**CS 6349 Network Security Fall 2018**

**Programming Project**

**Implementation of a Real Time e-Commerce Broker System**

**Project Draft**

In this project, you will implement an e-Commerce broker system which will facilitate anonymous online purchase process between a Client and an e-Commerce Website.

**Main components in the system:**

1. Client
2. Broker System
3. E-Commerce Website

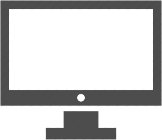
**The system will ensure the following rules:**

1. Client can browse to different e-Commerce site through Broker System.
2. Client’s profile as well as location is hidden from the e-Commerce site.
3. Broker System knows the Client and e-Commerce site but does not know about the purchased e-Product (music, movie or application) or its contents.
4. Broker System pays for the e-Product to the e-Commerce site.
5. Broker System has sufficient information about a purchase transaction (but not what is purchased) by a Client. This is to protect against non-repudiation.

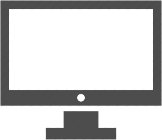
**Goal of the project:**

1. **Authentication:** Each of the components should be able to authenticate each other with the exception that e-Commerce sites do not need to authenticate clients since they do not know their identity.
2. **Message Integrity/Consistency**: Integrity of the message transmitted between different components through insecure network should be ensured.
3. **Privacy/Confidentiality:** Ensure confidentiality in the communication processes between (1) Client & Broker; (2) Broker & E-Commerce site; and (3) Client & E-Commerce site.
4. **Anonymity:** Client’s identity should be hidden to e-Commerce site.
5. **Non-repudiation**: Broker can prove that it is the Client who authorized the payment to the E-Commerce site.

**Figure 1:** Anonymous online purchase system



**Client:** Alice, Josh  
**Broker:** Paypal  
**Web Server:** Amazon, iStore, Google Play



Amazon

iStore

Google Play

PayPal

Alice

Josh

**A purchase scenario:**

Suppose Alice wants to purchase a softcopy of the book, Gray Mountain, from Amazon. For this, the following steps will be executed:

1. Alice will initiate the process by informing PayPal that she wants to browse Amazon website.
2. PayPal will create a connection with Amazon.
3. Alice and Amazon will communicate with each other through PayPal in such a way that PayPal cannot decrypt any information.
4. Amazon will send the product catalog file to Alice.
5. Alice will choose the product and will tell PayPal to pay $X to Amazon.
6. PayPal will store the payment order from Alice to its database and will pay Amazon.
7. Amazon will send the product to Alice over the tunnel through PayPal.

All communication should be encrypted. For encryption, each of the components should use session keys which are valid for only one session. The system should hide Alice’s identify and location from Amazon. PayPal should not be able to decrypt any information from the communication between Alice and Amazon. Furthermore, PayPal should keep a record about the payment order which can contain client ID, unique payment ID generated by PayPal, purchase date and webserver name. Using the record, PayPal should be able to verify in future that the payment was authorized by Alice.

**Group Project:**

For this project, you may work in groups of up to 3 students. Each team is required to submit a plan-of action document that will include details about the implementation of the project as your team interprets it. Each team will present their presentation as a discussion with the TA to ensure that they have understood the scope, goals and expectations of this project.

**NOTE:**

1. The deadline for submission of the plan-of-action document is October 21, 2018. This document will carry 10% of the project grade.
2. You can use C/C++, Python or Java for the implementation of this project. You may use existing cryptography libraries of these programming languages but cannot use secure communication services such as SSL.
3. Project submission deadline is December 1, 2018.
4. Project demo and grading information will be announced later.

**Submission guideline:**

1. Each team has to submit their source code and a readme file.
2. You have to show the working of at least one client, one Broker and two Webserver.
3. Display the messages sent and received on each component.
4. After downloading the file from the server, Client needs to show the contents of the file locally.
5. Each team member should be able to answer the questions regarding the code and the design protocol.