Ministério da Fazenda

Serviço Federal de Processamento de Dados (SERPRO)

Segurança no Desenvolvimento

Palestrante: Daniel Araújo Melo – Grupo de Resposta a Ataques da Intranet

00/00/0000

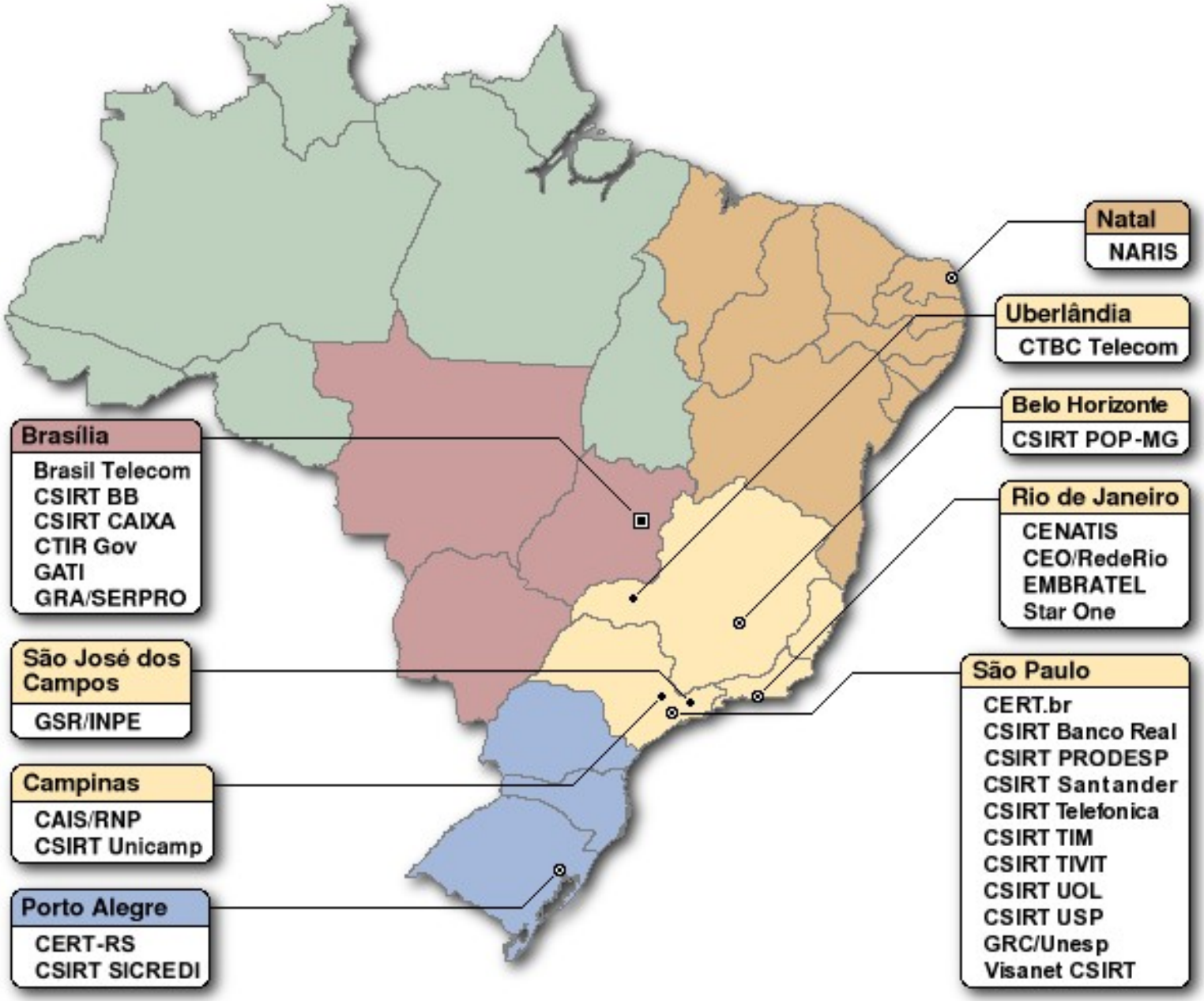
Agenda

- Apresentação do Grupo de Resposta a Ataques
- Segurança no Desenvolvimento
- Melhores Práticas
- ISO 15408
- OWASP
- BSIMM

