

Fundamentos de Segurança Informática (FSI)

2022/2023 - LEIC

Manuel Barbosa
mbb@fc.up.pt

Hugo Pacheco
hpacheco@fc.up.pt

Aula 2

"Segurança"

Correção \subset Segurança

- Correção: bom input \Rightarrow bom output
 - e.g. “utilizador e data *válidos*” \Rightarrow “histórico do utilizador nessa data”
- Segurança: mau input \nRightarrow mau output
 - e.g., “dados fidedignos” + “outros dados **nunca** são revelados”



Dois modelos: Binário

seguro vs inseguro

- Típico em criptografia e sistemas confiáveis
- Definir formalmente capacidades dos atacantes X
- Definir formalmente objetivos de segurança Y
- Nenhum atacante limitado a X consegue quebrar Y
- Terminologia típica: prova de segurança, secure by design



Dois modelos: Gestão de Risco

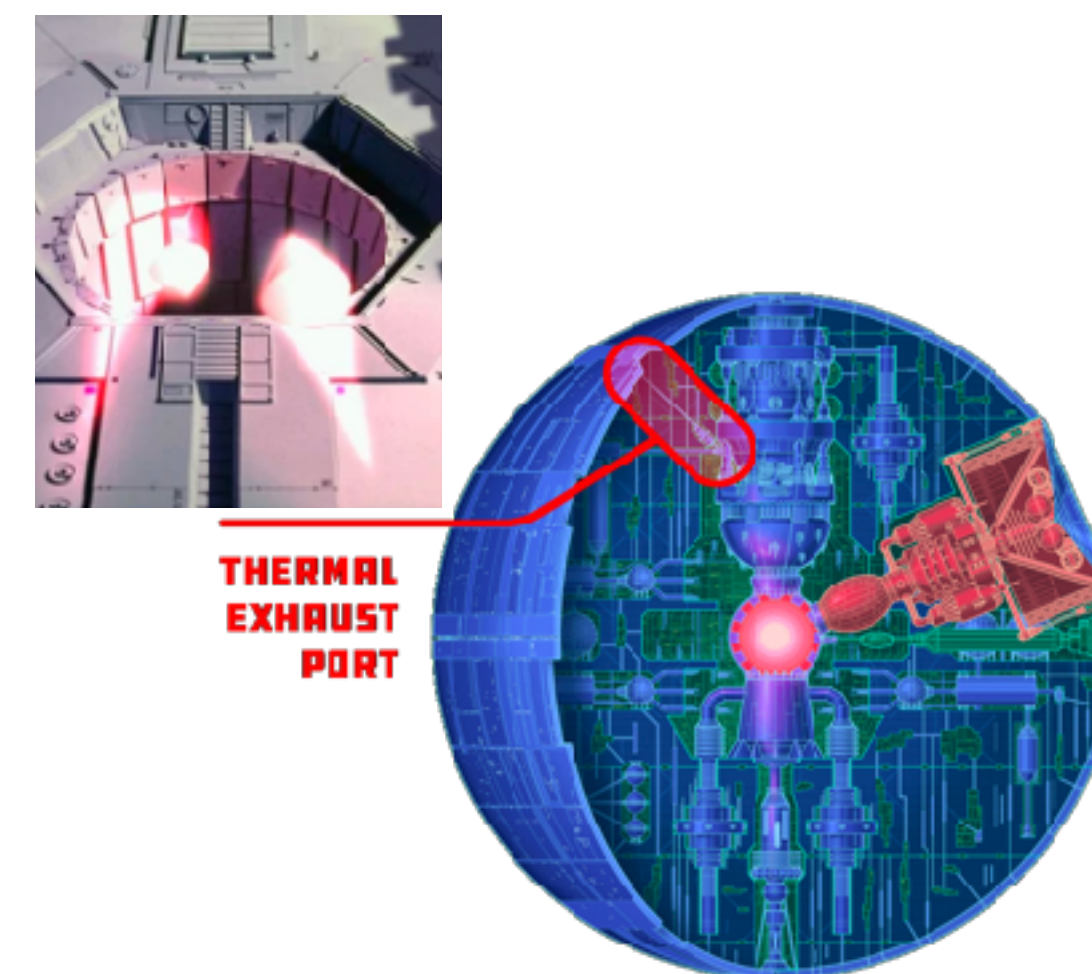
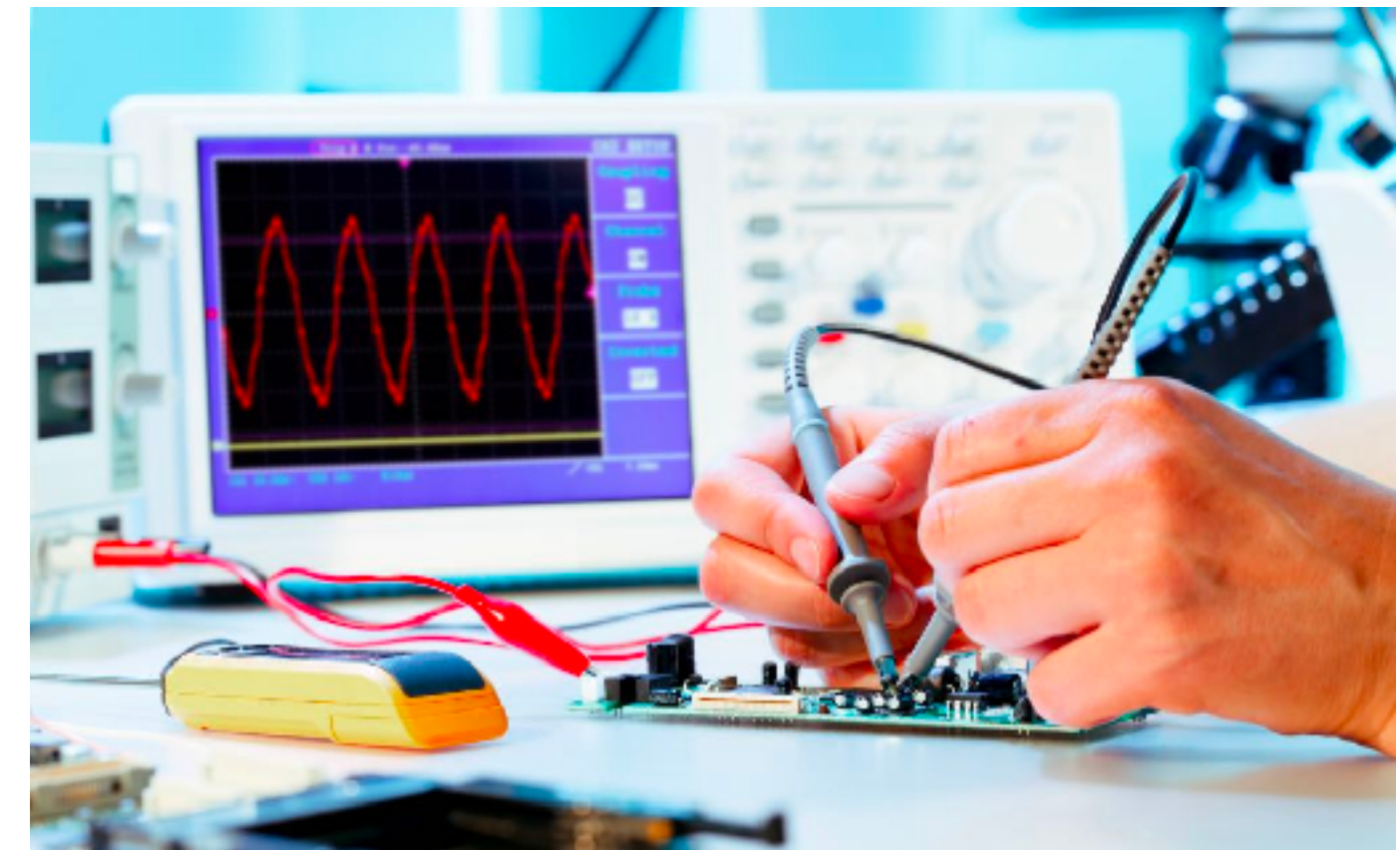
mais seguro vs menos seguro

- Típico em engenharia de software e segurança no mundo real
- Minimizar risco em função das ameaças mais prováveis
- Optimizar o custo das medidas de segurança vs potenciais perdas
- Terminologia típica: resiliência, mitigação, risco



