ABSTRACT

GOYAL, SHUBHAM. Exploring Sanctions in CyberSecurity Compliance. (Under the direction of Professor Munindar P. Singh.)

Over the last few years there has been a major uptick in cybersecurity regulations across many sectors, but ensuring compliance with regulations is challenging. Many security breaches occur due to human factors such as people not following security protocols. However, there is little research exploring what mechanisms could reduce the adverse influence of human factors in system security. Sanctions are an effective way to regulate human behavior and promote norm compliance. We investigate the effects of different sanctioning mechanisms, specifically, group, individual, and peer sanction in promoting compliance with cybersecurity regulations and detrimental effect of sanction on the ability of an agent to complete their daily tasks.

Exploring Sanctions in CyberSecurity Compliance

by
Shubham Goyal

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Computer Science

Raleigh, North Carolina

2018

APPROVED BY:

Dr. Emerson Murphy-Hill	Dr. Arnav Jhala

Professor Munindar P. Singh
Chair of Advisory Committee

BIOGRAPHY

Shubham was born in Patna on August 8, 1992. He completed B.Tech. in computer science from Thapar University, Patiala in 2014.

Shubham then joined an early age startup Squadrun based out of Noida, India. He helped them in building the Squadrun's crowdsourcing platform from the ground up. In Squadrun, he learned about web and mobile technology. He also learned how different teams in a company work in collaboration to make a product possible. He aspires to start his own company.

Shubham joined NC State University in 2016 to pursue Master's degree in Computer Science.

ACKNOWLEDGMENTS

I am deeply indebted to my advisor Professor Munindar Singh for his endless guidance, motivation, and support during this thesis work. Munindar read countless drafts, corrected numerous mistakes, and provided astute guidance. Through this work he introduced me to the world of research. As I move on to next stage of my life, I hope I have imbibed some of Munindar's enthusiasm, work ethic, vision, and generosity.

I am thankful to my committee members Dr. Arnav Jhala and Dr. Emerson Murphy-Hill for discussing and providing useful feedback for this work.

I have benefited greatly from interactions with Nirav Ajmeri who has supported me in my research work through discussions and meetings. I would also like to thank my family for being a constant support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Recent years have seen a rapid proliferation in cybersecurity regulations across many sectors, but ensuring compliance with regulations is challenging. Many security breaches occur due to human factors such as people not following security protocols. However, there is little research exploring what mechanisms could reduce the adverse influence of human factors in system security [15]. Sanctions [12] are an effective way to regulate human behavior and promote norm compliance. We investigate the effects of different sanctioning mechanisms, specifically, group, individual, and peer sanction in promoting compliance with cybersecurity regulations and detrimental effect of sanction on the ability of an agent to complete their daily tasks. Specifically, we investigate the following research questions.

RQ1. How effectively do people apply and respond to group and individual sanctions to improve cybersecurity compliance?

RQ2. How detrimental are sanctions to user productivity?

1.1 Example

Example 1 (Zumbl Inc.) *Consider Alex, Bob, Charlie, and Dave who are software developers working at Zumbl Inc. Each of them has a workstation connected to the same network. Software developers are tasked with completing projects. They use various tools available on their workstation to complete the project they are tasked with. Each software developer has a different risk attitude and other personality traits. Additionally, Zumbl Inc has defined cybersecurity regulations such as updating passwords periodically, installing security patches, and so on, that the developers are expected to comply with. Erin is an IT administrator who looks after compliance with cybersecurity regulations in Zumbl Inc.*

Consider a case where Alex does not patch his workstation in time for an OS vulnerability.

Erin observes that Alex's workstation was not patched. She disconnects it from the local network so that other workstations are not at risk, and patches it. Erin takes some time to patch the computer, during which Alex could not complete the project tasks. Erin disconnecting Alex's workstation is an example of an *individual sanction* where the person who failed to comply with the regulation is sanctioned.

Alternatively, on noticing that Alex's workstation was not patched in time and could have affected other workstations on the same network, Erin along with disconnecting Alex's workstation, disconnects Bob's, Charlie's, and Dave's workstations. Erin disconnecting all the workstations prevents all software developers from working on their respective project tasks. Erin disconnecting all the workstations is an example of a *group sanction*. Here, instead of disconnecting only Alex's workstation, who did not patch it in time, Erin disconnected all the workstations on the network.

Consider another case where Alex has not patched his workstation. Bob notices that Alex has not patched his workstation and fears that an exploit of the vulnerability on Alex's workstation could result in all the workstations getting compromised. Bob frowns on Alex and suggests that Alex patch his workstation, otherwise all the workstations on the network would be at risk. Bob's frowning is an example of a *peer sanction*.

Sanctions are an effective way to regulate human behavior and can promote norm compliance. We investigate the effectiveness of group and individual sanctions as introduced above. Individual and group sanctions are applied by an administrator, a central party who has the responsibility of monitoring and sanctioning.

Sanctioning is an effective way to regulate human behavior and can promote norm compliance But it leads to less of productivity as illustrated in the example above. It may be counterproductive or detrimental to the efficacy of the agents. Intuitively group sanction can be more detrimental than individual sanction as the group sanction involves sanctioning the whole group in case of noncompliance by some subset of the group.

1.2 Related Work

Norms are a common regulation approach in multiagent systems [2, 17]. In the cybersecurity field, vulnerabilities due to user behavior, e.g., sharing passwords or ignoring firewall software upgrade, arise from failing to respect appropriate norms. A user may violate norms because they are not stated, imperfectly stated, or with malice aforethought: a problem with norms that lies at the root of a cybersecurity vulnerability. Munindar [15] explains that norms apply to all members of a microsociety, both stakeholders and adversaries, and can emerge from users sanctioning or imitating others. Our work focuses on different sanctioning mechanism to promote cybersecurity norm compliance.

Mahmoud, Miles, and Luck [10] argued that negative sanctioning as a means for enforcing social norms, has assumed that agents applying such sanctions have unlimited resources. This assumption does not hold true in real world settings. This motivates us to investigate how efficient are the sanctioning mechanisms. Mahmoud, Miles, and Luck [10] proposed a resource-aware adaptive punishment technique based on metanorm model to address the resource limitation issue, and evaluated it in a simulation. They showed their proposed punishment technique enables norm establishment with larger neighborhood sizes than the static and original adaptive punishment mechanisms. Sanctions are also used to promote norms in social contexts with privacy-aware agents [1, 4]. Reinforcement learning uses negative rewards as sanctions [11, 9].