Grupo de Resposta a Ataques

- CSIRT Framework
  - Computer Incident Response Team
  - www.cert.org, www.cert.br
- Serviços Oferecidos
  - Detecção e Prevenção de Intrusão
  - Análise de Vulnerabilidades
  - Análise Forense Computacional
  - Treinamentos
  - Consultoria

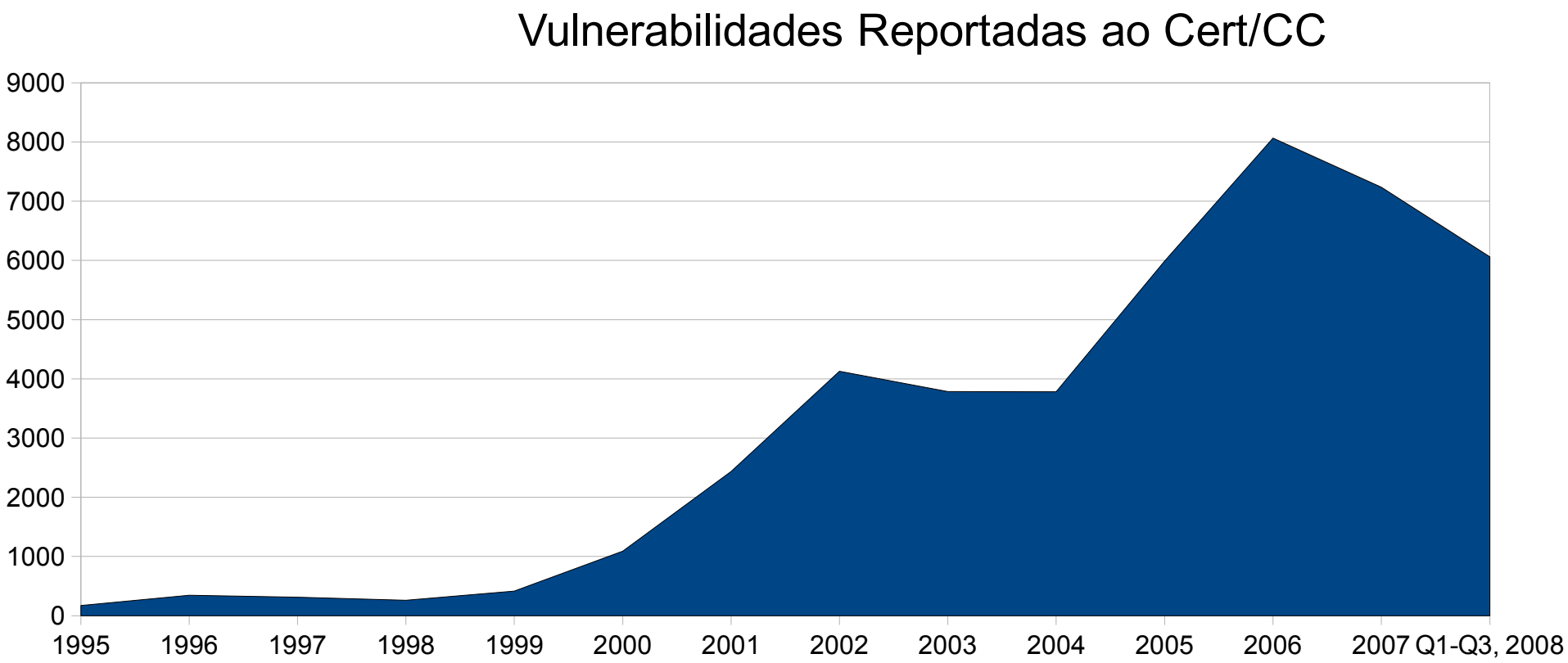
Csirts no Brasil



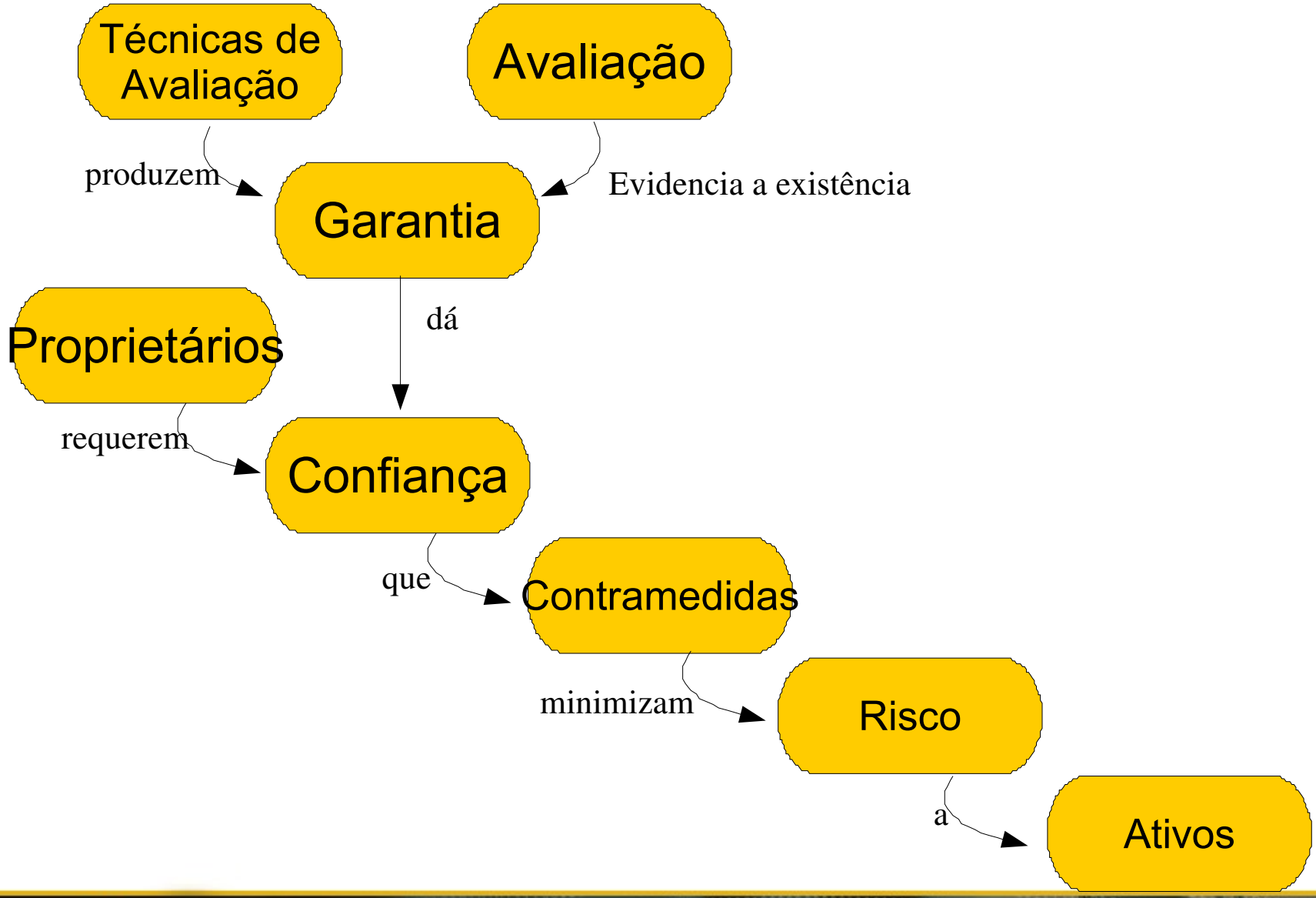
Vulnerabilidades

- A melhor forma de quebrar a segurança de um sistema é evitá-la...
  - Fonte www.securecoding.cert.org