Ambos têm limitações

- Modelo binário
 - Não escala para sistemas complexos
 - Modelo abstrato \neq sistema concreto
 - Os modelos formais podem ignorar problemas reais, e.g., side-channels
- Gestão de risco:
 - A análise de risco pode estar errada
 - Nunca estamos verdadeiramente seguros (se é que isso existe!)
 - Uma ameaça mal classificada pode deitar tudo por terra

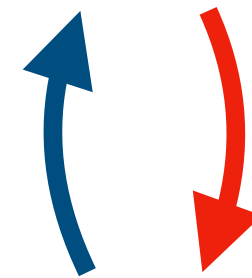


[fonte: Star Wars: Episode IV]

Em geral

- A segurança é um processo contínuo

Adversário inventa um novo ataque

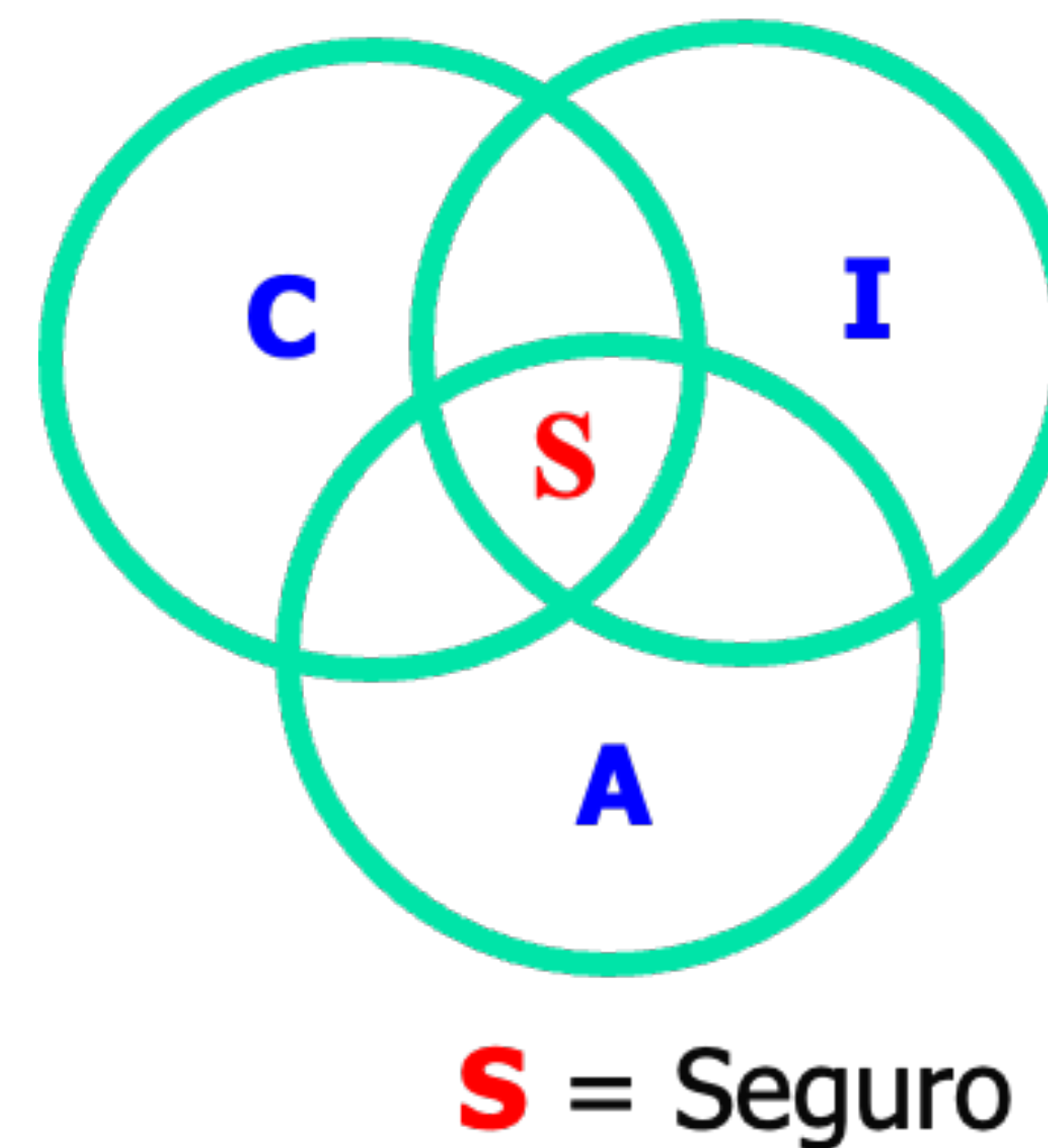


Defensor cria uma nova defesa

- O melhor a que podemos ambicionar é um equilíbrio ao longo do tempo

Segurança = CIA

- 3 eixos essenciais de segurança:
 - **Confidencialidade** (segredo, privacidade)
 - **Integridade** (não alteração, dados fidedignos, autenticidade de origem)
 - **Disponibilidade** (existência, constância)



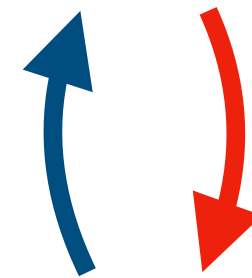
Sempre um compromisso

- Exemplo absurdo 1: C vs IA
 - Desconectar sistema da Internet para aumentar confiabilidade
 - Nenhuma disponibilidade, integridade pode sofrer sem updates