To understand sanctioning in a system that pursues security compliance, Du et al. [8] implemented an exploratory multiagent simulation framework with agents performing security and research tasks under the scrutiny of a governing principal in a lab. They explored the effect of variable observability (immediate or delayed) of the governing principle and the type of sanction (individual or group) applied in the system. They found that sanctioning a group for a violation leads to greater compliance to security norms than sanctioning an individual. Later they expanded the simulation framework to investigate a more complex environment called ENGMAS, with the lab administrator's observability varies in a finer scale (from 0% to 100%) [7]. One interesting observation is that for both sanctioning mechanisms (individual and group), a lower observability of violations leads to greater success in task completion though at the cost of reduced security compliance. In this manner, the simulation can help guide the decision-making process of security policy makers. Our work is different from theirs on at least two aspects. First, we introduced peer sanction, which is not investigated before. Peer sanction is an important component because it happens implicitly in a system and impacts agent interactions regarding to security compliance. Second, we obtained data from gameplay by human player instead of simulation, through an online game capturing the essentials of a secure environment.

Sebastian, Dan, Rilla, and Lennart [5] explains that the idea of using game design elements in non-game contexts to motivate and increase user activity and retention has rapidly gained traction. Several applications now incorporate as a software service layer of reward and reputation systems with points, badges, levels and leader boards. We used the idea of game design elements to see how the security related behavior of user is affected by sanctions. According to Sebastian, Dan, Rilla, and Lennart [5] an interactive computer-based platform agnostic game designed for one or multiple players and has been developed with the intention to be more than entertainment is defined as *serious games*. In our study, game play is piggybacked to collect security related behavioral data of users, hence our game can be considered as *serious game*.

With the game we intend to simulate real life environment like research lab or corporate office where people are tasked with completing their daily task using a PC. At the same time,

we challenge them to complete the tasks required to maintain the security of the PC. Failure to complete the security tasks are met with different types of sanctions. Tarja, Mikael, and Per [16] explains that games allow learners to experience situations that are impossible in the real world for reasons of safety, cost, time, etc. In the game players are asked to assume the role of a person working in an office environment. Michele [6] explains that role-playing has long been an established technique used for educational activities. Role-playing in a digital environment fosters intrinsic motivation.

Chapter 2

The Game Model

This chapter delineates formally the conceptual model which investigates how sanctioning affects the security related behavior of humans. We further discuss the components of conceptual model in detail.

To further investigate our research questions we design a game based on the conceptual model. We discuss the elements of the game and how they are related to the conceptual model.

We conducted a study on Amazon Mechanical Turk where we asked participants to play the game. We discuss the design and different phases of the study.

2.1 Conceptual Model

We wish to investigate the effect of the sanctioning to humans' security-related behavior, we model a multiagent system.

The system has different types of agent who play different roles within the system.

Worker perform their domain-related tasks, i.e., the tasks for which they have an external incentive to complete, such as office work than an employee is paid to. Each worker owns a PC which is required for completing the domain-specific task. Meanwhile, an worker needs to perform security-related tasks such as patching the operating system, upgrading the firewall or changing the password and so on to keep the PC from being vulnerable. In a system without sanctions, workers may lack the motivation of maintaining security and thus tend to be busy with domain-specific tasks and ignore the security tasks. Whereas with sanctions integrated, the system has a *centralized agent* as its administrator or manager who monitors the system's security and takes measures to maintain the system's security. Possible security measures include imposing sanctions by evicting workers.

Outside of the system, there are potential *attackers*. They attack the system with the goal of compromising the PC in the system. The degree of sabotage by the cyberattacks to the system

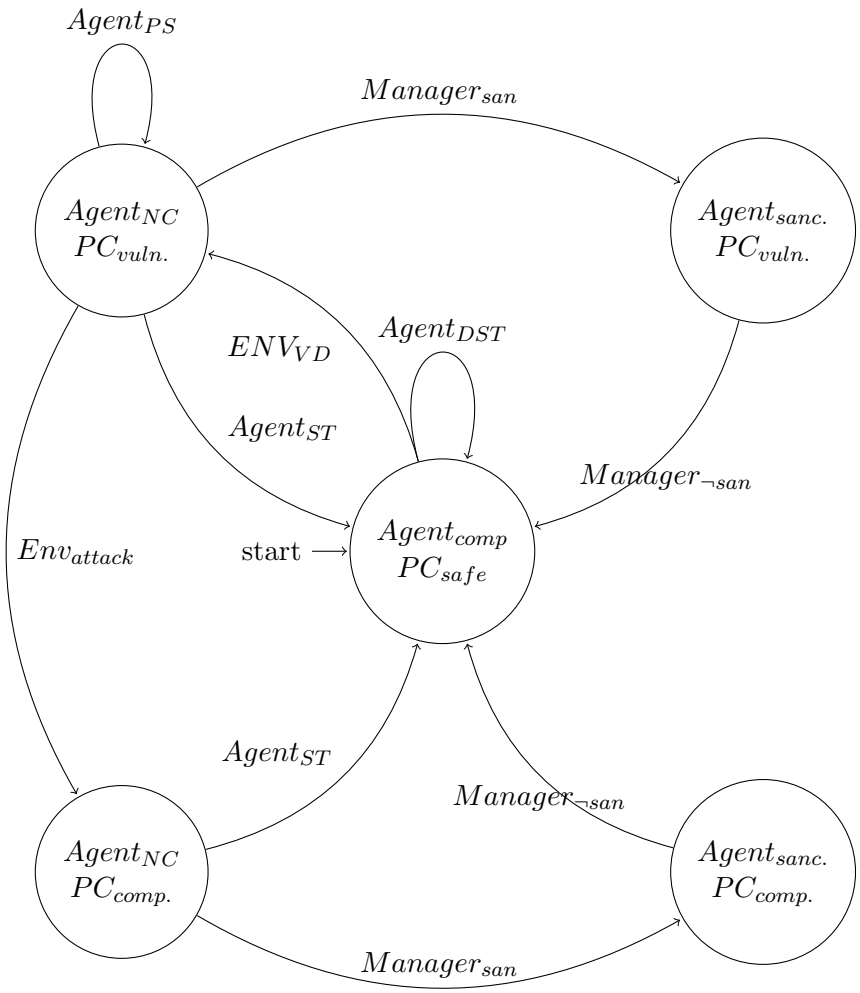

Figure 2.1 Conceptual model.

depends on how secure the PCs are. Below is the formal definition of multiagent model with cybersecurity concerns.

