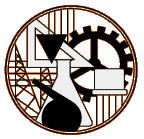
**Instituto Superior de Engenharia de Lisboa**

**Ano Lectivo 2011/2012 – Semestre Inverno**

**Segurança Informática**

**2ª Série**

**Professor:** José Simão

**Data:** 29 de Novembro de 2011

**Autores**:

Cláudia Crisóstomo nº 32142

Nuno Cancelo nº 31401

Telmo Rafael Pinto n.º30505

# Indice

# Exercício 1

**Considere o artigo de Halderman et al.**

* 1. **Qual o problema tratado neste artigo?**

O artigo “A Convenient Method for Securely Managing Passwords” trata o problema dos utilizadores serem obrigados a realizar registo em aplicações web que irá originar um aumento de passwords a memorizar pelo cliente relativas a todas as contas como por exemplo: as contas de email, sites comerciais, entre outros.

* 1. **O que é um “phishing attack”? De que forma a proposta apresentada neste artigo oferece protecção contra este tipo de ataque? Descreva sucintamente outros métodos propostos para evitar este tipo de ataque.**

Um “pishing attack” define-se por uma entidade passar-se por outra entidade tentando obter credenciais do cliente. Neste artigo é proposto uma protecção para este ataque que tem como base um plug-in para o browser mozilla firefox que através de dois procedimentos gera a password de acesso a um determinado site. Estes dois procedimentos baseam-se em gerar hash n vezes cujos paramêtros são recursivos e o segundo procedimento usa o resultado do primero. Mas a grande utilidade desta ferramenta encontra-se no facto de que se um atacante usar um site malicioso a chave usada para autenticação é uma chave criada a partir dos dois procedimentos e não uma chave do cliente em concreto. Também foram propostos outros métodos para evitar este ataque sendo referidos: Microsoft Passport e Lucent Personal Web Assistant (LPWA). O Microsoft Passport tem como finalidade centralizar a autenticação de serviços de confiança, assim, os utilizadores usam apenas uma password e quando esta é valida este pode aceder aos seus sites, mas este sistema tem duas limitações: requer mudanças técnicas por parte de cada site de modo a usar este tipo de autenticação e os utilizadores têm-se mostrado demasiados cuidadosos em relaçao em colocar demasiada confiança em dados privados sobre o controlo de um sistema centralizado.

* 1. **A informação inserida pelo utilizador para a derivação das “passwords” inclui um “username”. Qual a finalidade desta componente?**

A finalidade consiste em proteger a password original de ponteciais atacantes tendo em conta que os ataques de dicitionario e brute force terão de obter duas incógnitas o que implica maior tempo de computação. Além disso, para cada registo relativo a diversos sites e o username e a password poderão ser diferentes pelo que, se o atacante obtiver a “password” relativo a um “username” (ataque de dicionário ou brute force) não haverá possibilidade deste tirar vantagem desta noutros sites, pois este esquema gera uma hash por cada novo registo.

* 1. **Justique a divisão do processo de derivação de “passwords” específicas em duas fases.**

O processo de derivação de “passwords” é específica em duas fases, uma primeira na geração de uma variável V e posteriormente a geração da chave.  
O primeiro passo consiste em realizar uma função de hash com os seguintes parâmetros: username e password onde a esta é gerada k1 vezes cujo resultado será guardado em cache. Seguidamente, será novamente gerada um hash mas com três parâmetros sendo estes: o nome do site, a password do utilizador e a variável V, e consequentemente esta hash será gerada k2 vezes sendo esta denominada password do site. As variáveis k1 e k2 terão um valor coerente com o tempo de computação da geração das n hashs para que este sistema não se torne inconsistente para com os utilizadores.  
Este processo permite que um atacante ao usar o ataque de dicionário tenha de tentar validar as credenciais e voltar a iterar hashs numa lista de possíveis passwords.

