**PROGRAMA INGENIERÍA DE SISTEMAS**

**GUIA DE LABORATORIO**

**NOMBRE DE LA PRÁCTICA: RMI y SSL**

**OBJETIVO:** **Implementar un método remoto que funcione de forma distribuida utilizando RMI y SSL (Security Socket Layer).**

**OBJETIVOS ESPECÍFICOS:**

Aprender a establecer una conexión segura entre dos partes(cliente-servidor).

Conocer el funcionamiento de Secure Sockets Layer

Aprender como aplicar conexiones seguras SSL implementándolas en RMI

**TEMÁTICAS:**

**1.RMI**

**2.Secure Sockets Layer**

**3.**

**4.**

**5.**

**COMPETENCIAS A DESARROLLAR:**

**1.** Conocer los métodos que permiten ver el funcionamiento de los SSL

**2.** Implementar SSL dentro de RMI

**3.** Establecer comunicación segura entre cliente-servidor

**4.**

**5.**

**CONOCIMIENTOS PREVIOS REQUERIDOS:**

**1.**RMI

**2.**

**3.**

**4.**

**5.**

**RECURSOS A NIVEL HARDWARE Y SFOTWARE:**

**1.** Un PC.

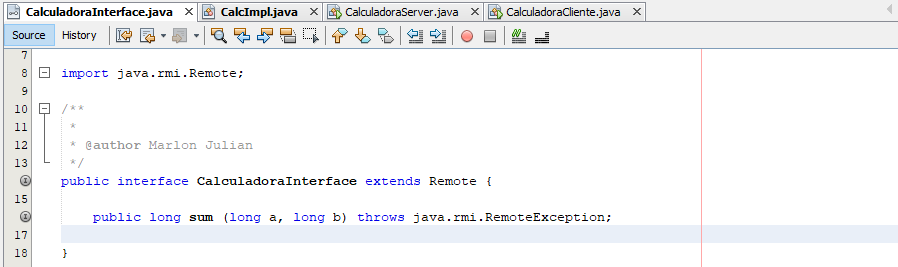
**2.**IDE NetBeans o cualquiera para JAVA

**3. Programa Suma RMI con SSL**

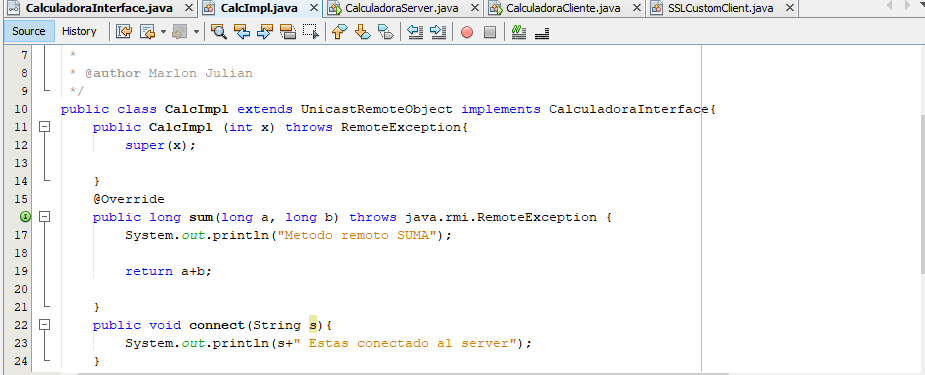
La seguridad se realizo por medio de SocketFactory factory = SSLSocketFactory.getDefault();

Usando un método de socket normal, creamos uno de tipo SSL o socket seguro, el cual utiliza un cifrado RSA de generación de certificados , donde las claves deben coincidir tanto del lado del servidor o como del lado del cliente, y esto se aplica a través de la dirección y del puerto establecido en el RMI.