Definition 1 (Secure Multiagent Model) *The secure multiagent system model $O = \{A, C, M, E\}$ contains four components: A is a set of workers who perform domain-specific tasks, security-related tasks and can sanction other workers. C represents a PC that is owned by the worker A. M is the manager in charge of the system, and environment E is everything besides the previous three entities, including the attackers.*

2.1.1 Norms

A norm characterizes sound or *normal* interactions among the participants of a social group, reflecting their mutual expectations [14]. For instance, in the cybersecurity case, Zumbl Inc. is an organization, and the workstations are connected to form a network that seeks security. The IT administrator, Erin, monitors the network and take measures to ensure network security, by implementing a set of rules, such as patching and updating passwords. This set of rules define the security norms which should be followed by the worker that is the software developers in the company, reflecting the mutual expectation of cybersecurity.

Definition 2 (Norms) *Norms are the directed normative interactions among the agents reflecting their mutual expectations from the system [14]. We define norms in our model using a set of interaction rules:*

$$N = \{Interaction\ Rules_{agent}\}$$

In a cybersecurity environment, norms could be $\{Patching,\ Updating\ Passwords\}$. Failure to follow the norms are considered as norm violations. For example, suppose Zumbl Inc. asks employees to update passwords every three months. An employee who updated his password more than 3 months ago, is considered a violator of security norm. The employee may be sanctioned. From our research objective, we think about norms only from the perspective of the workers. We ignore the compliance or noncompliance for others.

2.1.2 Sanctions

Through the experience of sanctions or observation of sanctions on others, an agent can learn about the norms [13]. Sanctions are the consequences of the action of agents. Sanctions can be positive or negative. In real life scenarios, completing security tasks like changing password, updating antivirus, and so on, does not have explicit rewards but not completing the security tasks can lead to negative consequences in the form of cyberattacks successfully exploiting the

system. In a similar manner, we do not consider explicit positive sanctions but we employ negative sanctions as a consequence of not following the norm.

Definition 3 (Sanctions) *The possible sanctions could be:*

$$S = \{S_{Individual},\ S_{Group},\ S_{Peer}\}$$

We consider peer sanction as a negative sanction from the agent in question to its peers because it believes that they have failed to comply with security norms. Group sanctions by manager can also motivate an agent to sanction another agent who has not completed their immunity task in order to avoid group sanction. The sanctioner who issues the sanction has to be vigilant towards the security health of the PC of their peers and spend time to sanction their peers.

The manager can observe the security of all the PCs and has the authority to sanction any worker in the system. The manager observes the norm violations of agents based on its own observability. Depending on the specifics of the sanction, sanctionees have to face the consequences of the sanction. For instance, the sanctionee might be forced to spend time on fixing the security issue or the sanctionee PC may be evicted for certain time duration to fix the security issue.

2.1.3 Actions

Each agent has different assigned tasks, different belief sets, and thus completes different actions in the different scenarios. The new state achieved after the action can be compliant or non-compliant. If the state is non-compliant the agent can be sanctioned for the action.

An agent at any time step can perform a domain-specific task, security task, observe and peer sanction other agents or do nothing because the agent was sanctioned.

Definition 4 (Agent's Actions) *An agent's actions could be:*

$$Action_{agent} = \{DST,\ ST,\ PS,\ NPT\}$$

where DST represents domain specific tasks, ST represents security tasks, PS represents productivity tasks, and NPT represents non productive tasks.

The manager observes (*Observe*) the state of the system and performs sanctions if the state is not norm compliant. We assume that the manager has limited observability and cannot observe the security states of all the PCs at all times. This assumption is in line with real world scenario where the IT manager may not be able to observe all the PCs at all times in the organization.

Higher observability may quell the amount of repeated norm violations and increase the number of sanctions as manager would be able to observe more violations, while low observability may allow agents to continually ignore normative expectations.

Sanctions could be imposed on an agent (*+Sanction*), or lifted from an agent (*-Sanction*). Lift of a sanction means the sanction no longer has an effect on the agent. For example, an agent is sanctioned because of security norm violations and it cannot use its PC for domain specific tasks. After the agent fixes its PC's security issues, the sanction is lifted and the agent could use its PC again.

Definition 5 (Manager's Actions) *The manager's actions could be:*

$$Action_{manager} = \{Observe,\ +Sanction,\ -Sanction\},$$

where Sanction could be Individual Sanction or Group Sanction.

Definition 6 (External Environment Actions) *The external Environments actions could be:*

$$Action_{EE} = \{Attack,\ Vulnerability\ Discovered\}$$

At any point, there can be a cyberattack attempting to compromise the PCs or there could be a new vulnerability discovered in one of the resources being used by PC.

2.1.4 States

Along with sanctions, PCs could have different states indicating its availability for performing tasks.

Definition 7 (PCs States) *The set of PCs is represented as*

$$PC = \{PC_1,\ \ldots,\ PC_n\}$$

Each PC is associated with an agent. A PC's state model CU = {Safe, Vulnerable, Compromised} indicates whether it is safe, vulnerable, or compromised.

In the beginning, PCs are in the safe state. The PC stays in the safe state if the agent stays in compliant state. PC moves to a vulnerable state if the agent moves to a non-compliant state. An agent moves to a non-compliant state when a vulnerability is discovered and the agent has not yet fixed it. At this stage, the agent can be sanctioned for noncompliance or the vulnerability of the PC can be exploited by external agents and PC moves to a compromised state. The agent loses all the work on the PC. Agents are forced to perform security task at this stage to restore the PC to safe state again.