Vulnerabilidades Reportadas



Contexto da Segurança



Avaliação da Segurança

- Pode melhorar produtos de TI de duas formas:
  - Identifica erros ou vulnerabilidades que podem ser corrigidas pela(o) Desenvolvedor(a), reduzindo a probabilidade de futuras falhas de segurança;
  - A Preparação para uma avaliação faz com que a(o) Desenvolvedor(a) tome mais cuidado com a estrutura e desenvolvimento do Sistema.

Segurança em Desenvolvimento

- Segurança no Ambiente de Desenvolvimento;
  - Manter fontes em segurança, evitar roubo de código ou indisponibilidade da equipe de desenvolvimento.
- Segurança do Sistema;
  - Seguir especificação de segurança, evitar falhas(vulnerabilidades) que comprometam a segurança.
  - Buffer Overflows, Backdoors, etc.
- Garantia de Segurança do Sistema.
  - Garantir ao Cliente a segurança do Sistema.

Ambiente de Desenvolvimento Seguro

- Algumas Características de um Ambiente Seguro:
  - Espaço físico restrito, com controle de acesso físico e proteção lógica dos servidores;
  - Separação entre ambiente de desenvolvimento, teste e construção (build);
  - Gerência de configuração dos fontes;
  - Processos de desenvolvimento bem estabelecidos e controlados, gerando evidências dos controles.

Exposição do Ambiente

- Scrapkut worm
  - Orkut
  - Contaminou 400.000 usuários
- Js-Exploit Messenger
  - Se replica através do Messenger
- Samy (also known as JS.Spacehero)
  - Myspace
  - 1.000.000 de infecções em 20 horas

Segurança do Sistema

- Normas e práticas de boa programação:
  - Funções intrinsecamente seguras;
  - Verificar códigos de erro retornado por função ou método;
  - Atentar para tamanho de *buffers* e *arrays* do sistema;
  - Documentar o código;

Top 10 Secure Coding Practices

- 1. Validate input.
- 2. Heed compiler warnings.
- 3. Architect and design for security policies.
- 4. Keep it simple.
- 5. Default deny.
- 6. Adhere to the principle of least privilege.
- 7. Sanitize data sent to other systems.
- 8. Practice defense in depth.
- 9. Use effective quality assurance techniques.
- 10. Adopt a secure coding standard.
- Bonus: Define security requirements, threat modeling
  - Fonte: www.securecoding.cert.org