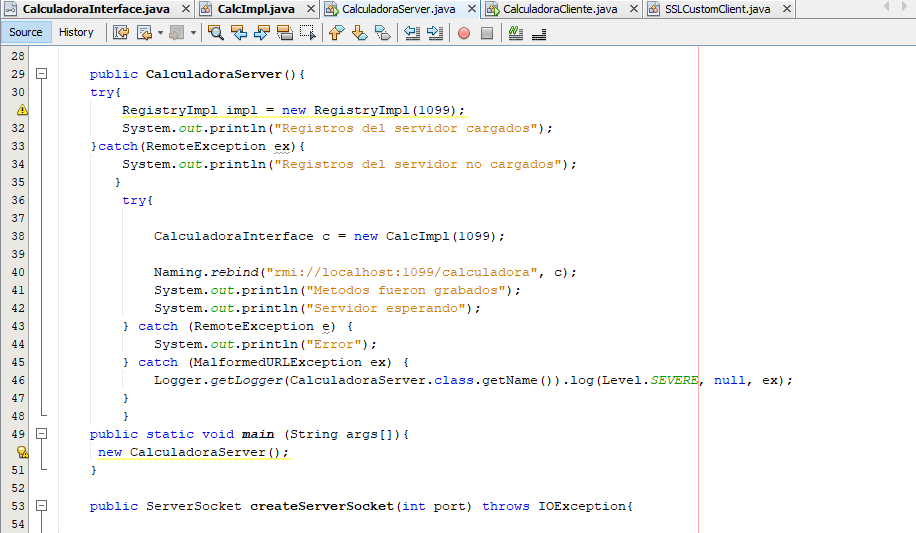
Creación de clase Interface, donde se encuentran los métodos

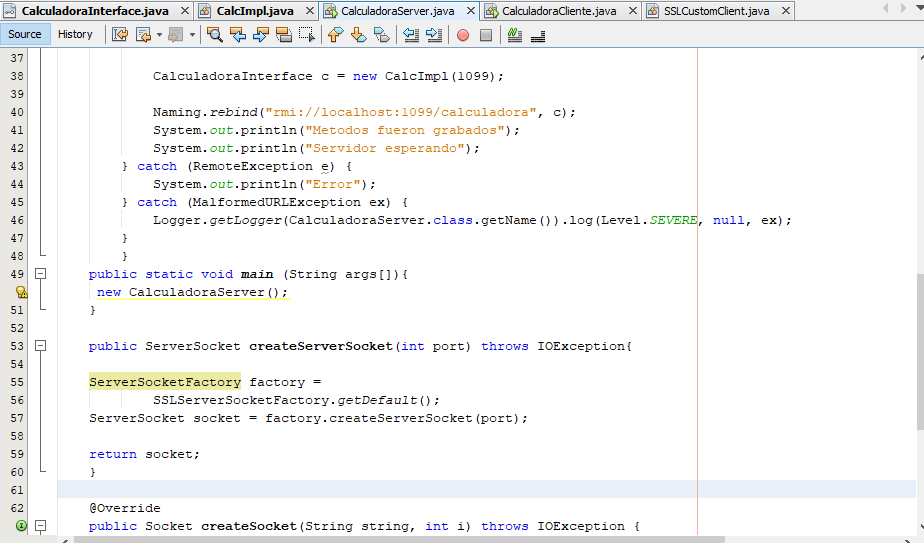


Clase IMPL que se extiende UnicastRemoteObject que viene implementada de la clase Interface.

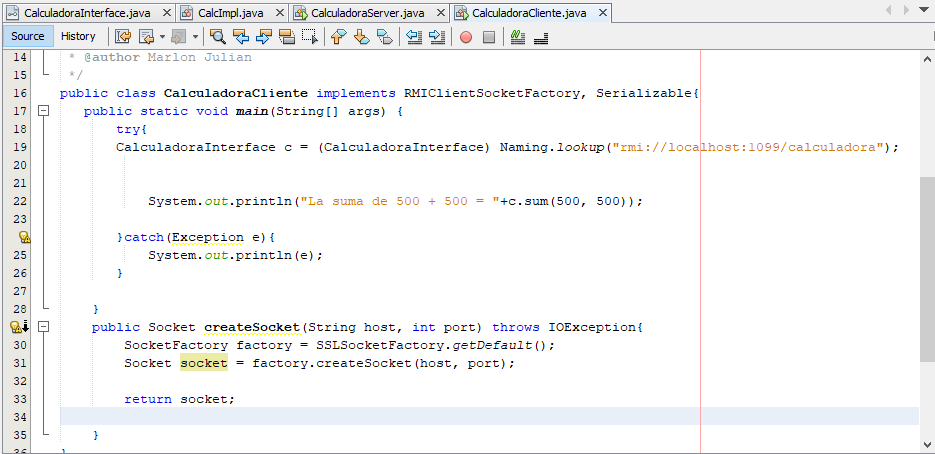


Clase servidor, donde establecemos la URL donde va a quedar almacenado el método, también donde están la capa de sockets segura SSL.





