1 and s_2 are γ_1 and γ_2, respectively.

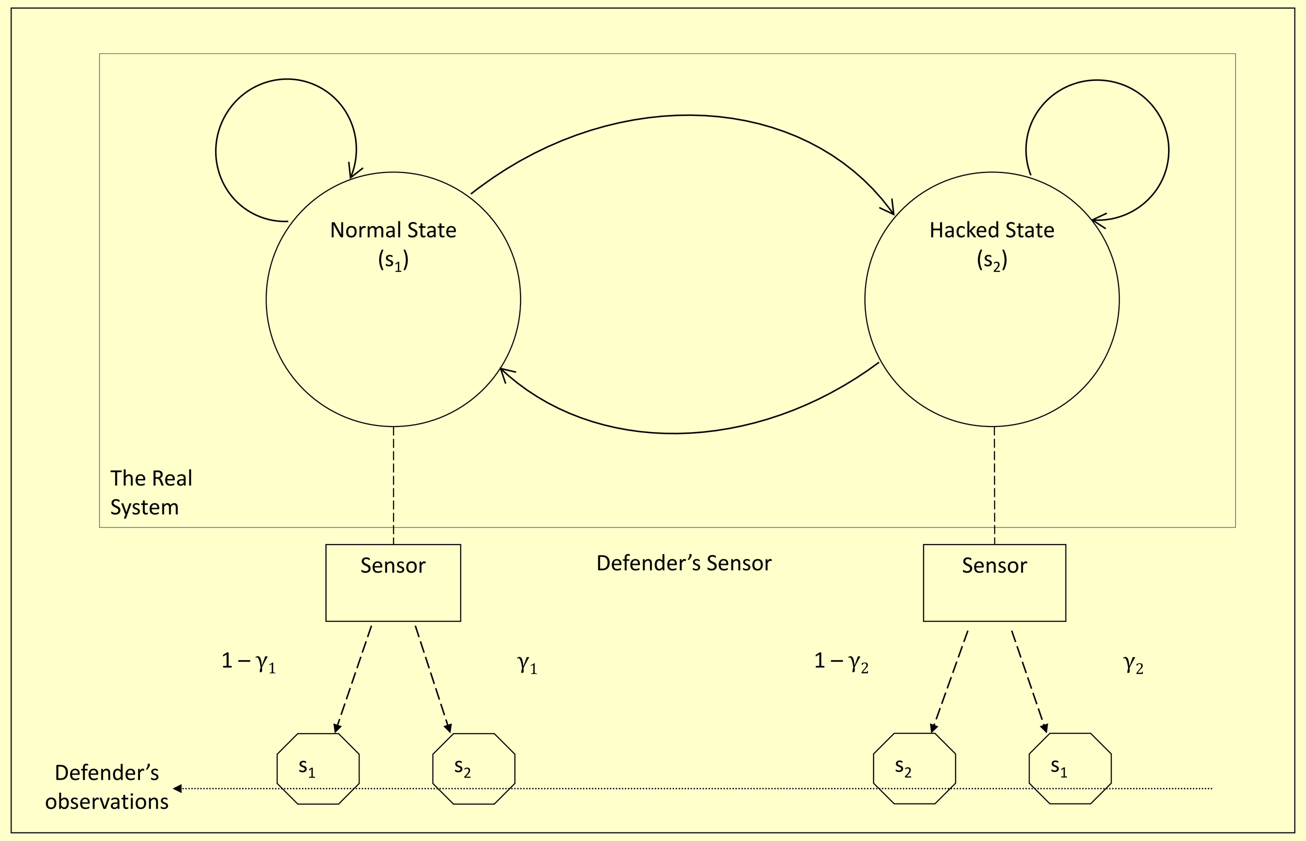

Figure 1: The state transition diagram and defender's observations — the same sensor is shown twice to indicate observations at different states

<u>Explanation:</u>

When the true state is s_1, with probability γ_1 the defender observes the state as s_2, and,

when the true state is s_2, with probability γ_2 the defender observes the state as s_1.

The action spaces of the players, A^1 and A^2 are as follows where a denotes 'attack', na denotes 'no attack', d denotes 'defense' and nd denotes 'no defense'. The first row in A^1 or A^2 represents the actions available in state s_1 and the second row is for s_2.

$$A^1 = \begin{bmatrix} a & na \\ a & na \end{bmatrix} \quad and \quad A^2 = \begin{bmatrix} d & nd \\ d & nd \end{bmatrix}$$

The strategy of the player k is represented by the probability distribution with which player k selects the available actions. The strategies of the players are represented by the following matrices π^1 and π^2 :

$$\pi^1 = \begin{bmatrix} \pi^1{}_{11} & \pi^1{}_{12} \\ \pi^1{}_{21} & \pi^1{}_{22} \end{bmatrix} \quad and \quad \pi^2 = \begin{bmatrix} \pi^2{}_{11} & \pi^2{}_{12} \\ \pi^2{}_{21} & \pi^2{}_{22} \end{bmatrix}$$

As an example, $\pi^1{}_{11}$ represents the probability with which player 1 selects action a and $\pi^1{}_{12}$ represents the probability with which player 1 selects action na at s_1 and $\pi^1{}_{11} + \pi^1{}_{12} = 1$.