* 1. **Quais os tipos de ataque considerados? Justique esta classicação .**

Este artigo considerou 5 tipos de ataque:

* **Sem informação:**
  + O atacante não tem qualquer informação acerca do utilizador, por isso, este será confrontado com a demora do mecanismo na validação de todas as possíveis credenciais o que poderá implicar a desactivação da conta se demasiadas tentativas falhas forem efectuadas.
  + O atacante poderá efectuar este ataque de forma faseada, mas terá um alto risco de ser detectado.
  + Caso o atacante consiga obter chave apenas conseguirá validar as credenciais obtidas naquele site em concreto e não outro.
* ***Password* do site roubada:**
  + Esta *password* poderá ser adquirida se o utilizador se autenticar num site malicioso.
  + O atacante poderá assim efectuar um ataque *offline* para tentar descobrir a password do utilizador. Para tal o atacante irá necessitar de efectuar de dicionário/*brute* *force* em que cada tentativa será submetida à expressão: p(k1+k2) o que irá implicar tempo de computação, cerca de 1 minuto por tentativa, mas a principal preocupação é o facto de este atacante não poder ser detectado.
* ***Cache* roubada:**
  + Neste ataque o atacante comprometeu a máquina do utilizador conseguindo obter os dados da váriável intermédia V.
  + Este atacante poderá assim efectuar um ataque no modo *offline* cuja intenção será encontrar a *password* do utilizador através da utilização do primeiro passo do algoritmo, assim, para cada possível *pasword* o atacante compara com o valor da variável intermédia V.
  + Cada tentativa de encontrar a *password* irá obrigar o atacante a efectuar a *hash* *k1* vezes o que implica maior demora na geração da mesma, além disso, como k1 tem um valor muito superior a k2 não terá muito efeito.
  + De qualquer forma, se o utilizador conseguir obter a *password*, o atacante terá a possibilidade de alterar as passwords sem que o utilizador se aperceba e mude a sua password antes do utilizador.
* ***Cache* e password do site roubada:**
  + Pode ocorrer se o máquina do cliente for roubada e este possuir um site malicioso na qual o atacante pode usar para aceder com a máquina do cliente obter a *password* especifica para este site através da autenticação do cliente.
  + Assim o atacante apenas necessitará de k2 computações no *hash* para cada tentativa para obter a *password* do cliente.

# Exercício 2

**Um programa é designado por “Cavalo de Tróia” se, parecendo cumprir um determinado objectivo legítimo, realiza outras acções não legítimas e que comprometem a integridade do sistema. Em que medida o controlo de acessos mandatório (MAC) impede esta ameaça? Use o modelo Biba como referência.**

O modelo Biba garante integridade do sistema pois não se pode ler informação de um nível inferior senão estar-se-ia a expôr informação de menor importância, isto porque não pode ser -1 dada a mesma relevância à informação de nível n-1 que no nível n.

Assim, se um cavalo de tróia for executado com intenção de ver informação trocada em ficheiros nos níveis inferiores e partilhá-la com os utilizadores de nível superior tal como acontecerá como expectado porque embora possa ser criado um novo ficheiro/s com a mesma informação a marca de integridade continuará a ser de um nível inferior pelo que o atacante nunca consegui ler este novo ficheiro, desta forma, é possível garantir que a integridade da informação de níveis inferiores não é afectada.

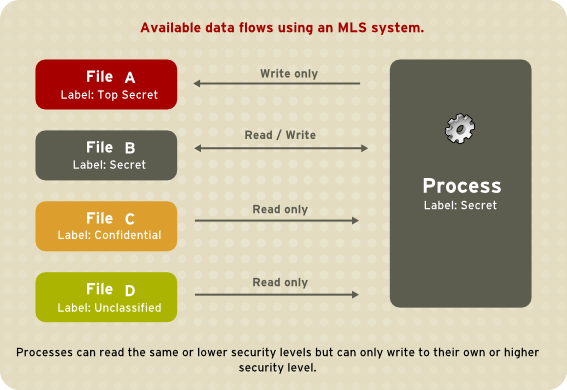
# Exercício 3

**Pretende-se implementar uma política de controlo de acesso, baseada no modelo de Bell-LaPadula, sobre um mecanismo que suporta o modelo RBAC1. A política de Bell-LaPadula é caracterizada por:**

* **Reticulado constituído pelo seguinte conjunto de etiquetas: , onde .**
* **Conjunto de utilizadores constituído por: onde o clearance do utilizador é igual a .**
* **Conjunto de recursos constituído por: onde a etiqueta do recurso é . Estes recursos apenas suportam duas operações: escrita e leitura.**

**Defina a política que implemente a política de Bell-LaPadula descrita anteriormente. Esta definição deve incluir: o conjunto de permissões, o conjunto de roles, o conjunto de utilizadores e as relações**

O modelo Bell La-Padula preserva a confidencialidade do sistema. A figura abaixo descreve este modelo:



Desta forma, começámos por analisar o RBAC1 que teria de ser desenvolvido para a leitura ser possível nos níveis: onde o utilizador está presente e nos inferiores a este.

