Introduction to Security and Policy Section

Dan Thomsen

The rewards and rules of human collaboration systems shape the behavior of the human participants, often leading to behaviors the human computation system designers never envisioned. Most software designers make the mistake of assuming that people will follow the intent of the rules they set up in the program. But, as the rising wave of cyber-crime shows, people do what they can get away with. People will do anything they can to achieve rewards, and sometimes the reward means breaking the system for the joy of figuring out how to solve a puzzle.

Most systems suffer from concentrating on the functionality of the system and the developers add security only when they realize someone has broken the rules. Developers assume that people will behave in "cyber-space" the same way the behave in the real world. Unfortunately the anonymity, and the cognitive leverage that computers give people result in a very different behavior.

Many human computation systems have altruistic goals, and will have a core assumption that people are participating to do "good". Even, if they recognize the existence of malicious users the developer's core assumption of "good" players tamps down their ability to see just how devious a malicious player can be. For example, will they seed problems to have the other human solvers help them in support tasks for a malicious goal? How will people mix in things from the real world into the human computation environment, like greed, jealousy and network packet lengths that the developer simply has not thought of?

People have postulated that reward mechanism in human computation games must be carefully constructed to reward the exact behavior desired. If the rules are poorly designed people playing within the rules may perform legal actions that result in different outcomes from what the developers intended. Now consider at the meta-rules; what the human computation system actually enforces at a low level. What parts of the system are easy to break, spoof, or compromise? Now can you predict the final outcome of a collaboration effort?

D. Thomsen (✉)
SIFT, LLC, USA
e-mail: dthomsen@sift.net

P. Michelucci (ed.), *Handbook of Human Computation*,
DOI 10.1007/978-1-4614-8806-4_66, © Springer Science+Business Media New York 2013

Growing up in rural Minnesota, there was a saying, “Locks are for honest people”. Meaning the people that locks stop from entering your house probably did not plan on taking your stuff anyway. We locked the door mainly to keep neighbors from leaving apples, or other produce we would have to eat or can. The one time our home was burglarized the thieves pushed the locked front door out of its frame to get into the house. Which was unfortunate since we left the back door open in case a neighbor really had to get rid of a bushel of apples. For human computation, this means two things, first you have to have set of clearly defined mechanisms to get the human behavior you want for the honest people. This first set of obvious security mechanisms guides the honest people to the desired behavior. Second, you need hard security mechanisms to enforce that behavior, detailed auditing for when the mechanisms fail, and policy to make it clear to malicious users the penalty for malicious behavior.

There is a complex dance between security and policy. Building secure software costs money and developers must always trade security for features to reduce development time, which leaves gaps in the security enforcement model. Often policy can cover these gaps cheaper than developing the necessary software. For example, developers can easily implement password authentication, but passwords do not really identify the person, only that the person knows the secret. A complete security system includes a policy, to inform the users not to share their passwords with anyone. This allows the developers to trade development costs for a weaker form of authentication. Human computation environments must make these same types of trade-offs; implementing mechanisms to encourage desired behavior, and policy to define behavior when the mechanisms are insufficient.

In this section we have five papers covering both security and policy. Felstiner writes about labor laws standards and human computation. Computation environments could be written that enforce fair labor laws for different countries, but think of the expense for simply understanding all those different laws, let alone implementing them in software. Instead developing clear policies about what labor laws mean in a global, anonymous job market can help ensure fairness with less expense. James Caverlee also discusses exploitation in human computation systems. System developers may assume that contributors are volunteering their time, but money and rewards attract middlemen who might be packaging laborers from depressed economics for a cut of their wages, making for a very different experience for the contributors than the developers imaged.

Tom Deutsch then looks at how our neurobiology plays a role in our digital interactions, specifically as it relates to privacy, and how human computations systems need to consider those dynamics to meet their own goals. Elena Ferrari and Marco Viviani look the evolution of privacy from offline to online communities and the impact for online collaboration systems. Finally in my chapter, I look at the risks involved in human computation systems. What are the assets worth protecting? How can those assets be degraded or lost? Since you never get enough budget to secure all your assets, you need to make sure the most valuable assets get protection.

The history of computer science has shown that when developers explore a new area of using computers, developers concentrate on the exciting new features and not security. Often security does not get integrated until someone suffers. With this new area of human computation we have another chance to get security right before people accept poor security practices as the norm. We can still prevent catastrophic failures in human computation by designing in the correct balance of security mechanism and policy to aid finding new solutions.