RMIT UNIVERSITY

School of Science

INTE1070/1071 Secure Electronic Commerce

Assignment

	Assessment Type: Individual assignment; no group work. Demonstration of prototype and understanding is required (in-lab). Submit online via Canvas→Assignments→Assignment. Marks awarded for meeting requirements as closely as possible. Clarifications/updates may be made via announcements/relevant discussion forums.
	Due date: **11:59pm, 18/Oct/2019**; Deadlines will not be advanced but they may be extended. Please check Canvas→Syllabus or via Canvas→Assignments→Assignment for the most up to date information. As this is a major assignment in which you demonstrate your understanding, a university standard late penalty of 10% per each working day applies for up to 5 working days late, unless special consideration has been granted.
	Weighting: 10 marks

1. Overview

Electronic Commerce has become a part of today's Internet-based economy. In line with that, Secure Electronic Commerce embodies a concept for doing reliable business online. It includes shopping and marketing products or goods through secure business-to-business transactions or events. This course is an introduction to secure e-commerce, from the principles and concepts to practical examples. The objective of this assignment is for you to gain a first-hand experience on how the security theories introduced in lectures are applied in the digital world.

Assume that you work in a team assigned to develop an online e-commerce (shopping) system using HTML, JavaScript, and PHP programming languages. You are free to choose any product for sale. There are several parts of the application, including the following functions with proper security guarantees (detailed in section 4.1 below):

1. Registration (Signing up) interface of the website, and keep the username and hashed password in database;

2. Login to the website using the credentials used for registration;

3. Shopping cart page accessible after successful login with update feature;

4. Post shopping cart information and credit card number to database server after encryption (with RSA and DES).

Note: You must not just “throw in the concepts” to your programs just because they need to be there; it should be clear from the code why a certain concept should be there and you must further explain these through your comments. You will also need debug your code on your own and document any issues, etc. You are given marks on your ability to fulfill all requirements of this document.

There are implementation requirements (9 marks) and documentation requirements (1 mark) for a total of 10 marks.

Develop this assignment in an iterative fashion (as opposed to completing it in one sitting). You can get started as soon as the concepts are introduced in lessons.
If there are questions, you must ask via the relevant Canvas discussion forums in a general manner (replicate your problem in a different context in isolation before posting).

2. Assessment Criteria

This assessment will determine your ability to:

1. Understand the concepts and techniques addressed in the lectures, tutorials and practical.
2. Implement the relevant encryption algorithms with HTML, JavaScript and PHP programming languages.
3. Write and debug the program independently.
4. Demonstrate the prototype properly.

5. Document the prototype.
6. Ability to provide references if necessary.
7. Meeting deadlines.
8. Seeking clarification from your “supervisor” (instructor) when needed via discussion forums.
9. Create a program by recalling concepts taught in class, understanding and applying concepts relevant to solution, analysing components of the problem, evaluating different approaches.

3. Learning Outcomes

This assessment is relevant to the following Learning Outcomes:

1. Explain the range of threats to e-commerce security.
2. Explain how cryptography can be, and is, used to achieve security.
3. Describe the different standards in use for secure electronic commerce, such as certificates, MACs, etc.
4. Describe and analyse standard security mechanisms, such as filters, proxies and firewalls.

4. Assessment details

Note: Please ensure that you have read sections 1-3 of this document before going further.

Your prototype must meet the following implementation requirements (section 4.1) and documentation requirement (section 4.2); also refer to corresponding rows in the rubric (section 9).

4.1) Develop an online e-commerce (shopping) system using HTML, JavaScript and PHP. **(totally 9 points)**

- **C1 - Registration and Login function (1 point)**
- **C2 - Shopping cart function (1 point)**
- **C3 - The function of posting shopping cart information and credit card number to Server (5 points)**
- **C4 - Assignment demo + questions by lab instructor (2 point)**

Deploy your system under the directory titan.csit.rmit.edu.au/~sXXXXXXX/assignment/ with the correct permission, with the main folder “assignment” and three subfolders “server”, “client” and “database” (more detail please see section 6).

C1. For the registration and login function **– up to 1 point**

- Do not allow register if entered password is less than 6 characters **(+0.25 points);**
- Enter plain password is hashed before register **(+0.25 point);**
- Save username in the database and hashed password in the database **(+0.5).**

C2. For the shopping cart function **– up to 1 point**

- Only successful login user can access to the shopping cart **(+0.5 point),**
- and the quantity of each item in the shopping cart can be updated **(+0.5 point).**

C3. For the function of posting shopping cart information and credit card number to Server **- up to 5 points**

- If you post plain shopping cart and credit card to server and display **(up to 1 point if you did this for C3);**
- If your post encrypted information and display plain information – RSA only **(up to 2.5 points if you did this for C3);**
  - Credit card number is encrypted with RSA encryption algorithm **(+1 point),**
  - and the server decrypts the information with RSA decryption algorithm and stores it in the database **(+1.5 points);**
- If you post encrypted information and display plain information - RSA and DES (**up to 5 points if you did this for C3**):
  - Credit card is encrypted by DES key (user entered before submission) **(+1 point)**
  - DES key is encrypted by RSA public key of server **(+1 points),**
  - Server reveal the DES key with server’s RSA private key **(+1.5 point),**
  - Server reveal user’s credit card by using revealed DES key **(+1.5 points).**

C4. Ask questions during Assignment demo **– up to 2 point**

- Lab instructor can ask you to run any specify function **(+1 point).**
- Lab instructor can ask you to show any part of code and explain **(+1 point).**

In places where this specification may not tell you how exactly you should implement a certain feature, the programmer (you) need to use your judgment to choose and apply the most appropriate concepts from class materials. Follow answers given by your “supervisor” (you instructor) under Canvas→Discussions→’Assignment’ when in doubt.

