Computação Distribuída [Engenharia Informática] - 2023

Iniciada	quinta, 23 de novembro de 2023 às 21:31
Estado	Terminada
Terminada	quinta, 23 de novembro de 2023 às 22:50
Tempo gasto	1 hora 18 minutos

Pergunta **1**

Respondida

Nota: 10,00

Considere 2 agentes, A e B, cada um com o seu par de chaves pública e privada, respetivamente:

A: KAPub, KAPriv B: KBPub, KBPriv

Considere que A quer enviar a B a mensagem M com **privacidade**. Represente a mensagem que A deve enviar a B e descreva porque é que a mensagem garante privacidade.

Se A quer enviar uma mensagem para B com privacidade deve enviar a mensagem cifrada utilizando a chave publica de B [A -> B {M}KBpub]. Desta forma quando B receber a mensagem de A, irá utilizar a sua chave privada de forma a decifrar a mensagem [{{M}KBpub}KBpriv]. Deste modo é garantida a privacidade no envio da mensagem visto a mensagem sendo cifrada com a chave publica de B, apenas pode ser decifrada com a respetiva chave privada que é só conhecida por B.

Pergunta **2**

Respondida

Nota: 10,00

Como é feito o controlo de acessos num sistema **MAC (Mandatory Access Control)** ? Ilustre com o seguinte exemplo: um cliente está a tentar ler um ficheiro remoto, tendo chamado o serviço: **read (ficheiro, utilizador)**

O cliente requisita determinado recurso. Este será associado a um agente no sistema que por sua vez irá representar o cliente. No caso de um cliente (utilizador) requisitar 1 ficheiro para leitura, o sistema operativo terá que validar se este utilizador (associado a um agente que o representa) tem efetivamente permissões de leitura para esse mesmo ficheiro. Caso tenha é garantida caso não tenha permissões não pode ler o ficheiro e portanto o seu pedido é rejeitado.

Pergunta **3**

Respondida

Nota: 10,00

Descreva 3 ameaças ou ataques à segurança em sistemas distribuídos

3 ameaças ou ataques a segurança de sistemas distribuídos pode ser:

- DOS (Denial of service) onde o servidor é sobrecarregado de mensagem até os seus recursos serem esgotados, resultando num crash ou outra falha no sistema.

- Man in the middle, onde um atacante pode fazer passar-se por determinado serviço requisitado por alguém de modo a tentar escalar acessos numa rede, entrar em sistemas protegidos, etc.

- Perturbação ou interferência no fluxo de mensagens, onde um atacante pode tentar requisitar recursos (ou transações, etc) por via de re-envio de pedidos anteriores feito em relação ao recurso desejado.

Pergunta **4**

Respondida

Nota: 10,00

Descreva resumidamente a função do seguinte componente do WSDL: PortType

O componente do WSDL:PortType define a interface do Webservice e descreve as operações realizáveis pela mesma, assim como os métodos que podem ser invocados.

Pergunta **5**

Respondida

Nota: 10,00

Considere o programa em SUN RPC abaixo. Diga onde está definida e descreva o funcionamento da função que o servidor exporta para o cliente.

```
/* date.x */
program DATEPROG {
    version DATEVERS {
        long BINDATE(void) = 1;
    } = 1;
} = 0x3012225;

/* dateproc.c - remote procedures; called by server stub */
#include <stdio.h>
#include <stdlib.h>
#include <rpc/rpc.h>
#include "date.h"
long * bindate_1_svc(void* arg1, struct svc_req *arg2) {
    static long timeval; /* must be static */
    timeval = time((long *) 0);
    return (&timeval);
}
```

A função está definida em dateproc.c:

```
long * bindate_1_svc(void* arg1, struct svc_req *arg2) {
    static long timeval; /* must be static */
    timeval = time((long *) 0);
    return (&timeval);
}
```

bindate_1_svc é exportada para o cliente e, quando chamada, retorna as horas atuais registadas pela função

Pergunta **6**

Respondida

Nota: 10,00

Como é feito o controlo de acessos num sistema **DAC (Discretionary Access Control)** ? Ilustre com o seguinte exemplo: um cliente está a tentar ler um ficheiro remoto, tendo chamado o serviço: **read (ficheiro, utilizador)**

O controlo de acesso num sistema DAC é feito através de um sistema de confiança entre cliente / servidor, onde de forma a que a primeira possa aceder a recursos, e segunda terá que ter conhecimento do cliente de forma a lhe dar as devidas permissões de acesso.