- Exemplo absurdo 2: I vs CA
 - Extensas verificações digitais e manuais dos dados para aumentar integridade
 - Confidencialidade em causa porque mais agentes acedem aos dados, disponibilidade sofre se processo de verificação “congelar” os dados
- Exemplo absurdo 3: A vs CI
 - Sistema geodistribuído na cloud com redundância para garantir *uptime*
 - Confidencialidade em causa se toda a Internet tem acesso, integridade em causa se dados forem alterados *in transit*

Em geral

- A segurança é um processo de gestão de custos

Quanto custaria um possível ataque



Quanto custa implementar defesa

- Por vezes sai mais barato simplesmente aceitar o risco de que o ataque ocorra (e.g., fraude no cartão de crédito vs ataque nuclear)

Gestão de Risco: Ativos

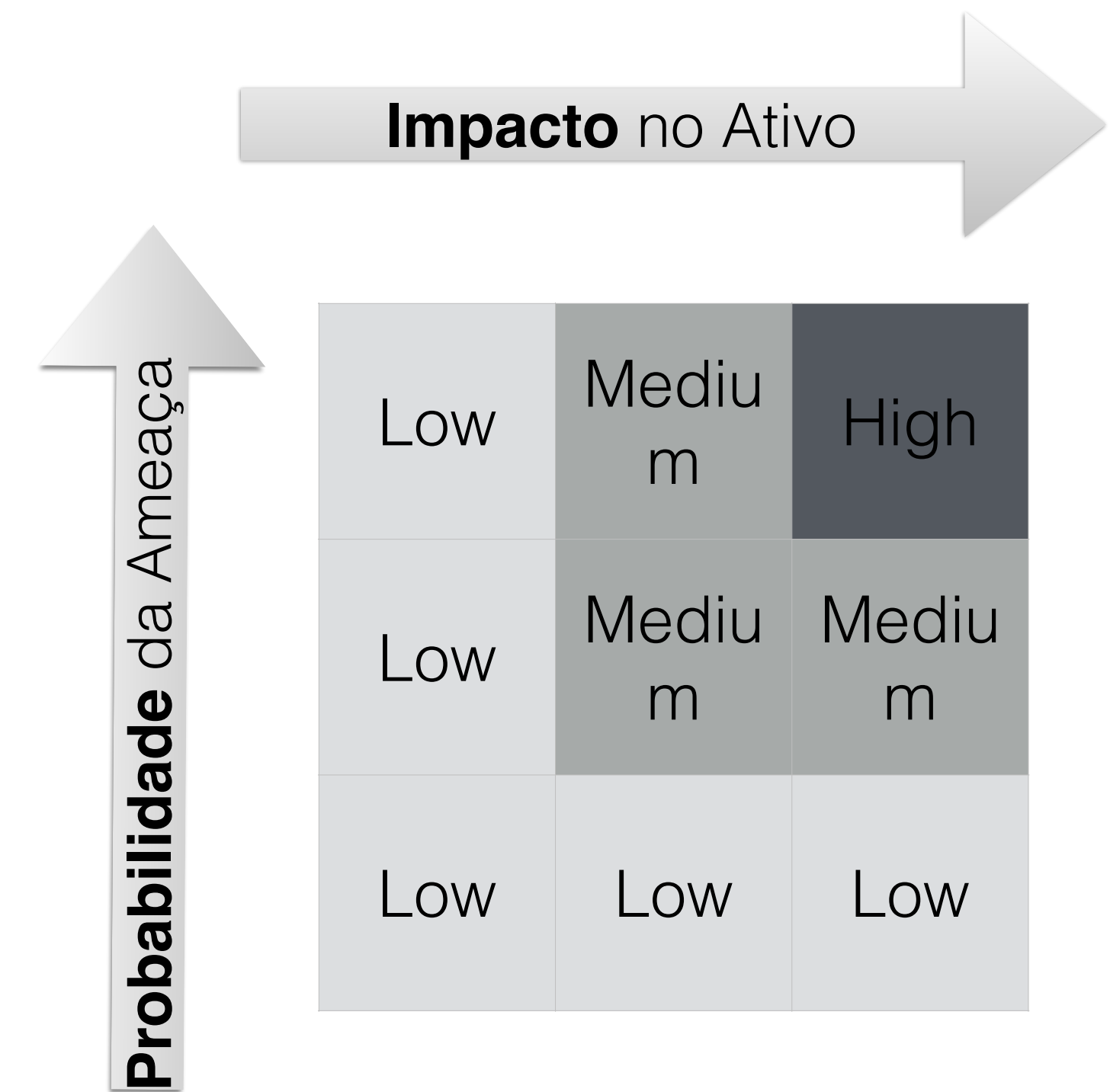
- **Gerimos o risco** de que ativos (recursos) relevantes sejam usados indevidamente ou estejam indisponíveis.
- Um **ativo** é um recurso que detém valor para um ator do sistema:
 - informação
 - reputação/imagem
 - dinheiro/recurso com valor monetário intrínseco
 - infra-estrutura
 - etc.

