Curso	**Engenharia Informática**			*Ano letivo*		2012/13	
Unidade Curricular	**Programação e Segurança**			ECTS		4	
Regime	**Opcional**						
Ano	**2º/3º**	*Semestre*	**2º sem**	*Horas de trabalho globais*			
Docente (s)	**José Carlos Fonseca**			*Total*	112	*Contacto*	75
Coordenador da área disciplinar	**José Carlos Fonseca**						

GFUC previsto

1. OBJETIVOS DE APRENDIZAGEM

Após a conclusão da UC, os alunos deverão ser capazes de:

1. Desenvolver software de acordo com a legislação nacional e normas internacionais para segurança do software
2. Usar um processo de desenvolvimento de software seguro
3. Identificar as ameaças, vulnerabilidades e ataques ao software mais comuns
4. Aplicar os controlos de segurança adequados ao projecto de software
5. Usar os vários tipos de encriptação para aumentar a segurança do software

2. CONTEÚDOS PROGRAMÁTICOS

1. Segurança do software e segurança da informação
2. Legislação sobre segurança
3. Normas internacionais de certificação de segurança
    a. NIST
    b. Common Criteria
    c. ISO/IEC 27001
    d. PCI-DSS
4. Ciclos de vida de desenvolvimento de software seguro

   a. CLASP
   b. Touchpoints
   c. Microsoft SDL

5. Ameaças, vulnerabilidades e ataques mais comuns
   a. Buffer overrun
   b. SQL Injection
   c. XSS
   d. CSRF
   e. Outros
6. Controlos de segurança
   a. Proteção do processo de autenticação
   b. Manutenção de sessão
   c. Controlo de acesso
   d. Passwords
7. Encriptação
   a. Simétrica
   b. Fluxo
   c. Troca segura de chaves de encriptação usando Diffie-Hellman
   d. Assimétrica
   e. Hashing
   f. Certificados digitais

3. DEMONSTRAÇÃO DA COERÊNCIA DOS CONTEÚDOS PROGRAMÁTICOS COM OS OBJETIVOS DA UC

1. Os Conteúdos 1, 2 e 3 estão coerentes com o Objetivo 1, pois focam aspectos fundamentais da segurança, a legislação portuguesa vigente e normas internacionais de segurança do software.
2. O Conteúdo 4 coerente com o Objetivo 2, pois foca os processos de desenvolvimento de software seguro mais usados na indústria.
3. O Conteúdo 5 coerente com o Objetivo 3, pois foca as ameaças, vulnerabilidades e ataques ao software mais comuns, como se manifestam e como podem ser minimizados.
4. O Conteúdo 6 coerente com o Objetivo 4, pois foca a aplicação de controlos para melhorar a segurança do software.
5. O Conteúdo 7 coerente com o Objetivo 5, pois foca as técnicas e algoritmos de encriptação e a sua aplicação no desenvolvimento de software.

4. BIBLIOGRAFIA PRINCIPAL

Obrigatória:

1. Apontamentos fornecidos pelos docentes
2. Whitman, M. e Mattord, H. (2011), Principles of Information Security, Cengage Learning
3. Gregory, P. (2010), CISSP Guide to Security Essentials, Cengage Learning
4. Dafydd Stuttard, Marcus Pinto, (2011), The Web Application Hacker's Handbook, 2nd edition, Wiley Publishing, Inc.
5. Michael Howard, David LeBlanc, (2003), Writing Secure Code, 2nd edition, Microsoft Press

Recomendada:

1. Michael Howard, David LeBlanc, (2005), 19 Deadly Sins of Software Security: Programming Flaws and How to Fix Them, McGraw-Hill/Osborne

2. William Stallings, (2011), Cryptography and Network Security Principles and Practices, 5th edition, Prentice Hall

3. Nuno Carvalho (2009) Organizações e Segurança Informática, Lugar da Palavra Editora

4. Zúquete, A. (2010), Segurança em Redes Informáticas, FCA Editora

5. METODOLOGIAS DE ENSINO (REGRAS DE AVALIAÇÃO)

Metodologias de ensino:

1. Lição expositiva
2. Lição interativa
3. Resolução de problemas
4. Trabalho de projeto

Regras de avaliação:

Avaliação contínua:

1. Teste escrito. (25%)
2. Trabalho prático realizado ao longo do semestre. É avaliado uma única vez, não havendo possibilidade de melhoria. Pode ser realizado fora da sala de aula. (75%)

Avaliação por exame final na Época Normal, Época de Recurso ou Época Especial:

1. Teste escrito. (25%)
2. Teste escrito avaliando a componente prática, ao qual o aluno poderá ficar dispensado caso seja avaliado pelo trabalho prático realizado ao longo do semestre. (75%)

6. DEMONSTRAÇÃO DA COERÊNCIA DAS METODOLOGIAS DE ENSINO COM OS OBJETIVOS DA UNIDADE CURRICULAR

1. Lição expositiva está coerente com os objetivos devido à necessidade de apresentar os conteúdos teóricos aos alunos, nomeadamente os vários aspetos relacionados com a segurança, a legislação e normas aplicáveis.

2. Lição interativa está coerente com os objetivos pois a interação alunos/docentes ajuda a aprendizagem dos conceitos para além da introdução de novas ideias, perspetivas e soluções que podem ser aplicadas tanto na fase de análise como na de desenvolvimento de software seguro, tendo em conta os agentes externos e como minimizar os seus efeitos.

3. Resolução de problemas está coerente com os objetivos pois a aplicação de conteúdos teóricos a exercícios práticos de inspiração realista, relacionados com o estudo da segurança do software, a aplicação dos controlos adequados, incluindo a encriptação, perante as possíveis ameaças, vulnerabilidades e ataques, ajuda a consolidar a matéria, realçando o saber fazer.

4. Trabalho de projeto está coerente com os objetivos pois abrange o desenvolvimento de software seguro, passando por todas as fases desde a sua concepção até à sua utilização, pelo que obriga à aplicação prática de todos os conceitos abordados ao longo do semestre a uma situação realista nova.

7. CONTATOS E HORÁRIO DE ATENDIMENTO

José Carlos Fonseca

josefonseca@ipg.pt

Gab. 25

Horário de atendimento:

3ª 11:30 – 13:30

5ª 16:30 – 17:30

Data: 21/11/2012

Docente e Coordenador da área disciplinar