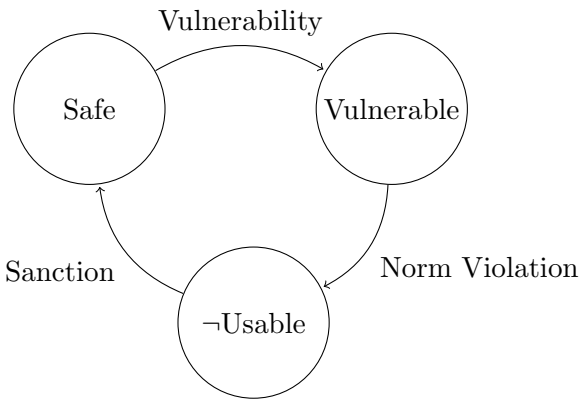

Figure 2.2 PC's usability model.

Definition 8 (Agent's States) *The set of agents is represented as* $AG = \{AG_1, ..., AG_n\}$. *Each agent has an associated PC. A agent's state model* $AU = \{Compliant, \neg Compliant, Sanctioned\}$ *indicates whether the agent is currently compliant, non compliant or sanctioned.*

In the beginning, an agent is in a compliant state. Later, agents perform actions. Based on the choice of actions, an agent can move to a non-compliant state or stay in a compliant state. If the agent moves to a non-compliant state, the agent can be sanctioned. Based on the nature of sanction, the agent needs to take certain actions or abide by certain limitations after which agent moves back to a compliant state.

The interaction between states and action is shown in Figure 2.1. Circles shows the state of the agent and the PC. The caption on the arrow from one circle to another shows the action.

2.2 Game Design

To investigate how our model fits into real life situation, we designed a web-based game following the model.

The game considers three resources: R_1, R_2 and R_3. The resources are represented by red, blue, and yellow color. Section I in Figure 2.3 shows the project section. The projects are equivalent to the domain-specific tasks. There are three types of project namely small, medium, and large with one, two, and three tasks respectively. Each task (T_i) is mapped to a resource (R_i) and requires that R_i be available for T_i to be completed. The availability of R_i is visible in Section III in Figure 2.3. The availability of resource is referred to as capability C_i in the game. The tasks that makes up a project is chosen randomly. For instance, in Figure 2.3, the medium project is made up by red and yellow resources. The choice of red and yellow resources are random. After a player completes all the tasks of a project, new resources are assigned to

the project randomly.

Further there is an external environment which tries to compromise the resource using attacks. Every attack is directed towards a particular resource. After an attack (A_i), the player loses the immunity (I_i). For instance, in case of yellow attack the player will lose the yellow immunity.

The immunity is gained back by the player by completing corresponding immunity task shown in section II in Figure 2.3. There is a deadline D_i associated with each immunity task. If the player does not complete the immunity task before the deadline, the player can be sanctioned by the manager.

In case of an attack A_i directed at the resource R_i whose I_i is already lost, the player loses the C_i. The availability is shown in section III in Figure 2.3. The availability C_i can be gained back by completing the immunity task I_i. For instance, in case an yellow attack and if the player does not have the yellow immunity, the player will loose the yellow capability as well. After losing the yellow capability the player would not be able to complete yellow tasks in the project section until the player completes the yellow immunity task. A completed immunity task persists until there is another attack directed at the resource.

Every player in the game aims to earn the most number of points. The points in the game is equivalent to compensation that an employee of the organization gets for completing their daily tasks. Every player can see the current score of other players in the 'colleagues' section on the left side in Figure 2.3. The visible score of other players may motivate the players to perform better and improve their score. At the same time it can have negative implication. A player can continually peer sanction other players to prevent them from completing project task and gaining more points. To avoid the misuse of score information, the score can be completely hidden but that would have a negative implication in the form that players would not know how they are performing in comparison with other players and might lose the motivation to perform better.

The game is divided into 40 rounds, and in each round every player has to make a move before the game moves to the next round. Further, we explore each component of the conceptual model and see how they are mapped to the game.

2.2.1 Norms

Table 2.1 shows that an immunity task has a deadline of three rounds i.e., the player gets three rounds to complete the immunity task after the immunity is lost without the risk of being sanctioned. Norm for a player is to complete the immunity task before the deadline ends while also completing project tasks to earn maximum points.