Gestão de Risco: Ativos

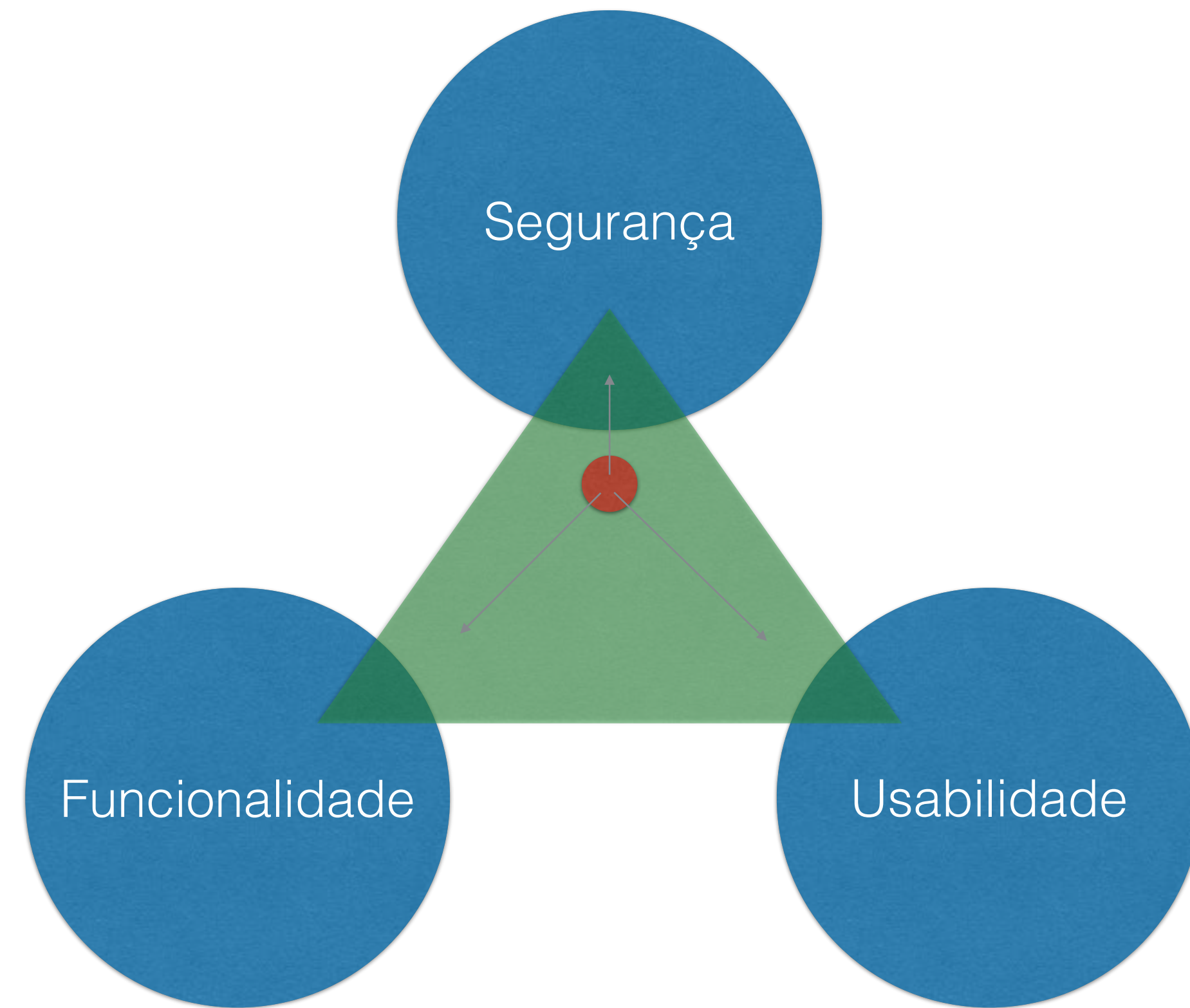
- O risco que se pretende acautelar é o de que um ativo perca o seu valor durante a utilização do sistema
- CIA \Rightarrow Risco de **perda de valor** por quebra de:
 - Confidencialidade (acessos não autorizados)
 - Integridade (modificações não autorizadas)
 - Disponibilidade (quebras de serviço)

