Criptografia em Sistemas Distribuídos

Sistemas Distribuídos e Mobile

Prof. Me. Gustavo Torres Custódio
gustavo.custodio@ulife.com.br

Criptografia em Sistemas Distribudos

Criptografia

Criptografia

- Um dos recursos utilizados para manter a segurança de sistemas distribuídos é a criptografia.

Criptografia

- Um dos recursos utilizados para manter a segurança de sistemas distribuídos é a criptografia.
- Mesmo que mensagens enviadas entre diferentes sistemas sejam interceptadas, a segurança é mantida.

Criptografia

- Na criptografia, tentamos converter um **texto puro** para um **texto cifrado**.

Criptografia

- Na criptografia, tentamos converter um **texto puro** para um **texto cifrado**.
- O texto puro só pode ser lido pelas pessoas autorizadas.

Criptografia

- Na criptografia, tentamos converter um **texto puro** para um **texto cifrado**.
- O texto puro só pode ser lido pelas pessoas autorizadas.
- As pessoas não autorizadas só possuem acesso ao texto cifrado.

Criptografia

- Na criptografia, tentamos converter um **texto puro** para um **texto cifrado**.
- O texto puro só pode ser lido pelas pessoas autorizadas.
- As pessoas não autorizadas só possuem acesso ao texto cifrado.
- Esse texto é incompreensível.

Criptografia

- O texto cifrado é obtido usando um algoritmo de criptografia (conhecido).

Criptografia

- O texto cifrado é obtido usando um algoritmo de criptografia (conhecido).
- O texto cifrado é traduzido utilizando uma chave de criptografia (secreta).

Criptografia

- O texto cifrado é obtido usando um algoritmo de criptografia (conhecido).
- O texto cifrado é traduzido utilizando uma chave de criptografia (secreta).
- O **princípio de Kerckhoffs** estabele que o segredo deve estar contido exclusivamente na chave e os algoritmos de criptografia devem ser públicos.

Criptografia de chave privada

- Considere o seguinte texto puro e sua cifra:

Criptografia de chave privada

- Considere o seguinte texto puro e sua cifra:
  - puro: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - cifrado: QWERTYUIOPASDFGHJKLZXCVBNM

Criptografia de chave privada

- Considere o seguinte texto puro e sua cifra:
  - puro: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - cifrado: QWERTYUIOPASDFGHJKLZXCVBNM
- Nesse caso, a chave privada é a cadeia de 26 letras que corresponde ao alfabeto.

Criptografia de chave privada

- Considere o seguinte texto puro e sua cifra:
  - puro: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - cifrado: QWERTYUIOPASDFGHJKLZXCVBNM
- Nesse caso, a chave privada é a cadeia de 26 letras que corresponde ao alfabeto.
- Se a chave for descoberta é fácil decodificar qualquer mensagem.

Criptografia de chave privada

- Considere o seguinte texto puro e sua cifra:
  - puro: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - cifrado: QWERTYUIOPASDFGHJKLZXCVBNM
- Nesse caso, a chave privada é a cadeia de 26 letras que corresponde ao alfabeto.
- Se a chave for descoberta é fácil decodificar qualquer mensagem.
- Esse tipo de criptografia é chamada **criptografia de chave simétrica**.

Criptografia de Chave Pública

- Na criptografia de chave simétrica (privada), tanto o receptor quanto o remetente precisam ter em mãos a chave privada.

Criptografia de Chave Pública

- Na criptografia de chave simétrica (privada), tanto o receptor quanto o remetente precisam ter em mãos a chave privada.
- A **criptografia de chave pública ou assimétrica** tenta contornar esse problema.

Criptografia de Chave Pública

- Na criptografia de chave simétrica (privada), tanto o receptor quanto o remetente precisam ter em mãos a chave privada.
- A **criptografia de chave pública ou assimétrica** tenta contornar esse problema.
- Nela, as chaves para criptografar e para traduzir a mensagem são diferentes.