Pergunta **7**

Respondida

Nota: 10,00

Considere 2 agentes, A e B, uma autoridade de certificação S, e as 4 mensagens do protocolo Needham - Schroeder de autenticação com chave secreta:

1. A → S: A, B, Na

2. S → A: {Na, B, Kab, {Ts, A, Kab}Kb}Ka

3. A → B: {A, Ta}Kab, {Kab, A, Ts}Kb

4: B → A : {Ta + 1}Kab

Na mensagem 2, como é que A obtém a chave Kab ? Como é que tem a certeza que a chave é válida ?

1. A pede autoridade de certificação (S) a chave publica de B de forma a poder comunicar com ele.

2. S envia para A Na, id do B e Kab e cifra um time stamp com id do requerente A e Kab cifradas com a chave publica de B.

Portanto A obtém Kab através de S e confia na validade da mensagem visto que a mesma provem de S.

Visto que a mensagem 2 enviada por S -> A contém uma parte cifrada com a chave publica de B, apenas B a pode decifrar usando a sua chave privada

Pergunta **8**

Respondida

Nota: 10,00

É boa ideia a segurança de uma comunicação cifrada estar baseada no secretismo do algoritmo de cifra ? Porquê ? Qual é a solução correta ?

Não é uma boa ideia a segurança de uma comunicação cifrada estar baseada no secretismo do algoritmo de cifra, visto que se houver, por algum motivo, um leak da mesma ou se algum partido menos desejado (hacker) ficar em posse dela, então este teria a possibilidade de decifrar todas as mensagens cifradas com essa chave, visto que é usada a mesma chave tanto para cifrar como a decifrar. Neste caso temos uma cifra síncrona.

Uma solução mais robusta e que portanto não teria este problema, seria utilizando uma cifra assíncrona por via de chaves publicas e chaves privadas de modo a mesmo que uma das chaves seja comprometida, o atacante não poderá decifrar ou intervier de algum modo com as mensagens.

Pergunta **9**

Respondida

Nota: 10,00

Quais são os problemas que se pretendem resolver com os Web Services e porque razão o Sun RPC e o Java RMI não são adequados para resolver esses problemas ?

O Web Services resolvem o problema de comunicação entre plataformas heterogéneas, isto é, independente de implementação ou tecnologia usada, pois define um padrão de comunicação que pode ser utilizado por qualquer sistema que o implemente.

Isto é vantajoso visto que no caso do Java RMI ou Sun RPC, este necessitam que as aplicações distribuídas sejam homogéneas na implementação e por isso dependem das respetivas tecnologias usadas. Logo isto cria imensos impedimentos de integração de determinada aplicação com outras aplicações visto que seria uma condição obrigatória para tal será a coesão entre plataformas. Os Web Services não tem este problema pois, por exemplo no caso do SOAP existe um padrão de comunicação geral, conhecido, que usa XML para comunicação entre serviços ou aplicações.

Pergunta **10**

Respondida

Nota: 10,00

```
public class Server implements Hello {
  public Server() {}
  public String sayHello() {
  return "Hello, world!";
  }
   public static void main(String args[]) {
    try {
      Server obj = new Server();
      Hello stub = (Hello)
                UnicastRemoteObject.exportObject(obj, 0);
      // Bind the remote object's stub in the registry
      Registry registry = LocateRegistry.getRegistry();
      registry.bind("Hello", stub);
  }
```

No exemplo de um servidor Java RMI acima, diga para que serve a linha **registry.bind("Hello", stub);**

A linha **registry.bind("Hello", stub)**, regista o objecto remoto (representado pelo stub) no registro RMI com o nome "Hello". Desta forma os clientes iram conseguir invocar métodos nesse objecto.

Previous Activity

Ir para...

Next Activity