4.2) Documentation requirement **(up to 1 point)**

D1. Write a report to describe what you have done and what you have observed with **screen shots** whenever necessary.

Recommended report format:

- Title, student name and id.
- Report is recommended to have a scenario of e-commerce with screen-shots containing the explanation.

5. Referencing guidelines

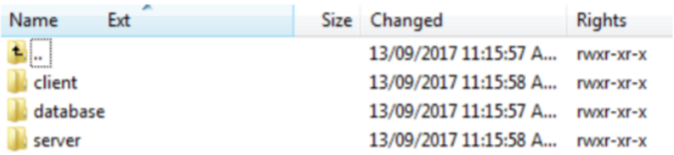
What: This is an individual assignment and all submitted contents must be your own. If you have used sources of information other than the contents directly under Canvas→Modules, you must give acknowledge the sources and give references using IEEE referencing style.

Where: Add a code comment near the work to be referenced and include the reference in the IEEE style.

How: To generate a valid IEEE style reference, please use the citethisforme tool if unfamiliar with this style. Add the detailed reference before any relevant code (within code comments).

6. Submission format

Please deploy your website under the folder “assignment” with three subfolders “client”, “server” and “database” as follows:



Name	Ext	Size	Changed	Rights
..			13/09/2017 11:15:57 A...	rwxr-xr-x
client			13/09/2017 11:15:58 A...	rwxr-xr-x
database			13/09/2017 11:15:57 A...	rwxr-xr-x
server			13/09/2017 11:15:58 A...	rwxr-xr-x

We will assess your assignment on the basis of the website under the following link: **titan.csit.rmit.edu.au/~sXXXXXXX/assignment/** where the sXXXXXXX is your student number. Please make sure to put everything (with correct permission) under the folder “assignment”.

Demonstration of all tasks in this assignment is required. Following the assessment details (section 4) and the rubric (section 9) to demonstrate your prototype to your lab instructors. Explaining your understanding and observations whenever necessary.

Submit **a .zip file** via Canvas→Assignments→Assignment. The submission has to include your programs, and a report to describe what you have done and what you have observed with **screen shots** whenever necessary. It is the responsibility of the student to correctly submit their files. Please verify that your submission is correctly submitted by downloading what you have submitted to see if the files include the correct contents.

7. Academic integrity and plagiarism (standard warning)

Academic integrity is about honest presentation of your academic work. It means acknowledging the work of others while developing your own insights, knowledge and ideas. You should take extreme care that you have:

- Acknowledged words, data, diagrams, models, frameworks and/or ideas of others you have quoted (i.e. directly copied), summarised, paraphrased, discussed or mentioned in your assessment through the appropriate referencing methods,
- Provided a reference list of the publication details so your reader can locate the source if necessary. This includes material taken from Internet sites.

If you do not acknowledge the sources of your material, you may be accused of plagiarism because you have passed off the work and ideas of another person without appropriate referencing, as if they were your own.

RMIT University treats plagiarism as a very serious offence constituting misconduct. Plagiarism covers a variety of inappropriate behaviours, including:

- Failure to properly document a source
- Copyright material from the internet or databases
- Collusion between students

For further information on our policies and procedures, please refer to the University website.

8. Assessment declaration

When you submit work electronically, you agree to the assessment declaration.

. Rubric/assessment criteria for marking

Code must be valid, runnable HTML, JavaScript, and PHP to be given a mark. Run-time errors will incur up to a 50% penalty (run-time errors due to data type mismatches in inputs are acceptable).

	Inadequate	**Partial**	**Complete** (Uses only the concepts covered in class materials for meeting stated criteria)
C1	Incorrect implementation of given task as instructed. If the solution does not match the requirements.	Keep username and plain password in the database. If the solution contains a password storing technique that only accepts plain password.	Should keep username and hashed password in the database. If the solution reflects the hashed password storing technique, and complete requirement as discussed in 4, the assessment would be regarded as complete one.
C2	More than one of the criteria in the ‘complete’ level missing/incorrect	Username and plain password are Posted to server for checking. If the solution contains a password storing technique that only accepts plain password.	Should maintain the Username and hashed password are Posted to server for checking. If the solution reflects the hashed password storing technique, and complete requirement as discussed in 4, the assessment would be regarded as complete one.
C3	Missing/incorrect implementation that does not satisfy the rubric requirement as outlined for a task to be considered as complete.	Only successful login user can access to the shopping cart. For example, if a solution submitted contains program that allow a valid user to login before accessing the shopping cart page without update or modification features.	The solution should be capable of updating each item; e.g. The quantity of each item in the shopping cart can be updated. For illustration, the solution should reflect the respective requirement following proper instructions. The submitted program should have shopping cart update feature as if a valid user can change or modify the items s/he selected previously.
C4	More than one of the criteria in the ‘complete’ level missing/incorrect. If the solution does not match the requirements.	Post plain information and display only plain information or Post encrypted information and display plain information only using either RSA or DES.	A successful login user posts an encrypted DES key (chosen by the user) to the server with RSA encryption algorithm. The server retrieves the DES key with RSA decryption algorithm and keeps the DES key for this user. The user will encrypt the shopping cart and credit card number with DES encryption algorithm and DES key (shared between the user and the server) before Posting to the sever, and the server decrypts the encrypted shopping cart and credit card number with DES decryption algorithm and the shared DES key and stores it in the database.
D1	Report not submitted properly or uploaded after the deadline.	Missing any report item such Screenshots, codes or any other required files	Submission of the report including codes and required screen-shots as zipped folder containing all resources associated.