Criptografia de Chave Pública

- Na criptografia de chave simétrica (privada), tanto o receptor quanto o remetente precisam ter em mãos a chave privada.
- A **criptografia de chave pública ou assimétrica** tenta contornar esse problema.
- Nela, as chaves para criptografar e para traduzir a mensagem são diferentes.
- Sob essas circunstância a chave para criptografar a mensagem pode ser pública, mas a chave para traduzi-la deve ser privada.

Criptografia de Chave Pública

- A encriptação faz uso de uma operação fácil, mas a decriptação sem a chave exige uma operação complexa (que provavelmente não pode ser terminada em uma vida).

Criptografia de Chave Pública

- A encriptação faz uso de uma operação fácil, mas a decriptação sem a chave exige uma operação complexa (que provavelmente não pode ser terminada em uma vida).
  - Operação fácil: multiplicação.
  - Operação difícil: fatoração de primos.

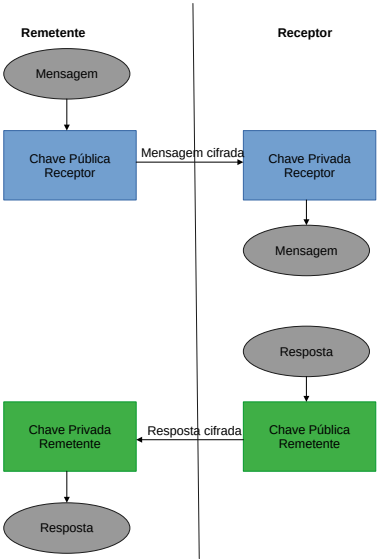
Criptografia de Chave Pública

- Na criptografia de chave assimétrica, um correspondente encripta a mensagem usando a chave pública do receptor.

Criptografia de Chave Pública

- Na criptografia de chave assimétrica, um correspondente encripta a mensagem usando a chave pública do receptor.
  - Apenas o receptor poderá descriptografar essa mensagem já que ele é o único que possui a chave privada.

Criptografia de Chave Pública



Exemplo em Java

```
public static KeyPair gerarChavesPublicoPrivada() throws NoSuchAlgorithmException{
   KeyPairGenerator geradorChave = KeyPairGenerator.getInstance("RSA");
   geradorChave.initialize(2048);
   KeyPair par = geradorChave.generateKeyPair();
   return par;
}

public static String
      cifrar(String mensagem, PublicKey publicKey) throws Exception {

   byte[] messageToBytes = mensagem.getBytes();
   Cipher cifrador = Cipher.getInstance("RSA/ECB/PKCS1Padding");

   // Cifrar mensagem
   cifrador.init(Cipher.ENCRYPT_MODE, publicKey);
   byte[] bytesCripto = cifrador.doFinal(messageToBytes);

   return Base64.getEncoder().encodeToString(bytesCripto);
}
```

Exemplo em Java

```
public static String
     decifrar(String mensagem, PrivateKey privateKey) throws Exception {

   // Converte a mensagem cifrada para bytes
   byte[] bytesCifrados = Base64.getDecoder().decode(mensagem);
   Cipher cifrador = Cipher.getInstance("RSA/ECB/PKCS1Padding");

   // Decriptografa os bytes
   cifrador.init(Cipher.DECRYPT_MODE, privateKey);
   byte[] mensagemDecifrada = cifrador.doFinal(bytesCifrados);

   // Cria os bytes da mensagem decifrada para uma string
   return new String(mensagemDecifrada, "UTF8");
}

// Converte os bytes de uma chave pública enviado pelo socket de volta para a chave
public static PublicKey bytesParaChave(byte[] bytesChave) throws Exception {
   X509EncodedKeySpec keySpec = new X509EncodedKeySpec(bytesChave);
   KeyFactory keyFactory = KeyFactory.getInstance("RSA");
   return keyFactory.generatePublic(keySpec);
}
```