Clase Cliente donde llama a c a través de Interface y luego llama al método suma para implementarlo, este también posee SSL creado en la clase createSocket, acá se realiza la creación de un socket seguro SSL como SSLSocketFactory.



**Código clase Interface**

package ssl;

import java.rmi.Remote;

/\*\*

\*

\* @author Marlon Julian

\*/

public interface CalculadoraInterface extends Remote {

public long sum (long a, long b) throws java.rmi.RemoteException;

}

**Codigo clase Impl**

package ssl;

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

/\*\*

\*

\* @author Marlon Julian

\*/

public class CalcImpl extends UnicastRemoteObject implements CalculadoraInterface{

public CalcImpl (int x) throws RemoteException{

super(x);

}

@Override

public long sum(long a, long b) throws java.rmi.RemoteException {

System.out.println("Metodo remoto SUMA");

return a+b;

}

public void connect(String s){

System.out.println(s+" Estas conectado al server");

}

}

**CLASE SERVER**

package ssl;

import java.io.IOException;

import java.io.Serializable;

import java.net.MalformedURLException;

import java.net.ServerSocket;

import java.net.Socket;

import java.rmi.Naming;

import java.rmi.RemoteException;

import java.rmi.registry.Registry;

import java.rmi.server.RMIClientSocketFactory;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.net.ServerSocketFactory;

import javax.net.ssl.SSLServerSocketFactory;

import sun.rmi.registry.RegistryImpl;

/\*\*

\*

\* @author Marlon Julian

\*/

public class CalculadoraServer implements RMIClientSocketFactory, Serializable{

public CalculadoraServer(){

try{

RegistryImpl impl = new RegistryImpl(1099);

System.out.println("Registros del servidor cargados");

}catch(RemoteException ex){

System.out.println("Registros del servidor no cargados");

}

try{

CalculadoraInterface c = new CalcImpl(1099);

Naming.rebind("rmi://localhost:1099/calculadora", c);

System.out.println("Metodos fueron grabados");

System.out.println("Servidor esperando");

} catch (RemoteException e) {

System.out.println("Error");

} catch (MalformedURLException ex) {

Logger.getLogger(CalculadoraServer.class.getName()).log(Level.SEVERE, null, ex);

}

}

public static void main (String args[]){

new CalculadoraServer();

}

public ServerSocket createServerSocket(int port) throws IOException{

ServerSocketFactory factory =

SSLServerSocketFactory.getDefault();

ServerSocket socket = factory.createServerSocket(port);

return socket;

}

@Override

public Socket createSocket(String string, int i) throws IOException {

throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

}

**CLASE CLIENTE**

package ssl;

import java.io.IOException;

import java.io.Serializable;

import java.net.Socket;

import java.rmi.Naming;

import java.rmi.server.RMIClientSocketFactory;

import javax.net.SocketFactory;

import javax.net.ssl.SSLSocketFactory;

/\*\*

\*

\* @author Marlon Julian

\*/

public class CalculadoraCliente implements RMIClientSocketFactory, Serializable{

public static void main(String[] args) {

try{

CalculadoraInterface c = (CalculadoraInterface) Naming.lookup("rmi://localhost:1099/calculadora");

System.out.println("La suma de 500 + 500 = "+c.sum(500, 500));

}catch(Exception e){

System.out.println(e);

}

}

public Socket createSocket(String host, int port) throws IOException{

SocketFactory factory = SSLSocketFactory.getDefault();

Socket socket = factory.createSocket(host, port);

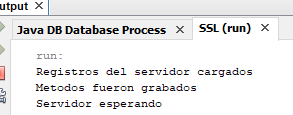
return socket;

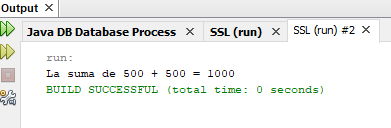
}

}

**PRODUCTOS Y RESULTADOS ESPERADOS:**

Se realiza la correcta conexión entre cliente y servidor, implementando correctamente el marco SSL para la comunicación segura entre cliente y servidor





