

Formação com Impacto

Liderar a Cibersegurança, transformar a Organização



Intellectual Property Rights

- This document was produced by NOVA FCT Executive Education or, acting on its behalf, by a faculty member, teacher, as part of a course, program, postgraduate program, or other initiative promoted by this school.
- The content of this document is the property of NOVA FCT Executive Education and its authors, where identified.
- The sharing of this document does not transfer any ownership or usage rights beyond the purpose for which it is intended.
- The sole purpose of this document is to support the learning of the topics taught in one or more sessions of a course, program, postgraduate programme, or other initiative in which the participant has been involved.
- For all purposes, the publication, duplication, sharing, distribution, or any other form of disclosure of this document is not authorized beyond its originally intended purpose, whether in a personal or professional context. This limitation also applies to the use of the document, which must respect its original purpose.

Propriedade Intelectual

- O presente documento foi **produzido** pela NOVA FCT Executive Education, ou, atuando em sua representação, um docente, colaborador ou parceiro, integrado num curso, programa, Pós-graduação ou outra ação promovida por esta escola.
- O conteúdo deste documento é **propriedade** da NOVA FCT Executive Education e dos seus autores, caso se encontrem identificados.
- A **partilha** do presente documento não transfere o direito de propriedade ou utilização, para além da finalidade a que se destina.
- A **finalidade única** deste documento é servir de apoio à aprendizagem dos tópicos lecionados numa ou em várias sessões de um curso, programa, Pós-graduação ou outra ação em que o participante tenha estado envolvido.
- Para todos os efeitos, **não está autorizada** a publicação, duplicação, partilha, distribuição ou qualquer outra forma de divulgação deste documento, para além da que lhe foi inicialmente destinada, seja em contexto pessoal ou profissional. A limitação estende-se à utilização do documento, respeitando a finalidade original.

Estrutura do programa

O programa tem a **duração total de 16 horas**, distribuídas por 4 tópicos.

Considerando a extensão e complexidade técnica dos conteúdos solicitados, bem como a carga horária disponível, a formação adotará uma abordagem introdutória e generalista, privilegiando o enquadramento conceptual, os princípios fundamentais e as boas práticas, não contemplando um aprofundamento técnico exaustivo de cada tema.

- 01** Segurança de Redes
- 02** Proteção e Privacidade de Dados
- 03** Princípios Técnicos em Cibersegurança
- 04** Formação Intermédia em Cibersegurança



Sessão 03: Princípios Técnicos de Cibersegurança

Formador: Henrique Domingos



PhD in Computer Science

Coordenador de formações e pós-graduações na áreas da Ciber-segurança e Ciber-resiliência, na Escola de Executivos da FCT/UNL

Professor na Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

Atividade letiva principal e extensões académicas no Departamento de Informática
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

Responsável pelas Unidades Curriculares de Segurança de Sistemas de Computadores. Privacidade de Sistemas e Segurança Confiabilidade de Sistemas na

- Licenciatura em Engenharia Informática
- Mestrado em Engenharia Informática
- Programa Doutoral de Engenharia Informática

Investigador principal integrado no Centro de Investigação NOVA-LINCS
NOVA Laboratory on Informatics and Computer Science



NOVA EXECUTIVE
EDUCATION
NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

5

Conteúdo Geral da Sessão 03 como solicitado

Princípios Técnicos de Cibersegurança

1. Fundamentos de Segurança Informática: Princípios base de cibersegurança; Superfície de ataque e vetores de entrada Ameaças, Ataques e Vulnerabilidades: Classes de ameaças e tipos de ataque; Identificação e exploração de vulnerabilidades

2. Servidores e Suporte de Comunicação Cliente-Servidor:

- Arquitetura cliente-servidor; Segurança em servidores; Gestão de serviços; Riscos e Níveis de exposição em superfícies de ataque; Contramedidas de Defesa e Tecnologias Relevantes
- Princípios de desenvolvimento seguro de serviços