Matriz de análise de risco

- 2 eixos: **ativos** + **ameaças**
- A acção a tomar depende de dois fatores críticos:
 - o potencial impacto no ativo
 - a probabilidade de materialização da ameaça
 - cada par (ameaça,ativo) tem de ser avaliado nestes dois eixos
 - a decisão de mitigação depende do risco avaliado
- Processo de análise, e.g., metodologia STRIDE



Compromissos: Um Triângulo

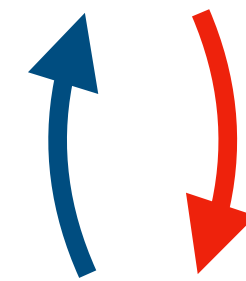


Trabalhar em Ciber-Segurança



Paranoia

Racionalidade



Mais terminologia ...

Vulnerabilidades

- Uma **vulnerabilidade** é uma falha que está acessível a um adversário que poderá ter a capacidade de a explorar
- As vulnerabilidades têm geralmente origem em erros de concepção:
 - Software de má qualidade
 - Análise de requisitos desadequada
 - Configurações erradas
 - Utilização errada

Vulnerabilidade



[source: Commandos game series]

Ataques

- Um **ataque** ocorre quando alguém tenta *explorar* uma vulnerabilidade
- Tipos de ataques:
 - Passivos (e.g., eavesdropping)
 - Activos (e.g., adivinhar passwords)
 - Denial-of-Service
- Quando um ataque é bem sucedido diz-se que um sistema foi *comprometido*



[source: Commandos game series]

Estrutura de um ataque

- Um ataque precisa de MOM:
 - **Motivo/Ameaça:** perturbação, roubo de informação, ...
 - **Oportunidade/Vulnerabilidade:** algo que pode ser explorado
 - **Método/Exploit:** forma de explorar a vulnerabilidade