Sabendo que cada utilizador tem associada uma role relativa ao ser nível de confidencialidade, definimos o seguinte conjunto:

**Permissões de Leitura**





**Permissões de Escrita**





# Exercício 4

**Considere o modelo de integridade de Clark e Wilson. Justique a divisão das regras deste modelo em**

**e .**

O modelo de integridade de Clark e Wilson foi criado com intenção de prevenir corrupção de dados e maliciosas intenções no sistema informático, pelo que Clark e Wilson defenderam que este modeloseria mais adquado para empresas em produção do que outros como Biba ou Bell-LaPadula. Por isso, Clark e Wilson criaram um conjunto de regras na qual se pretende tirar partido da vantagem da computação (através de software) e dos humanos para validar operações, esta última componente não requer computação porque poderia ser desenvolvida uma aplicação com intuito de quebrar o bom funcionamento desta validação.

Desta forma, Clark e Wilson criaram no seu modelo as regras de certificação (certification rules) e cumprimento de regras (enforcement rules) que manipulam dados integros (CDI), dados ainda não integros (UDI), procedimentos de transformação (TP) bem como de validação de integridade (IVP).

As certification rules: são cumpridas por humanos pelo estes devem garantir que a operação foi concluída e não houve qualquer extrapolamento no sistema.

As enforcement rules: são garantidas por software qualificado e que deve garantir que a operação a efectuar é possível perante informação do sistema.

Um pequeno exemplo de como diferenciar estas duas regras:

* é pretendida uma transferência bancária de 1000€.
* enforcement rules: deve ser garantido por software que a conta emissora tem pelo menos 1000€ no seu saldo(isto é, um dado integro - CDI), bem como validar a existência da conta receptora e por fim criar uma transacção (equivalente ao procedimento de transformação - TP) para transferir este saldo, esta transacção é necessária para garantir que se ocorre alguma falha no sistema todo o processo é anulado voltando os 1000€ para a conta receptora.
* certification rules: após a transferência concluída o humano (fundionário bancário) deve confirmar que na conta emissora existe um débito de 1000€ e na conta receptora existe um crédito de 1000€ (equivalente ao procedimento de validação de integridade - IVP - nas duas contas bancárias).

# Exercício 5

**Realize uma aplicação web para consulta de contactos de contas do serviço Gmail [2] ou informação de perfil de contas da rede social facebook. A aplicação obtém autorização para acesso aos recursos usando o protocolo OAuth 2.0.**

**5.1 Na alínea anterior o authorization token, fornecido pelo servidor de autorização, é entregue à aplicação web através de um callback HTTP. Pretende-se agora que este token seja entregue por HTTPS (e.g. https://localhost:8081/alinea5callback). Para tal, deve configurar a infra-estrutura que suporta a execução da aplicação para aceitar ligações HTTPS, usando o certicado localhost e respectiva chave privada.**

# Exercício 6

**Implemente um componente .NET que realize as func~oes de Policy Decision Point (PDP) com os seguintes requisitos:**

* **Suporte para o modelo RBAC1, em que os utilizadores, permissões e roles são definidos por cadeias de caracteres.**
* **Utilização dos cheiros de configuração da plataforma .NET ou de outro repositório alternativo (ex.base de dados) para o armazenamento da política.**
* **Independência do tipo de recurso controlado.**

# Exercício 7

**Pretende-se que seja desenvolvida uma aplicação com um Policy Enforcement Point (PEP) que usa o PDP realizado na alínea anterior.**

* **Crie um filtro ASP.NET MVC para aplicar a metodos de controladores. O ltro verica se o utilizador do pedido corrente tem a permissão exigida nos parâmetros do ltro. A informação de identidade é obtida após autenticação usando um dos métodos suportados pela infra-estrutura ASP.NET MVC.**

**Por omissão, assume-se que cada sessão RBAC activa todos os roles permitidos para o utilizador associado.**