|  |  |
| --- | --- |
| C:\Users\PM\Desktop\RC-Projecto-Rede.bmp | **Instituto Superior de Engenharia de Lisboa**  Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores  **Segurança Informática, 5º Semestre** |

Segurança Informática

*2ª Série*

Docente: Pedro Félix

|  |  |  |  |
| --- | --- | --- | --- |
| Elaborado por: |  |  |  |
|  |  |  |  |
| André Cunha  Pedro Marinho  Cátia Ormonde | G06  G06  G06 | LI51N  LI51N  LI51N | nº31612  nº36122  nº36923 |

Lisboa, 18 de Novembro de 2014

Índice

[1. Respostas ao enunciado 3](#_Toc403135465)

[3.1. O salt deve ser armazenado cifrado? 3](#_Toc403135466)

[Resposta: Não, o *salt* tem de estar armazenado em claro para poder ser adicionado à password fornecida e testado contra o existente na base de dados. O campo de password é constituído pelo *hash* (ou múltiplas aplicações de *hash* sobre ele mesmo) da password e o *salt*. 3](#_Toc403135467)

[3.2. Quais as consequências de usar um salt constante? 3](#_Toc403135468)

[4.1. *Qual a diferença entre o Resource Owner e o cliente?* 4](#_Toc403135469)

[4.2. *Qual a razão pela qual o cliente não se autentica quando realiza o pedido de authorization request ao Authorization Endpoint (fluxo Authorization Code Grant)?* 4](#_Toc403135470)

[4.3. *Quais os problemas dos bearer tokens?* 4](#_Toc403135471)

[5.1. Realize uma aplicação Web com a seguinte funcionalidade: *•* Autenticação baseada no fornecedor de identidade social da Google, usando o protocolo OpenID Connect. *•* Apresentação de *issues* do GitHub. *•* Criação de *tasks* Google a partir de *issues* do GitHub. 4](#_Toc403135472)

# Respostas ao enunciado

1. *Qual a utilidade de garantir a integridade de um key store onde apenas são guardados certificados auto-*

*assinados?*

R:

1. *No contexto do protocolo SSL/TLS?*

*2.1. Considere que o servidor malicioso S1 realiza uma instância do protocolo handshake com o cliente*

*C. Como é que este protocolo impede que S1 se possa autenticar como C perante um outro servidor*

*S2, nomeadamente através do reenvio das mensagens que C enviou para S1.*

R:

*2.2. Em que contexto e com que objectivo são usados esquemas de assinatura digital?*

*2.3. O que é e qual a importância da propriedade Perfect Forward Secrecy?*

R:

1. No contexto do armazenamento de informação de verificação para esquemas  
   de autenticação baseados em passwords:

## O salt deve ser armazenado cifrado?

## Resposta: Não, o *salt* tem de estar armazenado em claro para poder ser adicionado à password fornecida e testado contra o existente na base de dados. O campo de password é constituído pelo *hash* (ou múltiplas aplicações de *hash* sobre ele mesmo) da password e o *salt*.

Tabela de uma BD com passwords

|  |  |
| --- | --- |
| UserID | Password |
|  | Sha1(xpto+password)+ salt |

## Quais as consequências de usar um salt constante?

Resposta: Quando a base de dados de passwords é conhecida pelo atacante,  
este pode usar um dicionário para obter as passwords para cada um dos *hash*  
guardados. Se o *salt* for constante, e de alguma forma o atacante souber qual  
é, ele pode alterar as passwords do dicionário para já conterem o *salt*.

1. *No contexto da norma OAuth 2.0*

## *Qual a diferença entre o Resource Owner e o cliente?*

Resposta:

## *Qual a razão pela qual o cliente não se autentica quando realiza o pedido de authorization request ao Authorization Endpoint (fluxo Authorization Code Grant)?*

Resposta:

## *Quais os problemas dos bearer tokens?*

Resposta:

1. *Realize um programa que, dado o hostname para um site com suporte HTTPS, apresenta:  
   • Informação sobre a validade do certificado do servidor (e.g. validade temporal, cadeia de certificados, adequação ao hostname).  
   • A cadeia de certificados do servidor.  
   • A menor data de expiração dos certificados do servidor.  
   • As versões dos protocolos SSL e TLS suportadas (de entre as disponíveis na plataforma Java).*

Resposta: Ver código em anexo

## Realize uma aplicação Web com a seguinte funcionalidade: *•* Autenticação baseada no fornecedor de identidade social da Google, usando o protocolo OpenID Connect. *•* Apresentação de *issues* do GitHub. *•* Criação de *tasks* Google a partir de *issues* do GitHub.

Resposta: Ver código em anexo