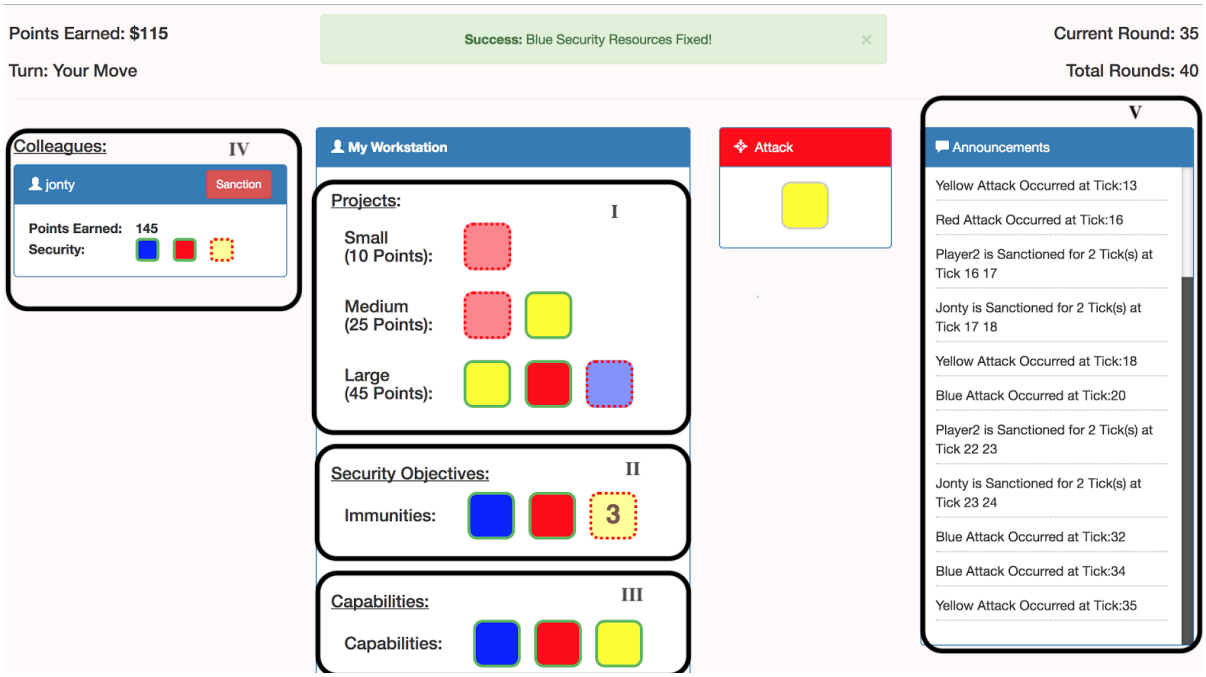

Figure 2.3 Game screenshot.

Table 2.1 Game Parameters.

Attack Probability	0.35
Manager Observability	0.3*Unfixed immunities
Small Project Score	10
Medium Project Score	25
Large Project Score	45
Tasks in Small Project	1
Tasks in Medium Project	2
Tasks in Large Project	3
Deadline for immunity task	3

2.2.2 Actions

The game has different agents. Each agent has different duties and, therefore, perform different actions in the system. We discuss the actions of each agent.

Player Actions

In a round, a player can take one of the following actions in game:

- *Complete Project task*: There are three types of projects based on the number of tasks required to complete the project. Table 2.1 shows the points awarded and number of tasks for each type of project. The player is awarded the score corresponding to a project only after the player has completed all the tasks of a project.
- *Complete immunity tasks*: Players do not get awarded any points for completing immunity tasks but are required to complete it to avoid sanctions and to retain the availability of a resource.
- *Peer Sanction*: A player could peer sanction other players if they observe that they have not completed their immunity tasks.
- *Do Nothing*: When the player is sanctioned by the manager, the player can only pass the round by clicking on the pass button. Clicking on the pass button is an acknowledgment from the player that he has completed his turn.

Manager Actions

The manager observes the immunities of each player at the beginning of each round. If the manager observes that a player has not completed the immunity tasks past the deadline, the manager sanctions the player based on a probability. The sanction is applied at the beginning of the round. Table 2.1 shows that for one incomplete immunity task, the probability that the player will be sanctioned is 33 percent, and for two incomplete immunity task the probability increases to 66 percent. When a player has three incomplete immunity task, the probability increases to 100 percent. The sanctioning can be a group or individual sanction, depending on the means of sanction employed in the game.

Environment Actions

At the beginning of every round there can be an attack. To decide whether or not an attack will happen in a particular round, an attack probability is set for each game which is fixed throughout the game. Table 2.1 shows that the attack probability of 0.35 for all the games. The attack probability was chosen such that the number of attacks in a game are neither too high nor too low. High attack probability would eventually lead to players losing their capabilities thus

forcing them to complete the immunity task and would lead to same behavior by every player. Low attack probability might lead to a behavior where every player completes the immunity task instantly. We wanted that players have to constantly choose between completing the immunity task and project tasks.

There are four types of attack in the game: A_1, A_2, A_3 and A_4. A_1, A_2 and A_3 attack results in the loss of corresponding immunities I_1, I_2 and I_3. If the corresponding immunity is already lost then the availability of the corresponding resource is lost. A_4 attack is equivalent to A_1, A_2 and A_3 attack happening simultaneously. This presents players with cognitively most challenging goal to the user. The player has to complete all the immunity tasks within their respective deadlines.

To decide the type of attack, a random number is chosen between 0 and 100. If the number lies between 0 and 30 A_1 attack is chosen, for 30 to 60 A_2 attack is chosen, for 60 to 90 A_3 attack is chosen and for 90 to 100 A_4 attack is chosen. This distribution ensures that the probability of A_1, A_2 or A_3 is equal whereas the probability of A_4 is one third of other attacks.

The attacks in the game emulate the attack attempts made by the external environment. The attack frequency is the ability of external hacker on how frequent attack attempts the hacker make. Different types of attack corresponds to attacker trying to exploit different services used in the PC.

2.2.3 Sanction

The means of sanction in each game is configured to be either group sanction or individual sanction in addition to peer sanction. The idea behind sanctioning is to have a negative reinforcement, to regulate the behavior of players.

There are two ways of sanctioning in the game. First is a group or individual sanction by the manager, where a player is forced to pass a certain number of rounds n, n is equal to twice the number of unfinished immunity task. The player passes the round by clicking on a pass button which appears after a player is sanctioned. The player cannot perform any other action than clicking the pass button for the rounds for which the player has been sanction. After n rounds, the player gains back the immunity for which he was sanctioned.

Second is losing the the availability of the resource. Each resource has a corresponding availability. Losing the availability means that player can no longer complete task associated with that resource until it completes the corresponding immunity task. For example if a player loses the red availability the player cannot complete red task anymore until the player completes the red immunity task. Losing the availability is equivalent to having the PC in a compromised state. Player also loses the completed task associated with that resource in the ongoing projects on losing the availability of the resource. A player loses the availability when an attack happens

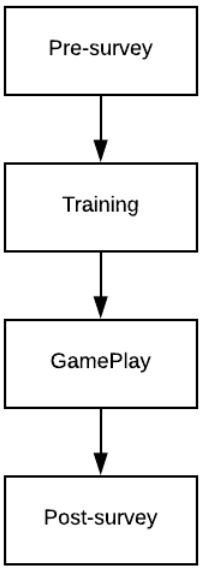

Figure 2.4 Flow of the experiment.

and the player does not have immunity for that attack. For example player will loose the red capability when a red attack happens and players does not have red immunity.

2.3 Experimental Design

We conducted a study on Amazon Mechanical Turk where we asked workers to play the game. Thirty workers participated in the study playing 107 games. The group size for the games varied between 2–5. The study was approved by our universityś Institutional Review Board. We collected an informed consent from each subject and provided a payment to each subject completing the study. Each task that a worker performs in mTurk is called human intelligence task (HIT). The researcher approved the HIT if the worker participated in all the phases of the study irrespective of how the worker performed in the games played during the study.