Exemplo em Java

```
public static void main(String[] args) {
   try {
      Scanner sc = new Scanner(System.in);

      KeyPairGenerator geradorChave = KeyPairGenerator.getInstance("RSA");
      geradorChave.initialize(2048); // Iniciamos uma chave de 2048 bits
      KeyPair par = geradorChave.generateKeyPair(); // Gera um par chave pública e
          privada

      System.out.println("Digite a mensagem secreta:");
      String segredo = sc.nextLine();

      PrivateKey privateKey = par.getPrivate();
      PublicKey publicKey = par.getPublic();

      try {
         String mensagemCifrada = Criptografia.cifrar(segredo, publicKey);
         System.out.println("Essa éa mensagem cifrada:\n" + mensagemCifrada);

         String mensagemDecifrada = Criptografia.decifrar(mensagemCifrada,
             privateKey);
         System.out.println("A mensagem: " + mensagemDecifrada + " foi decifrada
             com sucesso.");
      }catch (Exception e) {
         System.err.println(e.getMessage());

      }finally {
         sc.close();
      }
   }catch (Exception e) {
      System.err.println(e.getMessage());
   }
}
```

Exemplo em Java

Digite a mensagem secreta:
Hello World
Essa é a mensagem cifrada:
b/NV3Kd22fiRJ5klEHhgLgc9zJeDs2L9DGvb9x1bW5RwwirMAjXds...
A mensagem: Hello World foi decifrada com sucesso.

O que Aconteceu?

- Utilizamos o algoritmo de chave assimétrica RSA para cifrar uma mensagem.

O que Aconteceu?

- Utilizamos o algoritmo de chave assimétrica RSA para cifrar uma mensagem.
  - Primeiro geramos uma chave pública e uma privada aleatórias.
  - Utilizamos essas chaves, respectivamente, para cifrar e decifrar a mensagem.

O que Aconteceu?

- Utilizamos o algoritmo de chave assimétrica RSA para cifrar uma mensagem.
  - Primeiro geramos uma chave pública e uma privada aleatórias.
  - Utilizamos essas chaves, respectivamente, para cifrar e decifrar a mensagem.
- Utilizamos o algoritmo RSA para gerar as chaves.

O que Aconteceu?

- O RSA trabalha gerando números primos muito grandes.

O que Aconteceu?

- O RSA trabalha gerando números primos muito grandes.
  - Em nosso caso, números de 2048 bits (números com 617 casas).

O que Aconteceu?

- O RSA trabalha gerando números primos muito grandes.
  - Em nosso caso, números de 2048 bits (números com 617 casas).
- As chaves pública e privada são calculadas com base em operações realizadas nos números primos gerados.

O que Aconteceu?

- O RSA trabalha gerando números primos muito grandes.
  - Em nosso caso, números de 2048 bits (números com 617 casas).
- As chaves pública e privada são calculadas com base em operações realizadas nos números primos gerados.
  - Operações para encontrar a chave privada são muito custosas.

O que Aconteceu?

- As mensagens cifradas possuem tamanho constante.

O que Aconteceu?

- As mensagens cifradas possuem tamanho constante.
  - Não importa o tamanho da mensagem enviada, a mensagem cifrada sempre terá o mesmo tamanho.
  - Impossível decifrá-la sem a chave privada.