Reporting

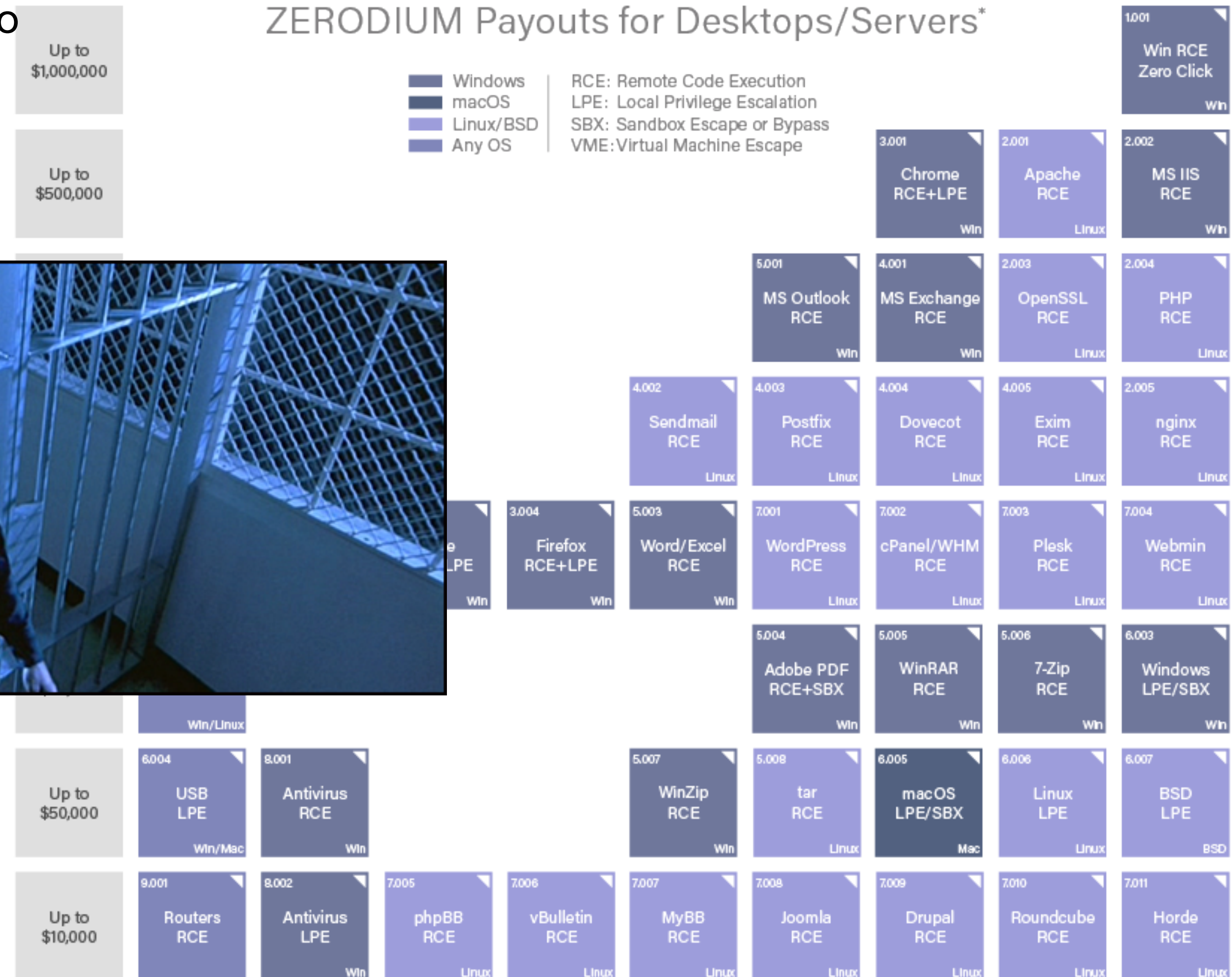
- Novas vulnerabilidades aparecem todos os dias
- A gestão desses processos chama-se vulnerability reporting
- Toda a comunidade trabalha em conjunto para:
 - Identificar, classificar, divulgar, detectar, eliminar
- Ver por exemplo:
 - CWE (cwe.mitre.org), CERT (<http://www.kb.cert.org/vuls/>)
- Um utilizador pode:
 - ser notificado, obter informação, obter uma ferramenta que verifica se a vulnerabilidade existe

Esta discussão explica o mercado para Zero-Day Vulnerabilities:

- Nunca foram reportadas
- Estão latentes
- São super-poderes



[fonte: Terminator 2]



* All payouts are subject to change or cancellation without notice. All trademarks are the property of their respective owners.