We created a project task on mTurk. In our project design, workers could see instructions and an informed consent form before they accepted the HIT. Workers need to sign the consent form to see the link to the game. We specifically asked the workers not to participate in the study multiple times, since we posted several batches on throughout the study. This was to mitigate the threat of learning. The game link redirected to our server, on which we hosted the game.

The study was conducted in slots of 60 minutes. We made multiple 60 minutes slots available for the study. Each worker after accepting the HIT was asked to select a slot. The worker was

expected to visit the game URL at the beginning of their slot and be available for next 60 minutes. To start with the study, participants were required to log in after visiting the game link. The username and password required for logging in was participants mTurk ID, which was communicated to them in the instructions of HIT. After logging in, worker were required to join a group chat whose link was available to the workers after logging in. The group chat made it easier for us to coordinate with the workers.

The study involved four phase as shown in Figure 2.4.

- *Pre-survey:* We asked workers to complete a survey which helped in assessing worker's personality and creative potential. First, we employ the Mini-IPIP (International Personality Item Pool) [3] scales to measure a worker's Big Five personality traits are Extraversion (E), Agreeableness (A), Conscientiousness (C), Neuroticism (N), and Openness to experience (O).

  Second, we employ the DOSPERT scale to assess a worker's risk attitude. DOSPERT is a psychometric scale that assesses risk taking in five content domains: financial decisions, health/safety, recreational, ethical, and social decisions.

  Both Mini-IPIP and DOSPERT are well-known scales. We choose these two scales because of their compactness. If longer alternatives were to be used, they would increase the time spent on completing the pre-survey and would leave us with less time for the gameplay.
- *Training:* In the training phase, each worker was asked to watch a five-minute video which explained the game. After watching the video, the workers played two games, these two games were not included in the statistical evaluation. These games helped workers in getting acquainted with the game. Workers were informed that these games will not be evaluated and they were encouraged to familiarize themselves with different elements of the game. First demo game employed individual sanction and the second demo game employed group sanction.
- *Gameplay:* After the training phase, each worker was asked to play four games. Two games employed individual sanction and the other two games employed group sanction in no particular order. Further, after each game workers completed a short post-game survey to record their feedback on the sanction policy employed in the game.
- *Post Survey:* After the gameplay, workers were asked to complete a post-study survey capturing their feedback on the overall study.

We encouraged and rewarded players to give their best in the game by promising them a bonus based on the score in the game.

Chapter 3

Result and Discussion

We now describe the empirical evaluation that compared group and individual sanctioning mechanism based on the study conducted on Amazon Mechanical Turk. The idea of the study was to compare how people respond to each sanctioning mechanism and how their productivity in completing daily task is affected by these sanctioning mechanism.

3.1 Metric

To compare group sanction and individual sanction we measure the following metrics:

- *Compliance*: Compliance measures the frequency of an agent being compliant with a norm such as how frequently an employee of an organization is in completing the security tasks of changing the password or updating the anti-virus in a timely manner. In this case, compliance means completing the immunity tasks before the deadline ends. We measure compliance via following measures:

  Completed Security Tasks: In the game, after an attack, a player loses the immunity (S_{i}) and is given a deadline (D_{i}) to fix the immunity. Fixing an immunity before the deadline implies completing the security task.

  Manager Sanctions: This measure tells us the number of sanctions by the manager. A player is manager sanctioned because the player did not complete the immunity task before the deadline. For individual sanction, it is calculated by the total number of sanctions issued to individual violators. For group sanction, it is calculated by the total number of sanctions issued to the group, despite the total number of players in the group. For example, for a game with three player group, an instance of group sanction would be counted as one, not three.

- *Efficacy*: Efficacy measures how productive workers are in completing their domain-specific tasks. Sanctions are detrimental to the productivity. We measure how detrimental group and individual sanctions are to the productivity. In the game, We measure efficacy via the following measures:

  Score: Every time a player completes a small, medium, or large project corresponding points are awarded to the player. Score measures how productive a player is. Higher scores indicate greater productivity.

  Rounds Passed: Whenever a player is sanctioned by the manager or the peers, player is forced to pass a certain number of rounds. Passing a round means that player is not able to complete any productivity or security tasks in those rounds. This is equivalent to an IT administrator taking away the PC of an employee when the IT administration finds out that the employee has not applied security patches. Employee is not able to complete their daily task until the PC is returned by IT administrator.
  Higher number of rounds passed would mean players were non productive for those rounds and overall were less productive in completing the productivity tasks.

- *Resilience*: Resilience is measured by how quickly the system gets back to the state of being norm compliant. For example, when a vulnerability is discovered on a PC of an employee, how soon the vulnerability is fixed either by the employee or the IT administrator. In our game, We measure resilience via the following measure:

  Rounds taken to gain back the immunity: Number of rounds taken by a player to gain back an immunity after losing it to an attack.

3.2 Experimental Results

Each worker played two games with group sanction and two games with individual sanction in span of 60 minutes. All the game parameters other than the sanctioning method were kept same throughout the study in all the games. These parameters can be seen in Table 2.1. We record every move made by a player in every game and evaluate this data to compare group and individual sanction with respect to each of the measures described above.

Compliance

Figure 3.1 shows the box plot of the ratio of number of immunity task completed before deadline and number of immunity task available by each player in each game. We apply paired sample t-test to determine whether their is significant difference between the number of immunity tasks completed in group and individual sanction. P-value of more than 0.05 shows that we cannot reject the null hypothesis and there is no significant difference between the mean of number

of immunity task completed. Although individual sanction leads to slightly better results that group sanction.

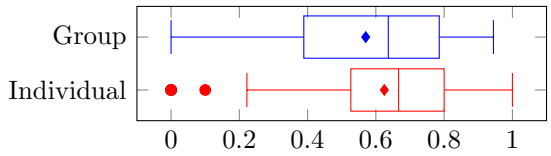

Figure 3.1 Ratio of immunity tasks completed before deadline and immunity task available.