3. Criptografia: Conceitos base de criptografia; Encriptação simétrica e assimétrica; Aplicações práticas na proteção de dados e visão do avanço da criptografia aplicada convencional para métodos criptográficos pós-quânticos

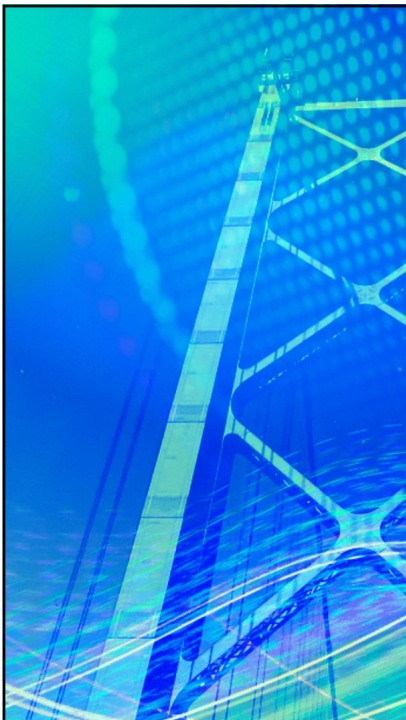
4. Segurança de Software e Bases de Dados: Princípios de desenvolvimento seguro; Segurança e controlo de acessos em bases de dados; Proteção de dados e integridade da informação

NOVA EXECUTIVE
EDUCATION
NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

6

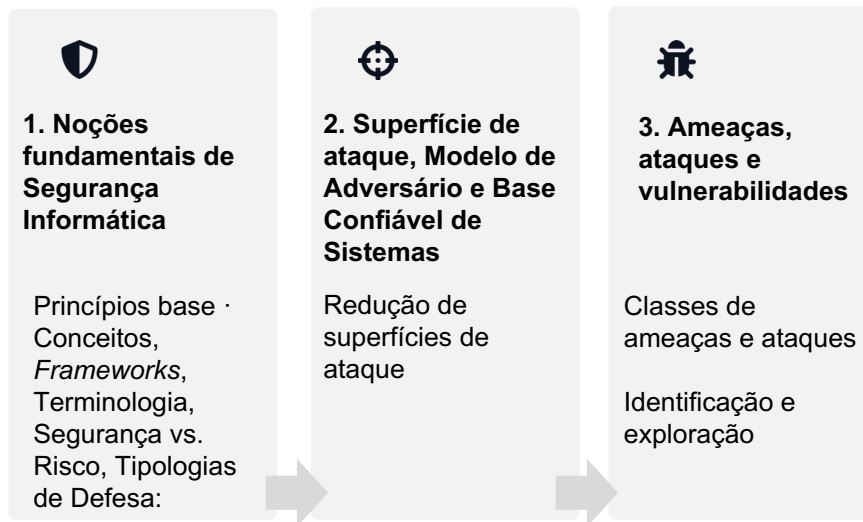
Agenda da Sessão: Blocos principais

- 1. Fundamentos de Segurança Informática**
- 2. Servidores e Suporte de Comunicação Cliente-Servidor**
- 3. Criptografia**
- 4. Segurança de Software e Bases de Dados**



1. Fundamentos de Segurança Informática

Tópicos neste bloco



Noções Fundamentais de Segurança Informática

Cibersegurança

Proteger sistemas, redes e dados contra acesso, uso, alteração ou destruição não autorizados, mantendo as organizações a funcionar corretamente e assegurando os objetivos previstos, com controlo do risco operacional controlado e assegurando o funcionamento com resiliência

Funções na cibersegurança organizacional

Ref. Ex. 6 funções do NIST CSF 2.0



Governar

Estratégia, papéis e políticas



Identificar

Ativos, riscos e dependências



Proteger

Controlos que evitam incidentes



Detetar

Identificação de ocorrência de eventos anómalos



Responder

Conter e tratar o incidente



Recuperar

Restaurar e aprender a operação correta

NIST Cybersecurity Framework (CSF) 2.0,
National Institute of Standards and Technology
<https://www.nist.gov/cyberframework/informative-references>