Garantia de Segurança do Sistema

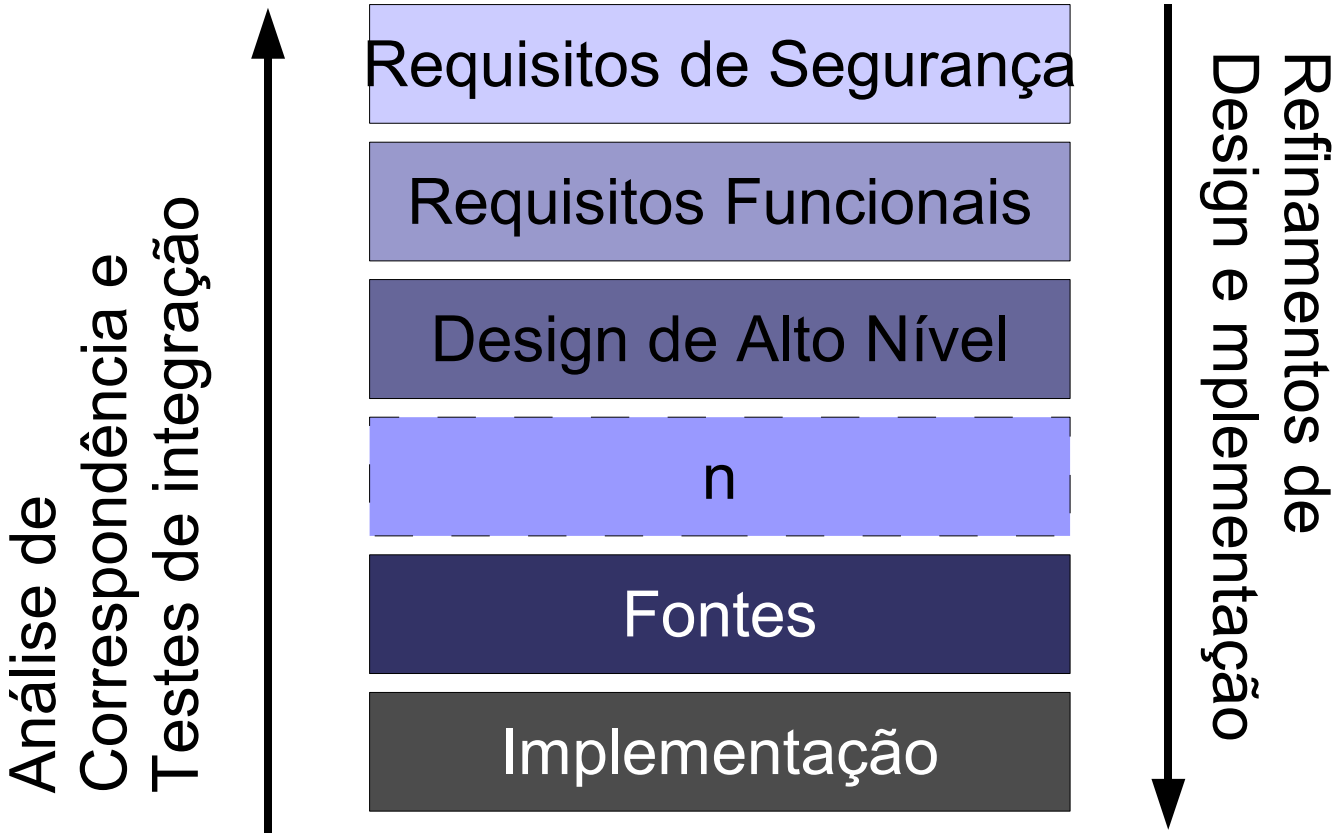
- Especificar a segurança de forma clara e objetiva;
- Construir conforme especificação;
- Testar para verificar se atende a especificação original

Common Criteria for Information Technology Security Evaluation

- Homolagada como ISO/IEC 15.408 – também chamada de Common Criteria ou CC.
- Objetivo:
  - Fornecer conjunto de critérios fixos que permitem especificar a segurança de uma aplicação de forma não ambígua a partir de características do ambiente da aplicação, e definir formas de garantir a segurança da aplicação para o cliente final.
- Evolução do TSEC – Trusted Computer Security Evaluation Criteria - “Orange Book”

Metodologia de Desenvolvimento

- Nenhuma metodologia específica



ISO 15.408

- Define 4 níveis de garantia de segurança (EAL – Evaluation Assurance Level)
  - EAL 1 – Testado funcionalmente
    - *Garante que o sistema funciona de acordo com o especificado em sua documentação, e que esta descreve as proteções necessárias contra ameaças identificadas.*
    - *Aplicável quando alguma confiança no funcionamento é necessária, porém as ameaças à segurança não são vistas como sérias.*
    - *A avaliação é feita na versão entregue ao Cliente, com base na documentação, sem o auxílio do desenvolvedor.*

ISO 15.408

- Continuação:
  - EAL 2 – Testado Estruturalmente
    - *Garante que foram utilizadas as práticas comerciais padrão no desenvolvimento;*
    - *Requer uma análise de vulnerabilidades do sistema, testes independentes e validação dos testes e estrutura utilizados pelo desenvolvedor, que também será avaliado.*
    - *Aplicável nos casos em que desenvolvedores e usuários requerem um nível entre baixo e moderado, de garantia de segurança. Ex.: sistemas legados ou nos quais o desenvolvedor possui acesso limitado.*