We calculated the number of manager sanctions in each game and divided it by number of attacks in that game to normalize the number of sanctions. Figure 3.2 shows that on an average there are 23 sanctions for every 100 attacks in individual sanction games whereas there are 47 sanctions for every 100 attacks in group sanction games. P-value of less than 0.05 shows a significant difference, while surprising, it is compatible with the result in Figure 3.1.

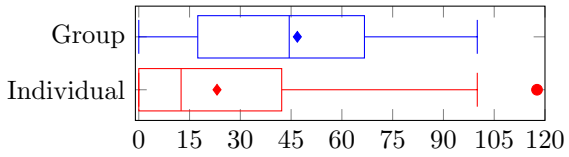

Figure 3.2 Number of manager sanctions for every 100 attacks.

We found that there were 14 instances of peer sanction in games with group sanction and seven instances of peer sanction in games with individual sanction. This indicates that group sanction promotes peer sanction more than individual sanction.

After each game we asked the players how effective was the sanctioning mechanism on a scale of 1 to 5 with 1 being not at all influential and 5 being very influential. 77 percent of participants said that sanctioning was a strong factor (4–5) in influencing their decision making.

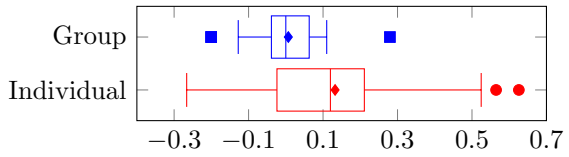

Figure 3.3 Difference between number of immunity tasks completed before deadline in first game and second game.

Each participant played two games each with individual and group sanction. We calculated the difference between the number of immunity tasks completed in the second game and the first game to measure the influence of sanction on the player. Figure 3.3 shows the same. The difference has been normalized by dividing it with number of attacks in the game. On average a player completed 13 percent more immunity tasks per attack in the individual sanction games whereas in group sanction games number of immunity task completed in two games was almost same. P-value of less than 0.05 shows a significant difference and signifies that individual sanction was more effective in motivating people to be security compliant as compared to group sanction.

To compare the compliance of workers based on their risk attitude, We divided the risk score of each domain obtained from DOSPERT survey into three categories: low, medium, and high. We calculated the number of manager sanctions and number of immunity tasks completed by players in each category. Figure 3.4 and Figure 3.5 shows that sanctions were more effective in promoting compliance among players with low and medium risk taking attitude in social domain as compared to people with high risk taking attitude.

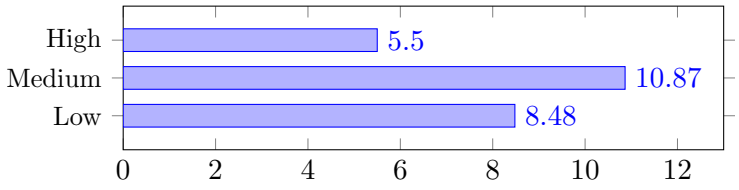

Figure 3.4 Immunity tasks completed per player grouped according to social risk taking personality.

Efficacy

Figure 3.6 shows the average score of players in a game. We found that participants were more productive in individual sanction games. The average score in group sanction games is 372.5 as compared with average score of 333.85 in individual sanction game. P-value of less than 0.05 shows that their is significant difference. The result is not surprising, as in group sanction, players who complete their immunity task are sanctioned because other players of the group.

Figure 3.7 shows that the number of rounds a player passed in a game. In one game of group sanction, a player had passed 15 rounds out of 40 rounds. On average, the number of rounds passed in group sanction is more than the double of number of rounds passed in the individual sanction. P-value of less than 0.05 also shows that their is significant difference. This also explains the low average score in Figure 3.6 in group sanction games as compared to

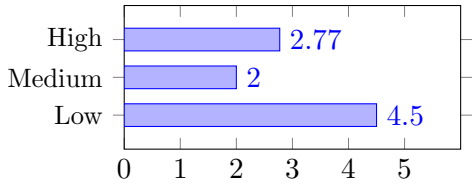

Figure 3.5 Manager sanction per player grouped according to social risk taking personality.

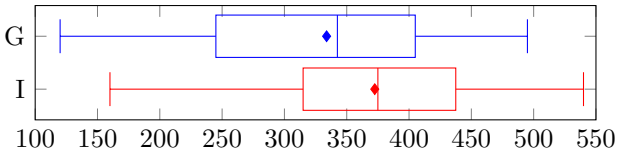

Figure 3.6 Average score.

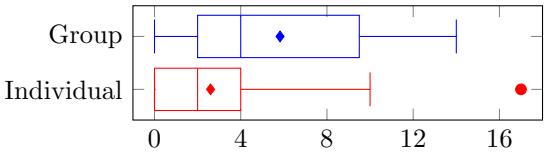

Figure 3.7 Average number of rounds passed.

individual sanction games.

After each game we asked our participants how detrimental the sanctioning mechanism was to completion of tasks in the game on a scale of 1 to 5 with 1 being very detrimental and 5 being not at all detrimental. Figure 3.8 shows that participants found group sanction and individual sanction almost equally detrimental. P-value of more than 0.05 shows no significant difference between the means.

Resilience

We calculated the number of rounds a player took to gain back the immunity lost to an attack. The immunity could be gained back either by player completing the immunity task or being sanctioned by the manager which in turn fixes their immunities. Figure 3.9 shows that both group sanction and individual sanction motivated participants to complete the immunity task in the same round as both of them have mean close to zero. P-value of more than 0.05 shows no significant difference between the means. In terms of resilience, group sanction and individual sanction were equivalent.

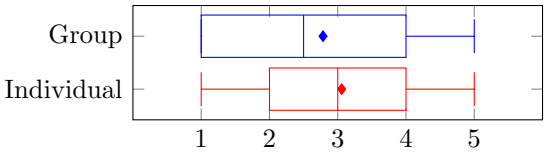

Figure 3.8 Perceived detrimental effect of sanction.

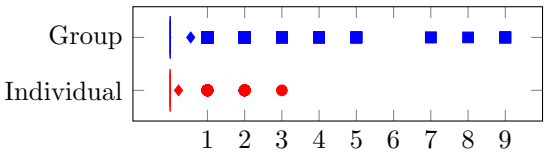

Figure 3.9 Rounds taken to gain back the immunity.