2019/01 © zerodium.com

Ameaças

- Uma **ameaça** é uma causa possível para um incidente que possa trazer consequências negativas para um sistema, pessoa ou organização
- Dependem do contexto:
 - Podem ser pessoas
 - Podem ser eventos naturais
 - Podem ser falhas acidentais
 - Podem ser causadas intencionalmente

Ameaças

- Geralmente definem-se pelo seu tipo e origem:
 - Tipo de ataque: dano físico, perda de serviços, quebra de protecção de informação, falhas técnicas, abuso de funcionalidades
 - Tipo de atacante: acção deliberada (implica definir a fonte), acção negligente (implica definir a fonte), acidente (implica definir o componente afectado), evento ambiental.
- Dependendo da comunidade (e.g., finança vs distribuição de água)
 - As ameaças estão identificadas e classificadas quanto à relevância



[source: Commandos game series]

Modelo de Ameaças

- Objetivos de segurança: ativos, o que queremos proteger e de quem?
- Quem são os nossos adversários?
 - Motivação
 - Capacidades
 - Tipo de Acesso
- Que tipos de ataque temos de precaver (pensar como um atacante)
- Âmbito: que tipos de ataque podemos descartar/ignorar?

Modelo de Ameaças

- Perímetro de segurança
 - fronteira que delimita um contexto com o mesmo nível de segurança
 - quaisquer inputs provenientes do exterior são suspeitos
- Superfície de ataque
 - pontos/formas de contacto com o exterior no perímetro de segurança

Mecanismo de segurança

- Um **mecanismo de segurança** é um método, ferramenta ou procedimento que permite implantar uma (parte de uma) política de segurança
- Os mecanismos podem ser não técnicos, e.g., exigir identificação pessoal na entrada de um edifício
- Parte do nosso modelo de confiança consiste em acreditar que um (conjunto de) mecanismo(s) de segurança cumpre a sua função, que está correctamente instalado e administrado, etc.

Exemplos

- Mecanismos de identificação/autenticação (e.g., biometria, one-time passwords)
- Mecanismos de controlo de acessos (e.g., RBAC)
- Criptografia (e.g., cifras, MACs, assinaturas)
- Controlos físicos (e.g., cofres, torniquetes)
- Auditorias (e.g., penetration testing)

Política de segurança

- Uma política de segurança determina:
 - um conjunto de processos/mecanismos que devem ser seguidos
 - para garantir segurança num determinado modelo de ameaças
- Uma política de segurança pode ter como objectivos:
 - prevenção, detecção e/ou recuperação

Exemplos

- Segurança física
- Controlo de acessos
- Política de passwords
- Política de email
- Política de acesso remoto
- ...

Gerir a Complexidade

- Em sistemas complexos o problema da segurança tem de ser tratado de forma estruturada
 - Segurança de redes
 - Segurança de sistemas/computadores
 - Segurança de programas/software
 - Segurança física
 - Privacidade
 - ...

Confiabilidade

- É essencial que a "segurança" não seja um “salto de fé”
- Engenharia de segurança:
 - construir confiança através de argumentos rigorosos

Confiabilidade I

- Definir o problema:
 - Análise de requisitos de segurança
- Definir o modelo de ameaças
 - Ativos e objetivos de segurança
 - Classes relevantes de ameaças (o adversário)

Confiabilidade II

- Definir o modelo de confiança
 - em que componentes/atores confiamos
 - para fazer o quê (o que nos é dado como ponto de partida/âncora)
- Definir a solução:
 - Conceber as políticas de segurança que se aplicam ao sistema
 - Identificar os mecanismos de segurança necessários e suficientes

Confiabilidade III

- Validar e justificar a solução:
 - Validar a adequação dos modelos à aplicação concreta em análise
 - (o adversário não é obrigado a seguir o nosso modelo de ameaças)
- Demonstrar (formal ou informalmente) que
 - os mecanismos de segurança subjacentes são suficientes,
 - em conjunto com os pressupostos de confiança assumidos,
 - para implantar as políticas de segurança (justificação)

**A auditoria de segurança visa
validar todo este processo**

Testes de penetração: ethical hacking

- Método de avaliação do nível de segurança de um sistema
- Simulação de um ataque de procura de vulnerabilidades que poderiam ser exploradas
- Black box: semelhante a um ataque real
- White box: com conhecimento privilegiado
- O valor está no relatório e recomendações finais

Acknowledgements

- This lecture's slides have been inspired by the following lectures:
 - CSE127: Introduction + Security Foundations