Universidade Federal da Fronteira Sul

Plano de Ensino

1. Dados de Identificação

Curso: Ciência da Computação
Componente Curricular: GEX112 - Segurança e auditoria de sistemas
Turma: 14957 - Ciência da Computação - 9ª Fase - Noturno - 2016/2
Numero de Créditos: 4
Carga horária - Hora Aula: 72
Carga horária - Hora Relógio: 60
Professor: Emílio Wuerges
Atendimento ao aluno:

- Quartas: 14:00 até 21:00, mediante horário marcado

2. Objetivo Geral do Curso

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional.

3. Ementa

Segurança em aplicações: programação segura, detecção de falhas, códigos maliciosos (malware). Segurança em sistemas operacionais: princípios de controle de acesso, sistemas confiáveis. Segurança em redes de computadores: ataques e defesas. Princípios de criptografia: criptografia simétrica e assimétrica, integridade de dados. Protocolos de autenticação: princípios, infra-estrutura de chaves públicas e aplicações, e protocolos criptográficos.

4. Objetivo

4.1 Geral

- Reconhecer e relacionar os principais riscos envolvidos no ambiente de informações e os principais pontos de controle de auditoria da tecnologia da informação no que se refere à auditoria do desenvolvimento e manutenção de sistemas, administração de dados, administração de banco de dados, e administração de redes de computadores.

4.2 Específicos

- Conhecer os principais mecanismos de criptografia clássica.
- Conhecer criptoanálise de criptografia clássica.
- Conhecer as principais técnicas de criptografia moderna: Funções hash, Criptografia de chave simétrica e criptografia de chave assimétrica.
- Conhecer técnicas de programação e as principais vulnerabilidades que ocorrem em *software*.

5. Cronograma e Conteúdo Programático

Encontros	Conteúdo	Atividade
1 e 2	Troca de chaves por Diffie & Hellman	Implementar DH.
3 e 4	Introdução a criptografia Simétrica	Implementar cifrador/decifrador usando AES256, usando bibliotecas da linguagem.
5 e 6	Introdução a criptografia assimétrica	Implementar troca de mensagens seguras usando RSA.
7 e 8	Introdução a hashes criptográficos	Implementar gerador de certificados digitais, usando RSA + SHA256.
9 e 10	Vulnerabilidades em código	Explorar vulnerabilidade de Buffer Overflow.
11 e 12	Criptografia clássica	Implementar cifradores e decifradores para as cifras clássicas: • Cifra de césar • Cifra de vigénere • Cifra de transposição • Cifra de substituição
13 e 14	Ataque com texto em claro	Implementar quebrador para cifras clássicas de posse das mensagens em claro.
15 e 16	Ataques de forca bruta	Implementar gerador de chaves para cifras de césar, vigénere e transposição.
17 e 18	Ataque com dicionário	• Elaborar Dicionario.

		• Implementar quebrador para cifra de césar, vigénere e transposição de posse apenas das mensagens cifradas.
19 e 20	Entropia	Implementar análise e gerar relatórios das: • Frequências dos símbolos. • Frequências dos padrões de palavras.
21 a 36	Criptoanálise baseada em entropia	Implementar quebrador para cifra de substituição de posse apenas das mensagens cifradas. 1. Usando apenas frequências dos símbolos. 2. Incorporando padrões de palavras. 3. Usando busca heurística.

6. Procedimentos Metodológicos (estratégias de ensino, equipamentos, entre outros)

Conduzir a disciplina através de exercícios práticos de implementação.
Discutir as implicações práticas dos resultados teóricos conhecidos através de avaliação construtiva.
O reuso de código de terceiros é incentivado, entretanto a nota será proporcional ao conteúdo original.
O uso da ferramenta de controle de versão GIT é obrigatorio. Quando código for reusado, o aluno deve indicar o repositório do qual o código foi incorporado. Caso contrário, a média do aluno será zero e a ocorrência será comunicada ao colegiado do curso.

7. Avaliação do Processo Ensino-Aprendizagem

Avaliação construtiva dos trabalhos de implementação. Serão feitos diversos trabalhos de implementação.
A média final será a menor nota dentre todos os trabalhos.

7.1 Recuperação: novas oportunidades de aprendizagem e avaliação

A recuperação será feita através do retrabalho da tarefa a ser recuperada.
A nota da tarefa retrabalhada substituirá a nota da tarefa original, entretanto, para ter direito de fazer a recuperação é necessário o aluno ter entregado a tarefa original no prazo.
O prazo máximo para a entrega da tarefa retrabalhada é de duas semanas após a entrega da tarefa original.

8. Referências

8.1 Básicas

SCHMIDT, Paulo; ARIMA, Carlos Hideo; SANTOS, José Luiz dos. Fundamentos de Auditoria de Sistemas. São Paulo: Atlas, 2006.
ONOME, J. Auditoria de Sistemas de Informação. Rio de Janeiro: Editora Atlas, 2005.
GREG, Hoglund; GARY, Macgraw. Como quebrar códigos – a arte de explorar e proteger software. São Paulo: Makron Books, 2006.
GIL, A. Auditoria de Computadores. São Paulo: Atlas 2000.

8.2 Complementares

MELLO, Sandro. Computação forense com software livre – conceitos, técnicas, ferramentas e estudos de casos. São Paulo: Atlas, 2009.
CARUSO, Carlos A. A.; STEFFEN, Flávio D. Segurança em Informática e de Informações. 2. ed. São Paulo: Senac, 1999.
GIL, A. L. Fraudes Informatizadas. São Paulo: Atlas, 1999.
DIAS, C. Segurança e Auditoria da Tecnologia da Informação. Rio de Janeiro: Editora Axcel Books, 2000.
SCHMIDT, P. Fundamentos de Auditoria de Sistemas. São Paulo: Editora Atlas, 2006.

Professor
SIAPE 2852314

Coordenador
MARCO AURÉLIO SPOHN
Siape nº.1521671
Coord. do Curso de Ciência da Computação
Universidade Federal da Fronteira Sul-UFFS
Campus Chapecó-SC