Ciber-segurança vs. Ciber-resiliência

Muito relevante hoje na visão conjunta de objetivos, como “duas faces de uma mesma moeda”

Ciber-resiliência (*Cyber Resilience*)

- Definir um plano integrado para ciber-resiliência, englobando as dimensões da Ciber-Segurança
- Manter a operação correta e garantir como recuperar perante eventuais ataques ou falhas
- Continuar a prestar serviços (*Organization ou Business Continuity*)
- Assegurar a recuperação de dados ou estado de sistemas após um incidente.
- Garantir cópias de segurança e definição clara de planos de contingência para recuperação dos sistemas no seu funcionamento correto e dados associados ao processamento de informação
- Aprender com eventuais falhas ou ataques e planear como melhorar depois de possíveis incidentes

Ciber-segurança (*Cybersecurity*)

- Proteger os sistemas e prevenir ataques
- Proteger os sistemas de possíveis falhas (sejam falhas por paragem, sejam falhas arbitrárias)
- Prevenir acessos não autorizados.
- Implementar os controlos de segurança a vários níveis de operação e utilização de sistemas (aos diversos níveis das pilhas de software de suporte) bem como das redes que os interligam)

Exemplo de uma visão integrada de ciber-segurança e ciber-resiliência (Ex: Prevenção de ataques de *ransomware*)

Ciber-Resiliência

A organização pretende manter os serviços essenciais em funcionamento

No seu plano de ciber-resiliência e gestão de risco, está (entre outros vetores de ataque ou falhas) a previsão de possíveis ataques de *ransomware*

Ciber-Segurança

Ex:

Prevenção:
Firewalls, Anti-Vírus, Isolamento (para ZERO TRUST) e Contenção e Bloqueio do Ataque

Deteção: colocou em operação Meios e Sistemas de Deteção de Intrusões (Ex: Plataformas de Gestão e Controlo de Evento Anómalos)

Recuperação: estabeleceu meios de recuperação ou de tolerância ao ataque, garantindo tolerância ou máxima disponibilidade após recuperação

