Building the Security Behavior Observatory: An Infrastructure for Long-term Monitoring of Client Machines

Alain Forget[a], Saranga Komanduri[b],
Alessandro Acquisti[c], Nicolas Christin[d], Lorrie Faith Cranor[e], Rahul Telang[f]
Carnegie Mellon University
[a]aforget@cmu.edu, [b]sarangak@cs.cmu.edu, [c]acquisti@andrew.cmu.edu,
[d]nicolasc@cmu.edu, [e]lorrie@cmu.edu, [f]rtelang@andrew.cmu.edu

ABSTRACT

We present an architecture for the Security Behavior Observatory (SBO), a client-server infrastructure designed to collect a wide array of data on user and computer behavior from hundreds of participants over several years. The SBO infrastructure had to be carefully designed to fulfill several requirements. First, the SBO must scale with the desired length, breadth, and depth of data collection. Second, we must take extraordinary care to ensure the security of the collected data, which will inevitably include intimate participant behavioral data. Third, the SBO must serve our research interests, which will inevitably change as collected data is analyzed and interpreted. This short paper summarizes some of our design and implementation benefits and discusses a few hurdles and trade-offs to consider when designing such a data collection system.

1. INTRODUCTION

Our understanding of computer and user behavior, with respect to security and privacy, has largely been based on studies of short duration and narrow focus which may lack ecological validity [2] or not reflect the users' actual behavior [1, 3]. These studies have helped guide research over the past 20 years. However, a large-scale field study permits the measurement of users' security and privacy challenges and behaviors with much greater ecological validity than in the lab, where the experimental setting might not reflect users' actual behavior in their natural environment [4]. Furthermore, a long-term longitudinal study would reveal how frequently users encounter security and privacy issues. These frequencies represent risk probabilities, which are a key element of any risk assessment or risk management strategy.

2. SECURITY BEHAVIOR OBSERVATORY

To fill this need for more ecologically-valid data, we are building the Security Behavior Observatory (SBO): a client-server architecture where participants' client computers are monitored over an extended period of time and upload collected user and computer behavior data to our servers. Examples of the data we intend to monitor from hundreds of client machines over several years, with IRB approval and under strict security and privacy safeguards. Our architecture is designed to provide data covering as much of the security and privacy space as possible. Some example research questions we intend to examine include: How up-to-date are operating systems? How long before a clean machine is infected, and how does it actually occur in the wild? What warning dialog messages do users encounter most often, and how do users respond? What are users' online social network privacy settings? Do they ever change?

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.
Copyright is held by the owner/author(s).
Symposium and Bootcamp on the Science of Security (HotSoS) '14, April 08 - 09 2014, Raleigh, NC, USA
Copyright 20XX ACM 978-1-4503-2907-1/14/04.
http://dx.doi.org/10.1145/2600176.2600200 ...$15.00.

3. ARCHITECTURE

Each type of data (e.g. network packets, file system contents, processes) is collected and output by our client-side *sensor*. Our client communication module periodically compresses, encrypts, and sends the data files over an SSL-encrypted channel to the *data collection server*. A few benefits of our client software design include:

Silent updates. We package all the client software components into a single installation executable. This provides functionality for cleanly installing and uninstalling the software, as well as silently upgrading the software.

Independent sensors. Each type of data is collected by a software *sensor* that is independent of the rest of the data collection system.

Least privilege. Some data we seek to collect is likely to require administrator access to the client system. Fortunately, our architecture's sensors are independent, so higher privileges can be given only to sensors that require them.

Minimal footprint. Since the study's primary goal is to *observe* computer users' typical behavior, we must take care to avoid experimental effects that may influence this behavior. Thus, users should not notice a decrease in computer or network performance during the study.

Figure 1 illustrates the deployment of our server architecture we believe most securely and efficiently collects data from participants, and allows researchers to access the data with as little inconvenience as possible.

Data collection server. The data flow from clients to the data collection server proceeds as follows. Data is continuously generated on client machines. At regular intervals, each client establishes an SSL connection to the data collection server. The client and server mutually authenticate each other by encrypting random numbers with a shared symmetric authentication key. When the server is ready to

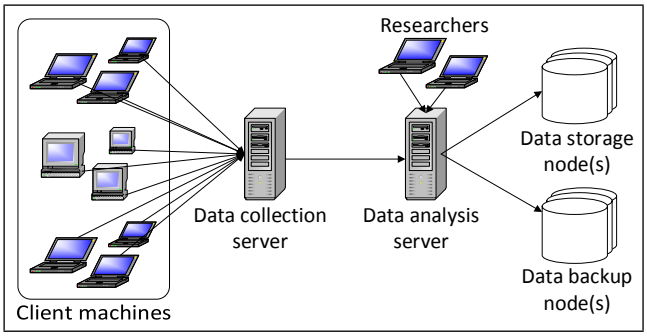


Figure 1: Our SBO high-level hardware architecture.

receive data, the client encrypts the compressed data, and sends it to the server. The server stores the data locally, still encrypted with the client's encryption key.