**CRITERIOS DE EVALUACIÓN:**

1. **Se tiene en cuenta el conocimiento de las definiciones, desarrollo de la práctica, análisis de los resultados, conclusiones y las habilidades, actitud, motivación para realizar la actividad de acuerdo a la rúbrica (Ver abajo).**
2. **Terminación del laboratorio**
3. **Cumplimiento de entrega**
4. **Cumplimiento de los objetivos del laboratorio**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **RÚBRICAS DE EVALUACIÓN** | | | | | |
| **CRITERIOS DE EVALUACION** | **4.5 - 5.0** | **4.0 - 4.5** | **3.0 - 3.5** | **2.5 - 3.0** | **CALIFICACION** |
| **1.- Definiciones** **Conceptos** | No es copia fiel de los textos consultados sino una síntesis de ideas completas y claras del tema. | Excede a cuatro cuartillas o no alcanza a cubrir una.  Algunos párrafos son copias fieles de los textos consultados.  Algunas ideas del tema están cortadas. | No se presenta de manera clara y completa. La relación con el problema planteado es prácticamente incongruente. El 60% del tema es copiado. | Realiza un 50% ó 60% de los experimentos, mencionando el procedimiento de manera completa incluyendo el material y equipo utilizado. Las respuestas son congruentes con los experimentos realizados. |  |
| **2.- Desarrollo de la Práctica - Dedicación** | Realiza todos los experimentos mencionando el procedimiento de manera completa, incluyendo el material y equipo utilizado. Las respuestas son congruentes con los experimentos realizados. | Realiza un 80% de los experimentos, mencionando el procedimiento de manera completa incluyendo el material y equipo utilizado. Las respuestas son congruentes con los experimentos realizados. | Realiza un 50% ó 60% de los experimentos, mencionando el procedimiento de manera completa incluyendo el material y equipo utilizado. Las respuestas son congruentes con los experimentos realizados. | Realiza un 40% de los experimentos, mencionando el procedimiento de manera completa incluyendo el material y equipo utilizado. Las respuestas son congruentes con los experimentos realizados. |  |
| **3.- Interpretación,**  **Análisis de los**  **Resultados** | Recopila y ordena los datos en relación al procedimiento. Se presentan los datos en tablas, gráficas, dibujos, etc. Claramente identificados. Los datos se interpretan y analizan comparativamente con la información bibliográfica consultada. | Presenta datos ordenados en relación al procedimiento. Se presentan en tablas, gráficas,  dibujos, etc. claramente  identificados, se interpretan y  analizan parcialmente en un  80%. | Tiene datos parcialmente ordenados, presenta algunas tablas o gráficas, los resultados  se interpretan y analizan en 50%  ó 60% | Tiene datos parcialmente ordenados, presenta algunas tablas o gráficas, los resultados se interpretan y analizan en 50%. |  |
| **4.- Conclusión** | Deduce el comportamiento de la(s) variable(s) estudiada(s) a partir del problema planteado. Rechaza o acepta la hipótesis e incluye propuestas de mejora o genera nuevos problemas. | Deduce el comportamiento de la(s) variable(s) estudiada(s) a partir del problema planteado.  Incluye el rechazo o la aceptación de la hipótesis, pero no las propuestas de mejoras. | Deduce el comportamiento de la(s) variable(s) estudiada(s) a partir del problema planteado.  No incluye el rechazo o aceptación de la hipótesis ni propone mejoras. | Deduce el comportamiento de la(s) variable(s) estudiada(s) a partir del problema planteado.  No incluye el rechazo o  aceptación de la hipótesis ni  Propone mejoras. |  |
| **5.- Actitud, Motivación, Disciplina** | Participa propositiva e integralmente en toda la práctica. | Participa ocasionalmente o lo hace constantemente pero sin coordinarse con su compañero. | Es un observador pasivo. | No participa en la realización de la práctica. |  |

**BIBLIOGRAFÍA LIBROS DIGITALES DE LA UNIVERSIDAD:**

**BIBLIOWEB:**

* <https://www.ibm.com/support/knowledgecenter/es/ssw_ibm_i_71/rzaha/rzahajsseexmpls15.htm>
* <https://programacion.net/articulo/rmi_mano_a_mano_con_ssl_construyendo_aplicaciones_distribuidas_seguras_79>
* <http://chuwiki.chuidiang.org/index.php?title=Socket_SSL_con_Java>