SCC 311 Distributed Systems Practical 311

Week 3 - Retrieval Exercise (20%)

The goal of this exercise is for you to program a small RMI client in Java that can retrieve the module's main coursework from a running RMI server. To retrieve the coursework, you will need to authenticate yourself with the RMI server. You can choose one of two options to do that, each worth a different amount of mark:

- The username and password option is worth 10 marks if successfully completed.
- The cryptographic option is worth 20 marks if successfully completed.

The registry and the server are hosted on host '**scc311-server.lancs.ac.uk**' **(IP 10.42.72.88) (registry run on the default port; and the name of the server is CW_server**). ***Note that the server is only accessible through the University Campus Network.*** If you are working off Campus, you will need to access the server through the university VPN (http://www.lancaster.ac.uk/iss/services/working-off-campus/).

In the attached archive (SCC_311_CodeRetrievalExercise.zip) you will find three java files to help you code your client:

- The remote server interface: CW_server_interface.java
- The two message classes: Server_response.java; Client_request.java

Note that you will no need to edit/modify any of these three classes to complete the exercise.

The remote CW_server offers three methods:

- `public Server_response getSpec(int uid,Client_request req) throws RemoteException;`
- `public SealedObject getSpec(int uid, SealedObject req) throws RemoteException;`
- `public int getStatus(int uid) throws RemoteException;`

1. The first method corresponds to the password based authentication option. It takes a Client_request (containing your username, password and a random number) and returns a Server_response. Server response offers a method that will allow you to extract the file that's contained within the Server_response. ***NOTE: the specification document is in MS word (.doc) format.***

2. The second method will use encryption to provide authentication and confidentiality. You need to read about SealedObject (http://docs.oracle.com/javase/7/docs/api/javax/crypto/SealedObject.html) to use this method. SealedObject is a Java mechanism that will allow you to encrypt Objects. The method will take a username and an encrypted Client_request (a SealedObject) as it's arguments and returns a "sealed" (encrypted) Server_response.

3. The third method will let you check your status. It returns, 0 if you have not retrieved the coursework spec successfully, 1 if you used password based authentication and 2 if you are successful using cryptographic keys.

Deadline and submission:

IMPORTANT NOTE: Your solution must be formally submitted through Moodle by **Friday (4pm) of week 3.**
You will be asked to demonstrate your work from your Moodle submission.

- The retrieval exercise will be marked during practical sessions of **week 4**.
- **You need to demonstrate your work to a member of the course team**, failing to do so will result in 0 mark.
- **You should leave the lab after marking and should not discuss** the making questions with anyone whose submission has not been marked.
- The demonstration must take place during the session you are allocated to. You cannot come to an earlier or later session for marking.
- You need to come to the marking session with your exercise completed. We will not be able to provide support during marking sessions.
- You should be able to explain what you have done clearly, to show that you understand the concepts introduced.
- Checks for plagiarism and collusion will be carried out on all work.