Data analysis server. Through a mutually-authenticated SSL connection, the data analysis server periodically moves the encrypted data from the data collection server to the data storage node(s). This server also provides researchers access to the data. The data must remain solely on the data analysis server and be accessible only to project administrators and researchers.

Data node(s). Participants' encrypted data is ultimately stored in two places; in the data storage node(s) and data backup node(s). The backup node(s) are located in a physically-separate building from the storage nodes. These nodes are accessible only through the data analysis server.

4. DISCUSSION

We now discuss a sample of issues warranting careful consideration when executing such a study.

Ethics and participant privacy. Although true for all user studies, it is critical that an institutional review board (IRB) approve the study's methodologies and procedures to ensure participants' are treated ethically and their data is kept confidential and secure. We spent considerable time iterating over our consent procedures with our IRB before their approval. However, many review boards do not have the expertise to understand the specific security and privacy challenges that may arise. Thus, the burden lies on the experimenters to consider carefully which data they are willing to collect and hold in trust, and to weigh the risk of a compromise with the value of such data to the advancement of the community's knowledge. Regarding deidentification, participants are assigned a random ID, which decouples their uploaded data from their provided personal information. We are also considering additional anonymization strategies and weighing their costs (e.g., loss of data richness, client-side computational loads) against possible threat models (e.g., client, network, server attacks).

Data Security. Given the sensitivity of the data we collect and transmit across the Internet and store on our servers, the data's security and confidentiality must be carefully considered and strictly enforced. Every client is assigned a unique key for encrypting the data before it is sent to the server through an SSL connection and stored. Although methods for computing on encrypted data exist (e.g., homomorphic encryption), our analyses across multiple sensors' data longitudinally across time are likely to be complex enough that they would not be practically feasible with such solutions. Instead, one researcher with access to all the clients' keys will decrypt and decompress each client's data into a TrueCrypt volume, to which all project researchers will have the key to analyze the data. Unencrypted data may temporarily exist in memory while and after working with it. However, the data must remain on the data storage nodes, which can be accessed only through a secure shell to the data analysis server from the specific IP addresses of the researchers' own campus machines. No other connections to this server are permitted.

Client upload bandwidth Given the wide breadth and depth of data we ideally wish to collect, we must restrict the amount of data we upload from participants' machines to avoid a noticeable reduction of their Internet connection bandwidth. We throttle clients' data upload speed by interleaving data transmission and sleep commands, which should cause the operating system to flush the uploaded data stream and free the network bandwidth and processing cycles for other applications until our application resumes. We also employ techniques to minimize the physical size of our data. Sensors that perform periodic snapshots only log *differences* between the previously-recorded and current state, rather than always logging the complete current state. Our data logs use the Binary JSON data format to minimize spatial overhead and be efficient to encode and decode. Even with such techniques, we anticipate difficult decisions about which types of data to prioritize. For example, while performing our preliminary data analysis, we may observe phenomena that we wish to further explore, but be unable to since we lacked the bandwidth to collect the required data.

5. CONCLUSION

Capturing data on the privacy and security challenges users face as they occur in the wild is essential if we are to conduct research and foster innovations with the greatest impact in improving the security and privacy of users and their machines. The Security Behavior Observatory (SBO) aims to collect this highly ecologically valid data on multiple security and privacy topics from hundreds of users' home computers over several years. We hope the data collected will yield insights on a wide variety of security and privacy challenges, and guide future research efforts towards solving the challenges users actually face in the wild.

6. REFERENCES

[1] B. Berendt, O. Günther, and S. Spiekermann. Privacy in e-commerce: Stated preferences vs. actual behavior. *Communications of the ACM*, 48(4), April 2005.

[2] M. Brewer. Research design and issues of validity. *Handbook of research methods in social and personality psychology*, pages 3–16, 2000.

[3] A. De Luca, M. Langheinrich, and H. Hussmann. Towards understanding ATM security – a field study of real world ATM use. In *Symposium on Usable Privacy and Security (SOUPS)*. ACM, 2010.

[4] A. Sotirakopoulos, K. Hawkey, and K. Beznosov. On the Challenges in Usable Security Lab Studies: Lessons Learned from Replicating a Study on SSL Warnings. In *Symposium on Usable Privacy and Security (SOUPS)*. ACM, 2011.