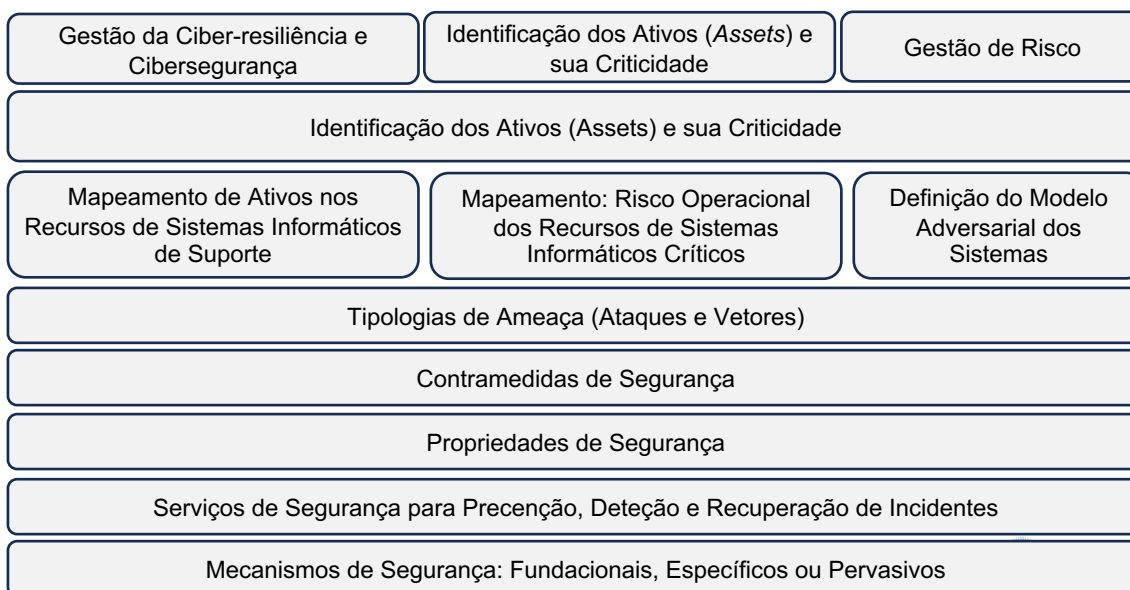
Face a um incidente

Definiu a estratégia comunicacional e vai proceder a uma análise crítica e detalhada do incidente e vai planear como melhorar a resiliência

Procedeu-se a uma análise técnica “forense” da “anatomia” do ataque

Reporte e documentação do incidente

Uma visão estruturante multi-nível



13

Níveis e responsabilidades organizacionais

Nível 1: Gestão estratégica ou de governança: **Nível estratégico**

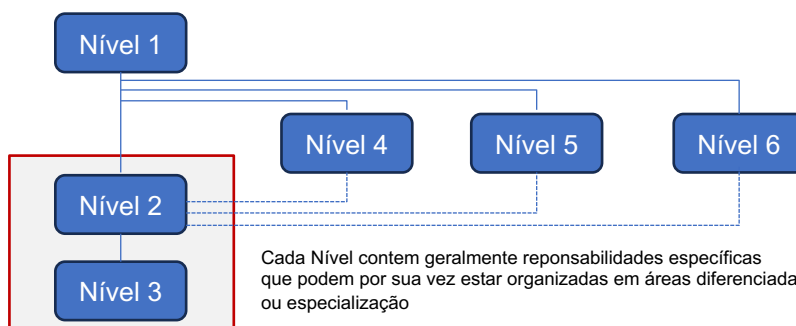
Nível 2: Gestão e Coordenação da Ciber-segurança: **Nível CISO**

Nível 3: Gestão Operacional e Técnica: **Nível Operacional, Equipas de Ciber-segurança**

Nível 4: Gestão de **outras Áreas de Responsabilidade ou de Gestão de Negócio**

Nível 5: Gestão de **Colaboradores** (Pessoas, Utilizadores)

Nível 6: Gestão de **Terceiros** (Fornecedores, Parceiros)

