CSE548 – Advanced Computer Network Security

Project Deadlines

|  |  |  |
| --- | --- | --- |
| Item | Deadline | Weight |
| Project Group and Registration | 1/29 | 0% |
| Project Proposal | 2/10 | 7.5% |
| Interim Report | 3/10 | 7.5% |
| Final Project Report | 4/7 | 15% |
| Demo Window | 4/7 – 4/21 | 10% (total 40% of grade) |

Project Proposal 7.5% (Due 2/10)

Project proposal preparation guide:

1. Format a. Word template is provided (note that you MUST use the provided template do not change the font size and margin, etc.) Without following the provided format will cause deduction of your report grade.

2. Proposal is limited by 3 pages providing detailed plan.

3. If you prefer using Latex, please let me know.

Project Ideas

1. Developing traffic dispatching/load balancing solutions with security policy control based

on flowvisor

2.  Developing audit and report services based on syslog, sflow, snort, etc.

3.  Developing a secure collaborative data sharing system based on GIT, SAN, and Openstack.

4.  Developing a certificate management system to support network supporting system such as

DNSSEC

5.  Developing a Moble VPN solution considering continuous network connectivity

6.  Developing a Wireless traffic monitoring system

7.  Developing an identity management framework (based on database & MobiID, Oauth) and

create APIs to interface various network services

8.  Developing a traffic virtualization/visualization system by using mirroring functions such as

SPAN/ESPAN

9.  Developing a secure signaling/control system based on XMPP and REST

10.  Developing efficient Mobiguard security solutions including IP/certificates verifier,

phishing site checker

11.  Developing an intrusion detection system based on vulnerability databases

**Project 1: Attribute-based secure XMPP group messaging system in cloud environment**

**Mentor:** Bing Li (bingli5@asu.edu)

**Project Description:** This project aims to realize an attribute-based secure group messaging system for XMPP in cloud data sharing environment. It is based on Attribute-Based Encryption (ABE). The idea is that each cloud user is assigned with a set of attributes, which are used as cryptographic keys for encryption. The group needs to use ABE encryption/decryption solutions to construct a secure group messaging system for cloud.

The group needs to get familiar with ABE solutions (<http://hms.isi.jhu.edu/acsc/cpabe/>), and XMPP protocol. The ABE toolkit takes attributes and other parameters from the command line and use them to generate the corresponding cryptographic results. You can use the toolkit directly by constructing the correct commands and call the shell in your project. Only users with satisfactory attribute combinations can successfully decrypt the ciphertext.

In your demo, each group member share a set of common attributes. When a group wide chat is in process, you can encrypt the text with the attribute combinations shared among the group so that users without authentic attribute combinations cannot recover the plaintext. You may need to modify the code in XMPP server (Openfire is recommended) so that the ABE module can seamlessly incorporate into the system. The client can be created using Smack APIs. You also need to manage the attributes properly. Status of contacts and group formations should be supported in the client UI.

Expected outcome includes, but not limit to, a report on the design of architecture, the realization of a prototype demo, and corresponding source code.

**Project Setup Environment:**

·      Existing ABE solutions can be used directly or indirectly;

·      Virtual machines, virtual networks, own cloud and xmpp services.

**Project deliverables:** You need to create a group messaging system based on ABE solutions with a demo scenario. The system should be able to show the status of the users, and support the secure attribute based group messaging communication. The size of the group should be no smaller than 3. The design of the protocol, the source code for both the protocol and the demo, and a report recording important procedures are required.

**Project Skill Needed:** Programming language: C/C++, Java; Operating System: Linux.

**References:**

[1]. ABE Toolkit: <http://hms.isi.jhu.edu/acsc/cpabe/>

[2]. XMPP standard: <http://xmpp.org/>

[3]. Openfire XMPP Server: <http://www.igniterealtime.org/projects/openfire/index.jsp>

[4]. Smack XMPP API: <http://www.igniterealtime.org/projects/smack/index.jsp>

ABE Implementation:

http://hms.isi.jhu.edu/acsc/cpabe/#documentation