UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
COLEGIADO DE INFORMÁTICA

Disciplina: Redes de Computadores — Data Entrega: 05/junho/2008
Professor: Luiz Antonio — Realização: em duplas.

Trabalho do 4º Bimestre - Comunicação segura com criptografia

O trabalho consiste em implementar um meio seguro para troca de mensagens entre dois usuários. O processo se inicia com a autenticação do usuário emissor A (aquele que inicia a comunicação) perante o usuário receptor B (aquele que responde a solicitação). Neste processo A solicita uma conexão com B. B gera um valor aleatório R o envia a A. A criptografa R com sua chave pública e o envia a B. B obtém a chave pública de A e verifica R. Se a verificação obtiver sucesso, B gera uma chave simétrica, criprografada com a chave pública de A e envia a A. Com isso, apenas A, que possui a chave privada correspondente, poderá obter a chave simétrica enviada por B. A e B agora podem se comunicar de forma segura.

Em resumo, o algoritmo segue a seguinte ordem:

1. A estabelece uma conexão com B, informando seu nome;
2. B gera um valor inteiro R aleatório e envia a A;
3. A criptografa R com sua chave privada e envia a B;
4. B obtém a chave pública de A e verifica a autenticidade da mensagem recebida;
5. B escolhe uma chave aleatória qualquer para um algoritmo de chave simétrica;
6. B usa a chave pública de A para encriptar a chave escolhida, e manda para A o resultado;
7. A usa a sua chave privada para decriptar a chave enviada por B;.
8. A e B usam um algoritmo de chave simétrica para trocar as mensagens.

As chaves públicas dos participantes devem ser registradas em uma autoridade certificadora CA (*certification authority*), assim que o processo entra em execução. As solicitações recebidas por uma CA podem ser de dois tipos:

- REGISTRO – mensagem com código 0 e que contém o nome e a chave pública do membro; a CA armazena estas informações para consultas futuras;
- CONSULTA – mensagem com código 1 que solicita a chave pública de um membro a partir de seu nome. A CA devolve a chave pública solicitada.

As solicitações de REGISTRO e de CONSULTA devem utilizar a classe `Membro`, descrita a seguir. Toda solicitação recebida pela CA deve conter um objeto serializado `Membro`. No caso de uma CONSULTA, a CA devolve um objeto `Membro` contendo a chave pública solicitada.

```
import java.io.Serializable;
import java.security.PublicKey;

public class Membro implements Serializable {
      private int codigo; //0 ou 1
      private String nome; //nome de registro
      private PublicKey pubKey; //chave publica do membro

      //construtores, gets e sets
}
```

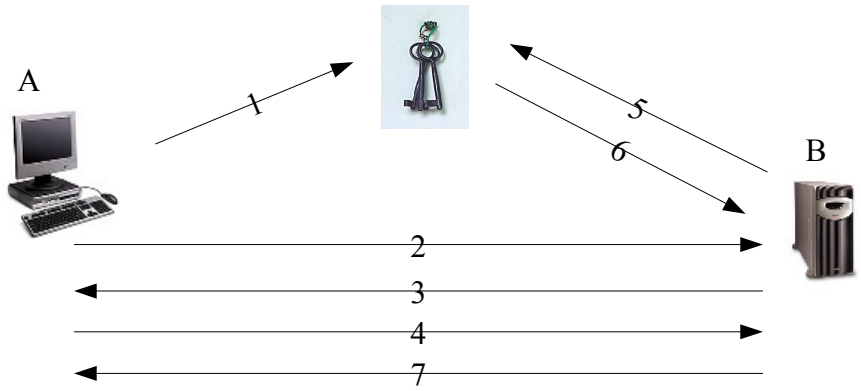
Dada a descrição observa-se que o trabalho deverá conter três processos: o Cliente, o Servidor e a CA.

O Cliente é o usuário que inicia a conexão.
O Servidor é o usuário que recebe a conexão.
A CA é a autoridade certificadora que mantém as chaves dos usuários.

Um exemplo de comunicação é mostrado na Figura 2. Os passos são os seguintes:

1. A registra sua chave pública na CA;
2. A solicita conexão com B;
3. B envia R aleatório para A;
4. A criptografa R com sua chave privada e envia a B;
5. B solicita chave pública de A para a CA;
6. CA responde a B com a chave pública de A;
7. B verifica R e envia a chave simétrica criptografada com a chave pública de A.

A partir desse ponto A e B trocam mensagens utilizando a chave simétrica.



Observações:

1. O algoritmo de criptografa de chaves públicas deverá ser o RSA e o algoritmo de chaves simétricas será o DES. A API `java.security` possui implementações desses algoritmos. Para maiores informações consulte no Javadoc as classes `KeyGenerator`, `KeyPairGenerator` e `Cipher`.
2. Após a troca da chave simétrica, A e B devem ser capazes de enviar mensagens aleatóriamente e/ou simultaneamente um para o outro, como ocorre em qualquer programa de bate-papo.
3. O trabalho deverá ser enviado por e-mail no formato aluno1_aluno2.zip.