

Research Summaries CS24

Christine Chen

May 8, 2015

1 Human-Centered Computer Security and Privacy

This TRUST Security Seminar focused on the idea that technology is rapidly expanding throughout the world and becoming integrated in almost every aspect of our lives. Even though this is a great achievement, it poses a threat to the livelihood of many, possibly compromising personal privacy and security. Tamara Denning is interested in the human aspect of technology and computer security. The four areas that she focuses on are how people can be hurt by technology, what predators want from users, how we can better design systems to protect users, and how we can inform the public and explore more preventive measures. Technology poses a real threat to people's privacy; some examples include robots eavesdropping on an av feed, spying on users, tripping the elderly, etc. Predators usually try to gain access to people's private information ranging to passwords to health issues that may be gained through wireless technology to be used against them in some way. One example that Denning used was cardiac defibrillators and that they need more security. Apparently it is all too easy for someone to obtain private information about a patient including the serial number of the heart defibrillators, the patient's name, and diagnosis. Denning's

presentation really opened up my eyes to the real threats that technology pose to the personal privacy to each and every one of us. There should be more preventive measures taken by developers to ensure that users aren't prone to sharing their information accidentally. With that said, it is a difficult task to formally educate everyone on the preventive measures that can be taken even if we have educators, developers, and researchers all working towards the same task to preserve the privacy and security of each and every user of a product.

2 Time Travel Debugging

This seminar presented by Mark Marron centered around the idea of Time Travel Debugging. Traditionally people usually only are aware of the step forward functionality and hardly use the step back function. The step back function is not widely used because many of the ones available are too slow to be practical to use on a day to day basis. Marron continued to present that maybe if they were more efficient, it could be more widely utilized to better identify bugs in programs faster. He suggested that if the software had a minimal performance overhead less than 25 percent, have low space overhead, and a small pause to start the time-travel that it would be more usable. One of the ways that he proposed was instead of trying to reverse the execution of the program, the software would take snapshots of the entire program state at regular intervals and save them to .json file. This was really cool to hear because in our project 2, we built gitlet a naive implementation of git, and used serializable to save the state of modified files so I could understand what he was talking about. These snapshots were efficient enough to meet the requirements that Marron set out initially in the presentation. One of the ideas that I did not quite understand was the relationship between the user and a web page and how certain user inputs at specific points could cause chaos. This new approach for debugging

has a lot of potential for various uses in computer science. Some include being an enabler for other research, interrogative debugging, fault localization and bug history analysis, fault recovery and process migration. I personally have not had much experience with debuggers, but I have heard that it saves a lot of time and headache.