ISO 15.408

- Continuação:
  - EAL 3 – Metodicamente testado e verificado
    - *Garante que a segurança foi aplicada no estágio de projeto do sistema, sem alteração substancial nas práticas de desenvolvimento;*
    - *Requer investigação completa do sistema e do seu desenvolvimento, sem a necessidade de reengenharia.*
    - *Aplicável nos casos em que desenvolvedores e usuários requerem nível moderado de garantia de segurança.*

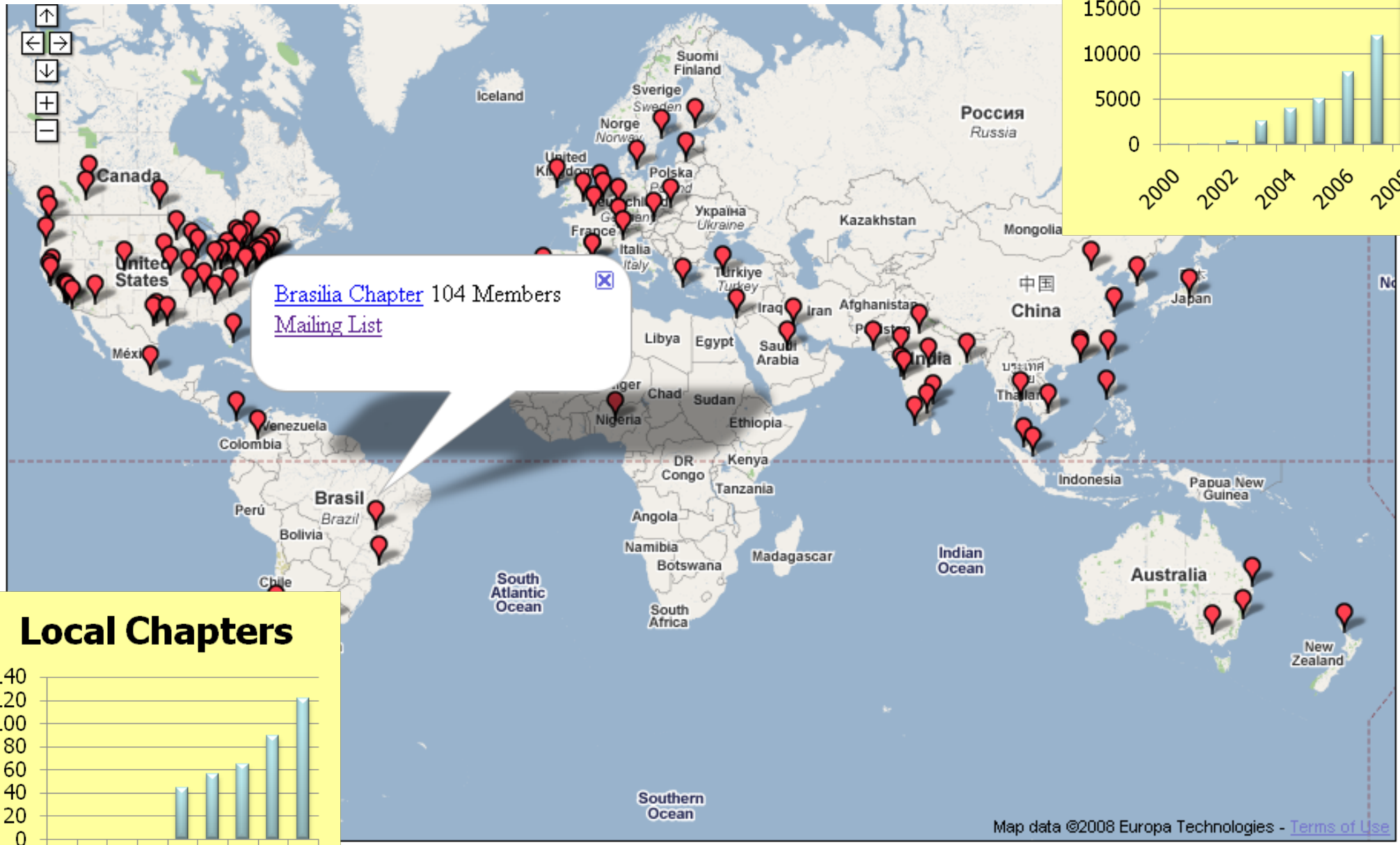
ISO 15.408

- Continuação:
  - EAL 4 – Metodicamente projetado, testado e verificado
    - *Garante o máximo em segurança baseada em boas práticas de desenvolvimento, sem a utilização de especialista.*
    - *Analisa as funções de segurança do Sistema utilizando:*
      - *especificações funcional e de interfaces;*
      - *estruturas de alto nível(high-level design) e baixo nível(low-level design);*
      - *subconjunto da implementação;*
      - *modelo informal da política de segurança do sistema.*

OWASP

- Open Web Application Security Project
  - Comunidade aberta