Chave Pública

```
Sun RSA public key, 2048 bits
  params: null
  modulus: 18783725138160042491088895584092771759374611948499813677552633632534165793849034003110284705
401127212234888223158933107138988968148555295172910997522685950190739676989458044774946837193036596691 8
825278931055879284511113645796330721576305690034568489230161569945947736016545002955502639371890848710 6
102667601226120317687521796146528313650027003745578135857630701422229729767392494902563126173786659063 6
632209483752589456354021530340160476274352106816929877726767886342927749709059399389922425521234682174 5
325792621941371815239792295404354128862609062173162254942128795122425716548205119908590648280588516312 3
5726538413
  public exponent: 65537
```

Chave Privada

```
SunRsaSign RSA private CRT key, 2048 bits
  params: null
  modulus: 18783725138160042491088895584092771759374611948499813677552633632534165793849034003110284705
401127212234888223158933107138988968148555295172910997522685950190739676989458044774946837193036596691 8
825278931055879284511113645796330721576305690034568489230161569945947736016545002955502639371890848710 6
102667601226120317687521796146528313650027003745578135857630701422229729767392494902563126173786659063 6
632209483752589456354021530340160476274352106816929877726767886342927749709059399389922425521234682174 5
325792621941378152397922954043541288626090621731622549421287951224257165482051199085906482805885163123
5726538413
  private exponent: 10962355907111178497629859137592204926263944913655200475136688148330969883930266148
144745249434400622756000499860885812317500801894287851591825988880291930098958010978421216969836376839 9
417695389035648085127870834398600404724324540928796558164734052154892957066885444549808707646704379979 1
885680071321769846203576618440920008743464772903442547959664291764021152066060475838284319707816221817 8
079315703122966583570927490375502156678526474860229835639917633482179303867456474644374058749958307876 6
748238939617847828208947922611250338229474257224189427190125797982603375505124037500756350176874966496 7
6606815498446682625
```

Chave Privada

```
SunRsaSign RSA private CRT key, 2048 bits
  params: null
  modulus: 18783725138160042491088895584092771759374611948499813677552633632534165793849034003110284705
401127212234888223158933107138988968148555295172910997522685950190739676989458044774946837193036596691 8
825278931055879284511113645796330721576305690034568489230161569945947736016545002955502639371890848710 6
102667601226120317687521796146528313650027003745578135857630701422229729767392494902563126173786659063 6
632209483752589456354021530340160476274352106816929877726767886342927749709059399389922425521234682174 5
325792621941378152397922954043541288626090621731622549421287951224257165482051199085906482805885163123
5726538413
  private exponent: 10962355907111178497629859137592204926263944913655200475136688148330969883930266148
144745249434400622756000499860885812317500801894287851591825988880291930098958010978421216969836376839 9
417695389035648085127870834398600404724324540928796558164734052154892957066885744549808707646704379979 1
885680071321769846203576618440920008743464772903442547959664291764021152066060475838284319707816221817 8
079315703122966583570927490375502156678526474860229835639917633482179303867456474644374058749958307876 6
748238939617847828208947922611250338229474257224189427190125797982603375505124037500756350176874966496 7
6606815498446682625
```

- Não mostre isso para os outros.

Chaves e Sistemas Distribuídos

- Vimos que o receptor precisa da chave pública do recipiente para cifrar as mensagens (e vice-versa).

Chaves e Sistemas Distribuídos

- Vimos que o receptor precisa da chave pública do recipiente para cifrar as mensagens (e vice-versa).
  - A chave pública do recipiente é gerada no recipiente.
  - Como o receptor pode ter acesso a essa chave?
  - A chave é pública...

Chaves e Sistemas Distribuídos

- Podemos simplesmente enviar a chave pública para o outro lado.

Chaves e Sistemas Distribuídos

- Podemos simplesmente enviar a chave pública para o outro lado.
  - Ela só vai servir para cifrar mensagens, não vai ajudar a decifrar nenhuma.
  - Isso pode ser feito utilizando sockets.

Criptografia em Sistemas Distribudos

Exercícios

Exercícios

- Modifique o código de sockets com TCP visto na anteriormente.
  - Faça com que as mensagens sejam criptogradas antes de serem enviadas.
  - Quando um cliente se conectar com o servidor, cada um deve gerar uma chave pública e privada.
    - As chaves públicas devem ser compartilhadas.

Obrigado

gustavo.custodio@ulife.com.br