Static Analysis for a Game with Two States (contd.)

State Occurrence Ratio (r_1, r_2): A stochastic game involves state transitions. The proportion of times a state s_i will occur during the whole play is called its occurrence ratio and is denoted by r_i. The value of r_i depends on the state transition probability function Q and the true strategies π^1 and π^2

Given true strategies π^1 and , π^2 we can compute the effective state transition probability matrix P whose dimension is $|S| \times |S|$. The element $P(i, j)$ represents the probability with which state s_i will switch to state s_j. Here, P is a 2 x 2 matrix.

We can compute r_1 and r_2 as follows. From basic theory of stochastic game (Filar 1997) we know that $P^i(1, j)$ represents the probability that state s_j will occur at the i-th transition.

$$r_1 = \lim_{n \to \infty} \frac{P(1{,}1) + P^2(1{,}1) + \ldots + P^n(1{,}1)}{n}$$

$$r_2 = \lim_{n \to \infty} \frac{P(1{,}2) + P^2(1{,}2) + \ldots + P^n(1{,}2)}{n}$$

We stress the fact that the defender's true strategy, π^2 is different from his/her apparent strategy, $\pi^{2'}$, which he/she observes being executed. We represent $\pi^{2'}$ as follows.

$$\pi^{2'} = \begin{bmatrix} \pi^{2'}{}_{11} & \pi^{2'}{}_{12} \\ \pi^{2'}{}_{21} & \pi^{2'}{}_{22} \end{bmatrix}$$

As an example, $\pi^{2'}{}_{11}$ represents the apparent probability of action d and $\pi^{2'}{}_{12}$ represents the apparent probability of action nd at s_1. Note that $\pi^{2'}{}_{11} + \pi^{2'}{}_{12} = 1$.

The defender's apparent strategy, $\pi^{2'}$ is determined by his/her true strategy, π^2, sensor error probabilities (γ_1, γ_2) and the true state transition ratios, (r_1, r_2) as described in the following matrix equation. The matrix IIF is called the Imperfect Information Factor and represents the influence of the sensor's errors.

$$\begin{bmatrix} \pi^{2'}_{11} & \pi^{2'}_{12} \\ \pi^{2'}_{21} & \pi^{2'}_{22} \end{bmatrix} = IIF \,.\, \begin{bmatrix} \pi^{2}_{11} & \pi^{2}_{12} \\ \pi^{2}_{21} & \pi^{2}_{22} \end{bmatrix} \quad \ldots \ (1)$$

$$where\ IIF = \begin{bmatrix} \dfrac{(1-\gamma_1)\, r_1}{(1-\gamma_1)\, r_1 + \gamma_2\, r_2} & \dfrac{\gamma_2\, r_2}{(1-\gamma_1)\, r_1 + \gamma_2\, r_2} \\ \dfrac{\gamma_1\, r_1}{\gamma_1\, r_1 + (1-\gamma_2)\, r_2} & \dfrac{(1-\gamma_2)\, r_2}{\gamma_1\, r_1 + (1-\gamma_2)\, r_2} \end{bmatrix}$$

Simulation Results

We simulate a stochastic game being played between an attacker and a system administrator using MATLAB. We implement an application that is able to produce the pair of optimal strategies for a zero-sum game with imperfect information. This application is based on the modified Newton's method as described under article 3.3 in (Filar 1997). An iterative non-linear optimization algorithm is used. The input to this algorithm includes the state transition matrix and the payoff matrix. As this is a zero-sum game, only the first player's reward matrix is given as input.

To compute the output, the modified Newton's method requires solving a *matrix* game in each iteration. This functionality is achieved by using an additional component that generates the optimal strategies and the value for a zero-sum matrix game as in (Williams 1966).

Result Summary:

- *Our first experiment shows that perfect information models* (Lye 2005) *can give higher payoff to the attacker compared to our model.*
- *Our second experiment shows the existence of such a game where strategies suggested by perfect information models could not be executed.*

Limitations and Scope of Future Work

- We assumed the defender knows the sensor's error probabilities (False Positive and False Negative ratios)
- Our current analysis is for only two states and each players' action set has only two actions.
- We performed a static (offline) analysis: Strategies are determined before the game is played.