

**Fall 2006: CS 431/631/731 Distributed Computing
Homework-4**

200 points. Individual Work Only. Due December 6, 2006 before class.

1. Modify the RMI-based client/server program developed in Homework-2 to include secure RMI communication. First test the simple calculator example provided and then update your program. [50 points]
2. **[Graduate Students Only]** Modify the socket-based client/server program developed in Homework-1 to include secure sockets communications. [50 points]

First read the article “Using JSSE for secure socket communication” by Greg Travis from IBM DeveloperWorks (<https://www6.software.ibm.com/developerworks/education/j-jsse/j-jsse-ltr.pdf>) and then test the programs (download the programs from <http://www.cis.uab.edu/cs631/software/jsse-sockets.tar.gz>). Follow the instructions provided to test these programs. Note that this program uses mutual authentication and hence requires key exchanges between client and server. Also this program requires X to display the GUI.

3. Follow the instructions provided to install Axis SOAP container and test the container with the Google Search Client. [50 points]
4. Develop and deploy a web service using the Axis container that will provide the following methods:
 - a. String hashvalue(String passphrase)
Takes a passphrase of type String and returns the corresponding MD5 hash value as a String.
 - b. boolean authenticate(String username, String hashvalue)
Takes username and the MD5 hash value of user’s password and checks if the username and hashvalue matches the corresponding entry in the database.
 - c. void create(String username, String hashvalue)
Takes username and hashvalue and creates an entry in the database.
 - d. void delete(String username)
Deletes the corresponding entry from the database.
 - e. String gethashvalue(String username)
Returns the hashvalue from the database corresponding to the given username.

The database schema is shown below:

USERNAME VARCHAR(32)	HASHVALUE VARCHAR(64)
-------------------------	--------------------------

Test the web service by developing a web interface to register the username and password and then providing a login page to perform the authentication. Use a secure connection (https) to the web server to perform the authentication. See instructions for configuring the Apache Tomcat Web Server for HTTPS. [100 points]

5. **[Bonus Question]** Integrate the user authentication web application developed in question 4 with the web application developed in Homework-3. [50 points]

Fall 2006: CS 431/631/731 Distributed Computing
Homework-4

Sample session for creating a table with PostgreSQL on one of the Vulcan machines:

```
$ psql -h vulcan -U puri
Password: <enter your PostgreSQL password here>
Welcome to psql 8.0.8, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help with psql commands
      \g or terminate with semicolon to execute query
      \q to quit

puri=> create table usertable (username text, hashvalue text);
CREATE TABLE
puri=> \d usertable
Table "public.usertable"
Column | Type | Modifiers
-----+-----+-----
username | text |
hashvalue | text |

puri=>
```

Sample session for configuring Tomcat for HTTPS:

1. Create a certificate keystore

```
$ keytool -genkey -alias tomcat -keyalg RSA
Enter keystore password: changeit
What is your first and last name?
[Unknown]: vulcan8.cis.uab.edu
What is the name of your organizational unit?
[Unknown]: CIS
What is the name of your organization?
[Unknown]: UAB
What is the name of your City or Locality?
[Unknown]: Birmingham
What is the name of your State or Province?
[Unknown]: AL
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=vulcan8.cis.uab.edu, OU=CIS, O=UAB, L=Birmingham, ST=AL, C=US correct?
[no]: yes

Enter key password for <tomcat>
(RETURN if same as keystore password):
$
```

**Fall 2006: CS 431/631/731 Distributed Computing
Homework-4**

2. Remove the comments around the following lines (delete lines 93 and 99 in the original file) in the file \$CATALINA_BASE/conf/server.xml:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<Connector port="8443" maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS" />
```

3. You should also change the port number from 8443 to 9443 or 10443 or 11443 based on the corresponding port number (9090 or 10090 or 11090).
4. Now start Tomcat Server and connect to <https://HOSTNAME:PORTNUM> where HOSTNAME is the assigned hostname and PORTNUM is the corresponding port number ending with XX443.

Working Environment:

Use the CIS machines that are assigned to you to start the web server on the assigned ports (same as Homework-3). There is no need to be physically in the lab to use these machines, you can connect to these machines remotely using an SSH Client. When you are not in the department you have to setup a SSH tunnel for HTTP access to the webserver. Follow the directions given at: <http://www.cis.uab.edu/cs633/SSHTunnelSetup.html> to setup the tunnel (you have change the hostname titanic.hpcl.cis.uab.edu to the hostname assigned to you and also the port # 6060 to your assigned port for the webserver).

Feedback Questions (answer to these questions has no impact on your grade):

- Was this homework too difficult, or too easy?
- Was the assignment fun or challenging?
- Was there something that was unclear?
- Was the homework too long for the given amount of time?
- What did you learn from this homework?

Submission Instructions:

List ALL the references you used in this homework as well as test cases used to test your programs. This includes any classes that you used that you did not write and any help you received from any other sources. Use appropriate class name and include comments to indicate various operations performed by the program. Your program and other documents must have the following header information within comments:

```
/*
Name:
BlazerId:
Homework #:
*/
```

**Fall 2006: CS 431/631/731 Distributed Computing
Homework-4**

Create a tar/zip file of the homework4 directory using the following filename format: blazerId-cs431-hw4.tar or blazerId-cs431-hw4.zip (graduate students use 631 or 731 instead of 431). The tar/zip file must include the source code, txt/word/pdf/ps document for the typed solution to short answer questions and the feedback questions, instructions for executing the programs. Login to WebCT and go to the assignments section (Homework-4) and upload a single tar/zip file (using the browse button under attachments). There is no need to turn in any printed solutions for this homework, WebCT submission is sufficient.

Late Submissions:

Submissions must be made on the due date before the beginning of the class. Late submissions will lose 10% for every 24-hour period, up to a maximum of 50% (weekends and holidays count as one 24-hour period). Any submissions made after one-week will receive a score of 0 for this homework.

Resources:

1. Java Developer's Almanac - Code Samples:
<http://java.sun.com/developer/codesamples/examplets/>
2. Java RMI Tutorial:
<http://java.sun.com/developer/onlineTraining/rmi/RMI.html>
<http://java.sun.com/j2se/1.5.0/docs/guide/rmi/hello/hello-world.html>
3. JSP Tutorial: <http://www.jsptut.com/>
4. Java Database Connectivity (JDBC) Basics:
<http://java.sun.com/docs/books/tutorial/jdbc/basics/index.html>
5. Creating a Keystore to Use with JSSE
<http://java.sun.com/j2se/1.5.0/docs/guide/security/jsse/JSSERefGuide.html#CreateKeystore>
6. Using Custom Socket Factories with Java RMI
<http://java.sun.com/j2se/1.5.0/docs/guide/rmi/socketfactory/index.html>