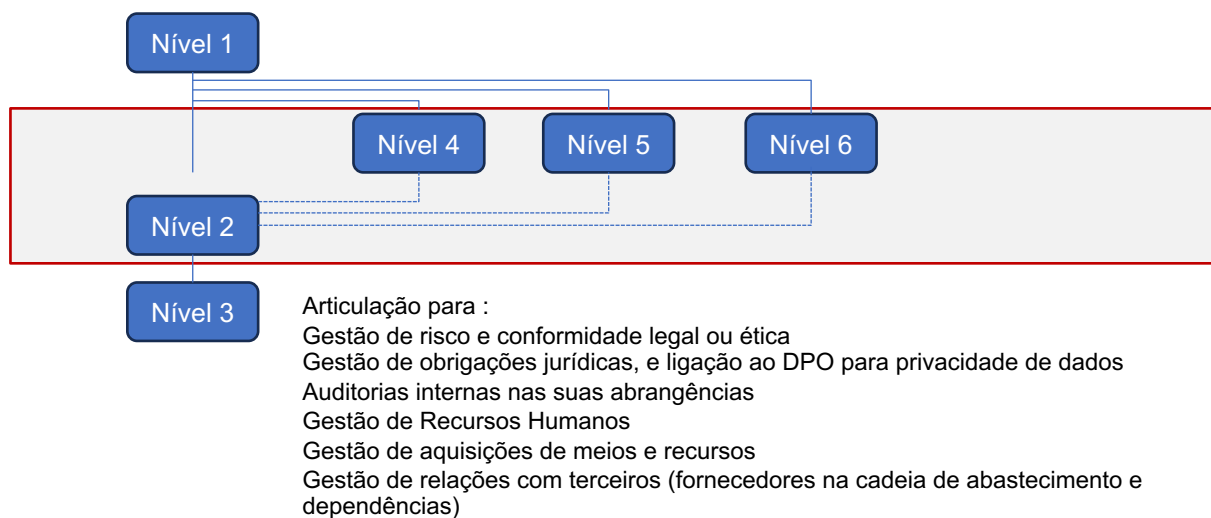


Abordagem de cada nível nos próximos slides

Cada Nível contém geralmente responsabilidades específicas que podem por sua vez estar organizadas em áreas diferenciadas de atuação ou especialização

14

Exemplo de organização: Nível 3



15

Exemplo de organização: Nível 3



16

Responsabilidades na visão estruturante multinível

Nível 1: Gestão estratégica ou de governança (em estreita articulação com os níveis adequados de topo na gestão da organização)

- Conselho de Administração, Comissão executiva e CEO
- Define estratégia, aprova o apetite ao risco e assegura os recursos

Nível 2: Gestão e Coordenação da Cibersegurança

- CISO (*Chief Information Security Officer*) – Responsável da Cibersegurança na Organização
- Define e coordena o plano de segurança, políticas, riscos e indicadores (ou métricas) de verificabilidade e conformidade
- Articula com os níveis adequados (diversos setores) na organização (ex., Recursos Humanos, Logística / Abastecimentos, Área Financeira, Área da Comunicação, etc...). Assegura a articulação direta ou indireta na ligação com outros Níveis: Níveis 3, 4, 5 e 6
- Assegura a ligação ao suporte de segurança para requisitos legais (ex., DPO e RGPD ou outros quadros específicos legais ou éticos para funcionamento da organização)

Responsabilidades na visão estruturante multinível

Nível 3: Gestão Operacional e Técnica

- Gestão ao nível IT e Segurança Operacional
- Equipas SOC (*Security Operation Center*)
- Gestão de Infraestruturas, Sistemas, Postos de Trabalho, Servidores, Redes, Aplicações e sua Operação
- Implementa controlos, monitoriza ameaças e responde a incidentes e reporta os mesmos
- Verifica e avalia continuamente o estado de segurança dos sistemas, gerindo a segurança como um processo
- Teste, operacionalização, colocação em produção e acompanhamento da produção de sistemas e soluções, com análise de conformidade da segurança exigida (com cumprimento e implantação da normalização técnica adequada)
- Pode envolver controlo de gestão de projetos de desenvolvimento de sistemas (que podem ser desenvolvidos em quadros de parcerias para desenvolvimento)

Responsabilidades na visão estruturante multinível

Nível 4: Plano da Gestão Operacional e Técnica

Diretores, gestores de processo, responsáveis pelos sistemas e pelos dados associados
Gerem os riscos das suas áreas, articulam a proteção dos respetivos ativos no cumprimento dos requisitos específicos sectoriais de segurança

Por exemplo, um DPO poderia articular-se como uma das entidades relacionadas no nível 4

Nível 5: Plano da gestão de colaboradores e acompanhamento das suas operações

- Cumprimento das regras, promoção de boas práticas, proteção de credenciais, formação, comunicação de situações suspeitas
- Envolvimento e promoção da adesão dos colaboradores na importância do fator humano

Nível 6: Plano da gestão entidades terceiras

- Cumprimento de regras, promoção de boas práticas nos aspetos que envolvem negociação, aquisição e conformidade de soluções, serviços ou consultadoria de terceiros
- Envolvimento na interlocução para estabelecimento de condições de conformidade, legalidade ou lealdade de parceiros ou fornecedores na cadeia de fornecimento

VE
ION

Mapeamento de Responsabilidades em matriz RACI

RACI:

Matriz RACI ajuda a definir quem é responsável por ações/operações, quem responde pelo resultado, quem deve ser consultado e quem tem que ser informado.