  - Dedicada a habilitar organizações à desenvolver, adquirir e manter aplicações que possam ser confiáveis.
- Ferramentas e Documentos organizados nas seguinte categorias
  - Proteção
    - *proteção contra falhas de implementação e design de segurança*
  - Detecção
    - *Detecção de falhas de implementação e design de segurança*
  - Ciclo de vida
    - *Adicionam segurança ao ciclo de vida de desenvolvimento de software.*

OWASP Comunidade Mundial



OWASP

- Ferramentas
  - Proteção
    - *OWASP AntiSamy Java Project*
    - *OWASP AntiSamy .NET Project*
    - *OWASP Enterprise Security API (ESAPI) Project*
  - Detecção
    - *OWASP Live CD Project*
    - *OWASP WebScarab Project*
  - Ciclo de vida
    - *OWASP WebGoat Project*

OWASP

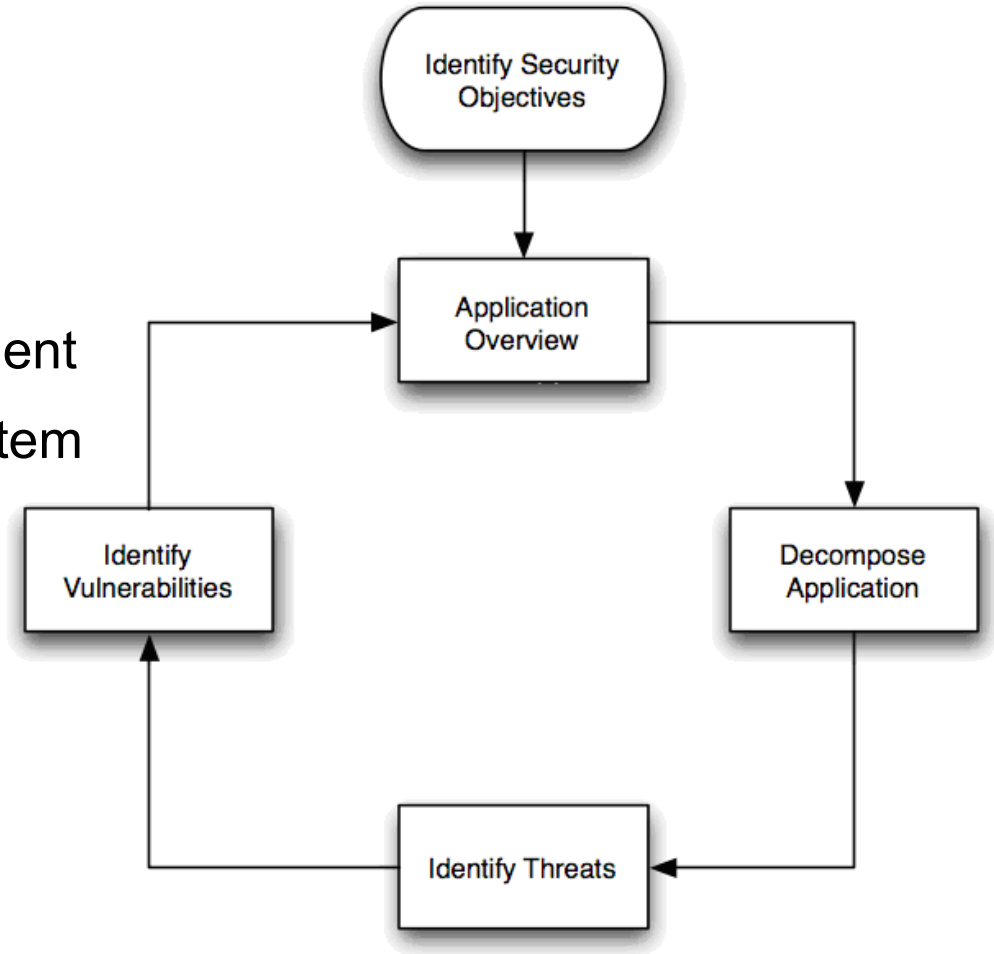
- Documentos
  - Proteção
    - *OWASP Development Guide*
    - *OWASP Ruby on Rails Security Guide V2*
  - Detecção
    - *OWASP Code Review Guide*
    - *OWASP Testing Guide*
    - *OWASP Top Ten Project*
  - Ciclo de vida
    - *OWASP AppSec FAQ Project*
    - *OWASP Legal Project*
    - *OWASP Source Code Review for OWASP-Projects*

