Secure Coding for Banking and Finance Eğitimi

Açıklama

Sadece bankacılık ve finans sektörü için uygulama geliştiren mühendisleri hedefleyen özel bir eğitim programı sunuyoruz. Özel eğitmenlerimiz, deneyimlerini ve uzmanlıklarını uygulamalı laboratuvarlar aracılığıyla paylaşıyor ve bankacılık sektöründe uygulanmış gerçek hayattan alınmış çözüm örnekleri sunuyorlar. Katılımcılar eğitimde güvensiz kodlamanın neden olduğu korsanlık olaylarının sonuçlarını eğlenceli bir şekilde görebiliyorlar.

Bu eğitimde neler öğreneceksiniz?

- Güvenlik, BT güvenliği ve güvenli kodlamanın temel kavramları
- Bankacılık ve finans sektörüne özel tehditler
- Regülasyonlar ve standartlar
- OWASP Top Ten sıralamasının ötesindeki Web güvenlik boşlukları ve bunlardan nasıl kaçınılacağı
- XML güvenliği hakkında bilgiler
- İstemci tarafı güvenlik boşlukları ve güvenli kodlama uygulamaları
- JSON güvenliği hakkında genel bilgiler
- DDoS saldırıları ve bu saldırılara karşı korunma yöntemleri
- Kriptografı hakkında pratik bilgiler
- Temel güvenlik protokolleri
- Güvenli kodlama uygulamaları hakkında çeşitli kaynaklar ve diğer bilgiler

Eğitim İçeriği

IT security and secure coding

- Nature of security
- What is risk?
- IT security vs. secure coding
- From vulnerabilities to botnets and cybercrime
- Classification of security flaws

Special threats in the banking and finance sector

- Banking and finance threats – trends
- Banking and finance threats – some numbers
- Attacker profiles
- Most significant targets
- Attacker tools and vectors

Regulations and standards

- The fintech cybersecurity regulatory / compliance landscape
- Important organizations and regulations from an IT standpoint
- Data protection
- Breach disclosure obligations
- PCI DSS compliance

Web application security

- A1 – Injection
- A2 – Broken authentication
- A3 – Sensitive data exposure

Web application security

- A4 – XML external entity (XXE)
- A5 – Broken access control
- A6 – Security misconfiguration
- A7 – Cross-Site Scripting (XSS)
- A8 – Insecure deserialization

- A9 – Using components with known vulnerabilities
- A10 – Insufficient logging and monitoring

Client-side security

- JavaScript security
- Same Origin Policy
- Simple requests
- Preflight requests
- Exercise – Client-side authentication
- Client-side authentication and password management
- Protecting JavaScript code
- Clickjacking
- AJAX security
- HTML5 security

XML security

- Introduction
- XML parsing
- XML injection

JSON security

- Embedding JSON server-side
- JSON injection
- JSON hijacking
- Case study – XSS via spoofed JSON element

Denial of service

- DoS introduction
- Asymmetric DoS
- Case study – ReDos in Stack Exchange
- Hashtable collision attack

Practical cryptography

- Rule #1 of implementing cryptography
- Cryptosystems
- Symmetric-key cryptography

- Other cryptographic algorithms
- Asymmetric (public-key) cryptography
- Public Key Infrastructure (PKI)

Security protocols

- Secure network protocols
- Specific vs. general solutions
- SSL/TLS protocols
- Improper use of security features
- Input validation

Principles of security and secure coding

- Matt Bishop’s principles of robust programming
- The security principles of Saltzer and Schroeder
- SEI Cert top 10 secure coding practices

Ön Koşullar

C# Programlama tecrübesi