3.3 Threats to Validity

- *Limited Sample Size*: One limitation to the generalizability of our study is the sample size. Although we made 30 workers from mechanical turk participate in the study, they may not be representative of the larger population of the actual industry. Although high number of worker on mechanical turk signed up to participate in the study, the show rate at the designated time slot was low.
- *Motivation*: A player may not be motivated enough to give their best in the game and thus the moves would not be well thought. We motivated the players by promising them a bonus based on the performance in the game.
- *Game Understanding*: To make sure that the players understood the game, we made a demo video with game screen-shots explaining the game in detail. In addition, first two games played by every player were not used for statistical evaluation as these were supposed to be trial games helping the players in understanding the game.

3.4 Conclusions

Using the results obtained from statistical analysis of the data collected from the games played by workers on mechanical turk, we conclude that individual sanctions are more effective in terms of motivating people to be compliant with the security regulations that group sanction.

Workers were less productive in completing the projects tasks in group sanction games as compared to games with individual sanction but they were more peer sanctions in case of group sanction mechanism.

3.5 Future Directions

In future work, it would be interesting to experiment with combination of group and peer sanction with variable manager observability. Observability of manager can be decreased gradually relying more on peer sanction to keep the agents security compliant. This type of system would scale better, but the effects of this on the productivity of workers is yet to be seen.

Group sanction along with peer sanction has the potential to function in a self sustaining system with no or little central agent involvement, where as in case of individual sanction there would be always be a requirement of central agent to make the system work in a security compliant manner.

It would also be interesting to explore positive sanctions in future studies where an explicit positive reward is awarded for complying with the security regulations. The final score could be calculated by taking weighed average of number of security tasks and daily tasks instead of just daily tasks which is done in the current game.

REFERENCES

[1] Nirav Ajmeri, Pradeep K. Murukannaiah, Hui Guo, and Munindar P. Singh. Arnor: Modeling social intelligence via norms to engineer privacy-aware personal agents. In Kate Larson, Michael Winikoff, Sanmay Das, and Edmund H. Durfee, editors, *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil*, pages 230–238. ACM, 2017.

[2] Giulia Andrighetto, Cristiano Castelfranchi, Eunate Mayor, John McBreen, Maite Lopez-Sanchez, and Simon Parsons. (Social) Norm Dynamics. In Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W. N. van der Torre, editors, *Normative Multi-Agent Systems*, volume 4 of *Dagstuhl Follow-Ups*, pages 135–170. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2013.

[3] M Brent Donnellan, Frederick Oswald, Brendan Baird, and Richard E Lucas. The mini-ipip scales: Tiny-yet-effective measures of the big five factors of personality. 18:192–203, 2006.

[4] Amit K. Chopra and Munindar P. Singh. Custard: Computing norm states over information stores. In Catholijn M. Jonker, Stacy Marsella, John Thangarajah, and Karl Tuyls, editors, *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, Singapore, May 9-13, 2016*, pages 1096–1105. ACM, 2016.

[5] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: Defining gamification. In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011*, volume 11, pages 9–15, 09 2011.

[6] Michele D. Dickey. Game design and learning: a conjectural analysis of how massively multiple online role-playing games (mmorpgs) foster intrinsic motivation. *Educational Technology Research and Development*, 55(3):253–273, Jun 2007.

[7] Hongying Du, Bennett Narron, Nirav Ajmeri, Emily Berglund, Jon Doyle, and Munindar P Singh. Engmas–understanding sanction under variable observability in a secure environment. In *Proceedings of Second International Workshop on Agents and Cybersecurity (ACySE)*, pages 15–22, Istanbul, 2015.

[8] Hongying Du, Bennett Narron, Nirav Ajmeri, Emily Berglund, Jon Doyle, and Munindar P. Singh. Understanding sanction under variable observability in a secure, collaborative environment. In *Proceedings of the 2015 Symposium and Bootcamp on the Science of Security*, HotSoS, pages 12:1–12:10, New York, NY, USA, 2015. ACM.

[9] Jiaqi Li, Felipe Meneguzzi, Moser Fagundes, and Brian Logan. *Reinforcement Learning of Normative Monitoring Intensities*, pages 209–223. Springer International Publishing, Cham, 2016.

[10] Samhar Mahmoud, Simon Miles, and Michael Luck. Cooperation emergence under resource-constrained peer punishment. In *Proceedings of the International Conference on Autonomous Agents, Multiagent Systems*, AAMAS, pages 900–908, Richland, SC, 2016. International Foundation for Autonomous Agents and Multiagent Systems.

[11] Mehdi Mashayekhi, Hongying Du, George F. List, and Munindar P. Singh. Silk: A simulation study of regulating open normative multiagent systems. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI*, pages 373–379. IJCAI/AAAI Press, 2016.

[12] Luis G. Nardin, Tina Balke-Visser, Nirav Ajmeri, Anup K. Kalia, Jaime S. Sichman, and Munindar P. Singh. Classifying sanctions and designing a conceptual sanctioning process model for socio-technical systems. *The Knowledge Engineering Review (KER)*, 31:142–166, March 2016.

[13] Bastin Tony Roy Savarimuthu, Rexy Arulanandam, and Maryam Purvis. Aspects of active norm learning and the effect of lying on norm emergence in agent societies. In *Proceedings*

of the 14th International Conference on Agents in Principle, Agents in Practice, PRIMA, pages 36–50, Berlin, Heidelberg, 2011. Springer-Verlag.

[14] Munindar P. Singh. Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):21:1–21:23, December 2013.

[15] Munindar P. Singh. Cybersecurity as an application domain for multiagent systems. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1207–1212, Istanbul, 2015. IFAAMAS.

[16] Tarja Susi, Mikael Johannesson, and Per Backlund. Serious games : An overview. Technical Report HS- IKI -TR-07-001, University of Skvde, School of Humanities and Informatics, 2007.

[17] Daniel Villatoro, Giulia Andrighetto, Rosaria Conte, and Jordi Sabater-Mir. Self-policing through norm internalization: A cognitive solution to the tragedy of the digital commons in social networks. *Journal of Artificial Societies and Social Simulation*, 18(2), 2015.