- | | |
|------------------------|--|
| R — <i>Responsible</i> | : responsabilidade de execução |
| A — <i>Accountable</i> | : responde pelo resultado e pela decisão final |
| C — <i>Consulted</i> | : deve ser consultado |
| I — <i>Informed</i> | : deve ser informado no conhecimento |

Exemplo de Responsabilidades em matriz RACI

Exemplo de mapeamento RACI:

	Nível 1	Nível 2	Nível 3	Articulação Níveis 2, 4, 5 e 6	Articulação Níveis 2, 1 e Gestão de Risco
Atividade	Administração	CISO	IT / Segurança	Negócio	Risco / Auditoria
Aprovar estratégia de cibersegurança	A	R	C	C	C
Definir políticas de segurança	I	A/R	C	C	C
Avaliar e tratar riscos técnicos	I	C	R	A*	C
Implementar controlos técnicos	I	C	A/R	C	I
Gerir riscos dos processos de negócio	I	C	C	A/R	C
Responder a incidentes	I	A	R	C	C
Supervisionar e auditar controlos	I	C	C	C	A/R**

21

Visão estruturante multinível

Estabelecimento dos princípios fundamentais e requisitos funcionais da Cibersegurança

Nível da Gestão Estratégica / Governança

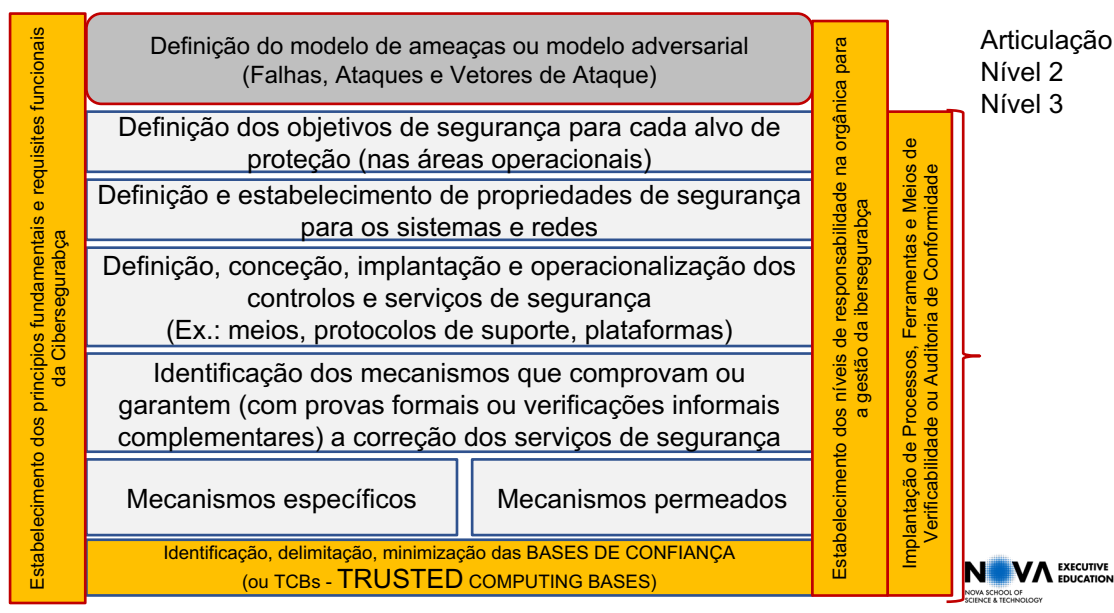
- Definição da política e plano geral de segurança (e cibersegurança)
- Deve identificar os ativos e a criticidade dos mesmos
- Definição do modelo de gestão e avaliação de risco
- Estabelece indicadores de desempenho e risco
- Definição formal de responsabilidades orgânicas e papéis
- Pode estabelecer comités de ciber-segurança (por exemplo, atualmente incorporando as preocupações com a IA)
- Estabelece as premissas para gestão de incidentes e continuidade
- Deve assegurar meios de formação e sensibilização continua aos diversos níveis da organização
- Deve assegurar a adequação de meios
- Os planos devem prever na gestão de Ativos
 - O Mapeamento em recursos da base IT
 - Recursos: sistemas computacionais e redes de suporte

