Computer Science 758 – Project Proposal

Alexander Pokluda

June 22, 2012

Introduction

I am currently working with a number of other University of Waterloo researchers on a peer-to-peer web hosting project called pWeb. We have already laid many of the theoretical foundations for the system including how content will be identified, stored and retrieved. One of the specific aspects of the pWeb system that I am looking at is dynamic web page generation. This encompasses a number of challenges. For instance, we would like dynamically generated web pages to be able to access persistent storage to facilitate the development of applications such as blogs, comment pages, and wikis.

Our focus right now is to bring all of our ideas together and develop a robust software implementation. BitTorrent is a simple yet highly successful peer-to-peer system and we are using it as as example to guide our implementation. The integrity of the BitTorrent system is ensured by the extensive use of hashes as all content is identified and validated using hashes. However BitTorrent also has a number of limitations that must be overcome in the context of a web hosting platform. For instance, it is impossible to verify the publisher of content in BitTorrent and once content has been published it can never be updated or deleted. pWeb has many unique challenges that can be solved through the well-advised application of cryptography and cryptographic principles. The next two sections give a brief overview of how the pWeb system will function and describe the work that will be completed for this project.

Background

The pWeb application will consist of a lightweight background process that communication with other nodes using the pWeb protocol and will optionally provide a system tray icon and control panel to enable the user to configure the system. The application must be resource efficient and scalable so that it makes full and efficient use of the hardware it is running on whether it is a resource constrained embedded system or powerful many-core server. Thus the cryptosystems used by the application must provide adequate security without compromising the performance of the system. To give you a basic idea of how the system will function, here is a brief description of how Alice would publish her homepage to pWeb:

1. First, she must generate a unique pWeb ID for her site. This could be a SHA-1 hash of her existing domain name, alicehomepage.net, her email address, her public encryption key, or something else.
2. Next she generates meta-data for all of the files in her web site. This basically entails concatenating a revision number, filename, and the contents of each file and then generating a cryptographic signature of the whole thing.
3. Alice tells her pWeb client to publish the web page. To do this, the pWeb application uses its DHT routing table to contact the node(s) responsible for Alice's website pWeb ID. She inserts a record into the DHT to direct people looking up her pWeb ID to contact her pWeb client.
4. Alice's pWeb client uses the DHT routing mechanism to discover peers with complimentary

uptime patterns. Her client then selects a subset of these peers such that the probability that at a specified number of peers in this group will be on-line at any given time is above a defined threshold.

5. Alice's pWeb client will contact each of these peers and ask them to also host a copy of Alice's home page. Each peer that receives these files verifies Alice's signature, and if it is valid, the will accept and follow a similar process.

Other maintenance processes will take place to ensure that the availability of content keeps up with demand and stale content is removed from the system, and the system will be extended with additional functionality, but the above sequence captures the most fundamental operations.

Proposed Work

For this project, I will identify all of the instances where cryptography would be desirable or necessary to prevent abuse in pWeb and evaluate the merits of various solutions in the pWeb context. More specifically, I intend to carry out three specific tasks:

1. Identify all of the security related requirements for the core pWeb system then identify the cryptographic protocols that meet those security requirements (for example, key distribution or entity authentication). I expect to identify several discrete problems that can be solved by applications of cryptography.
2. Survey the literature to identify approximately two to three cryptosystems to solve each problem that meet the security requirements. I will list the pros and cons of each cryptosystem and select the best candidates to solve each problem.
3. For each of the cryptosystems selected in part 2 above, I will either implement the cryptosystem myself in C/C++ or find an implementation in a commonly used library (such as OpenSSL) and compare the performance of each. Based on my findings, I will recommend a specific cryptosystem (and possibly a specific implementation of that cryptosystem) for each problem.

Conclusion

I believe this project is a great opportunity for me to make a contribution to both the field of cryptography as well as the pWeb project. As mentioned in Proposed Work, my contributions to the field of cryptography will be evaluating the applicability of different cryptosystems to a real-world problem. I may also be implementing and evaluating the performance of specific cryptosystems for the first time. It is possible I may end up selecting a “new” cryptosystem for use in pWeb that has not been used in a real-world software project before. In a sense, this will validate the cryptosystem as practical and bring additional attention to the cryptosystem and its authors—although this is outside of the scope of this project.