Owasp top 10

- A1 - Cross Site Scripting (XSS)
- A2 - Injection Flaws
- A3 - Malicious File Execution
- A4 - Insecure Direct Object Reference
- A5 - Cross Site Request Forgery (CSRF)
- A6 - Information Leakage and Improper Error Handling
- A7 - Broken Authentication and Session Management
- A8 - Insecure Cryptographic Storage
- A9 - Insecure Communications
- A10 - Failure to Restrict URL Access

Modelagem de Ameaças com Owasp

- Owasp indica:
  - Microsoft threat modelling
- Outras abordagens:
  - Trike
  - AS/NZS 4360:2004 Risk Management
  - Common Vulnerability Scoring System
  - OCTAVE
-
-
-
  - Fonte:
    - *http://www.owasp.org/index.php/Threat_Risk_Modeling*



The Building Security In Maturity Model

- BSIMM - http://www.bsi-mm.com
  - Modelo de maturidade para segurança no desenvolvimento
  - SSF – Security Software Framework
  - Práticas adotadas por nove das mais bem sucedidas iniciativas em segurança do desenvolvimento:
    - *Adobe, EMC, Google, Microsoft, QUALCOMM, Wells Fargo,The Depository Trust and Clearing Corporation (DTCC)*
- Apresenta 2 papéis (roles) para implementação
  - Liderança executiva
    - *Accountability e enpowerment*
  - SSG – Software Security Group
    - *Pessoal com experiência em codificação e arquiteturas.*
    - *1 membro para cada 100 desenvolvedores.*

The Building Security In Maturity Model

- SSF
  - 12 práticas em 4 domínios

The Software Security Framework (SSF)			
Governance	**Intelligence**	**SSDL Touchpoints**	**Deployment**
Strategy and Metrics	Attack Models	Architecture Analysis	Penetration Testing
Compliance and Policy	Security Features and Design	Code Review	Software Environment
Training	Standards and Requirements	Security Testing	Configuration Management and Vulnerability Management

FIM

- Photo credits: “365/25” originally uploaded by teachingsagittarian
- http://www.flickr.com/photos/teachingsagittarian/3225013302/

Contato

daniel.melo@serpro.gov.br
TIGRA – Grupo de Resposta a Ataques da Intranet