Definição do modelo de ameaças ou modelo adversarial
(Falhas, Ataques e Vetores de Ataque)

Estabelecimento dos níveis de responsabilidade na orgânica para a gestão da cibersegurança

Articulação
Níveis 1 e 2
Níveis 2-4-5-6

22



23

Como abordar conceptualmente e de forma sistemática a abordagem nos diferentes níveis ?

Relevância na utilização de terminologia e domínio de bases conceptuais de referência

Fontes importantes: Quadros de Referência para Cibersegurança (ou *Security frameworks*)

O que são?

Bases documentais com conjunto de diretrizes, boas práticas e padrões que ajudam as organizações a avaliar, desenvolver e melhorar as suas estratégias de cibersegurança cibernética. Esta documentação pode estar associada quer a quadros regulatórios/legais, quer a quadros recomendatórios/informacionais, podendo estes conter estruturas flexíveis e adaptáveis para aplicação sistemática a uma ampla gama de setores e tipos de organizações, independente do tamanho, complexidade ou especialidade de atuação.

Podem estar mais vocacionados para os diferentes níveis de abordagem (na visão multinível anterior)

24

Que quadros de referência (*frameworks*) são relevantes?

Que exemplos notáveis são relevantes ?

- Qual a sua orientação (ou público-alvo) ?
- Têm impacto em aspetos de certificação ?
- Têm impacto em aspetos de conformidade legal ?
- Como articular os diferentes níveis de orientação e enfoque?

Noções Fundamentais de Segurança Informática

Frameworks para cibersegurança

EUA: NIST CSF 2.0

UE: Diretiva NIS2 (2022/2555)

Em Portugal: Decreto-Lei 125/2025

EU GDPR: Regulamento EU 2016/679

PT RGPD: Lei n.º 58/2019, de 8/AGO

Padrões ISO/IEC 27000

OSI X.800

IETF RFC 4949 e. RFCs de normas

“de facto” (ex., Segurança na pilha

TCP/IP, Segurança Internet)

FIPS PUB e Standards tecnológicos

Frameworks: Referências

EUA: NIST CSF 2.0	https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf
UE: Diretiva NIS2 (2022/2555) Em Portugal: Decreto-Lei 125/2025	https://eur-lex.europa.eu/legal-content/PT/TXT/?uri=CELEX:32022L2555 https://diariodarepublica.pt/dr/detalhe/lei/59-2025-941547424
EU GDPR: Regulamento EU 2016/679 PT RGPD: Lei n.º 58/2019, de 8/AGO	https://eur-lex.europa.eu/PT/legal-content/summary/general-data-protection-regulation-gdpr.html https://diariodarepublica.pt/dr/detalhe/lei/58-2019-123815982
Padrões ISO/IEC 27000	https://www.iso.org/standard/iso-iec-27000-family
OSI X.800	https://www.itu.int/rec/T-REC-X.800-199103-1
IETF RFC 4949 e. RFCs de normas “de facto” (ex., Segurança na pilha TCP/IP, Segurança Internet) FIPS PUB e Standards tecnológicos	https://datatracker.ietf.org/doc/html/rfc4949 , https://www.ietf.org/technologies/security/ https://csrc.nist.gov/publications/fips

Frameworks e sua orientação

EUA: NIST CSF 2.0	Base conceptual e de terminologia comum para redução de “gaps” entre equipas técnicas e os níveis de gestão, abrangendo o cumprimento de boas práticas e metas orientadas a resultados
UE: Diretiva NIS2 (2022/2555) Em Portugal: Decreto-Lei 125/2025	Importante no quadro regulatório na Europa, obrigando à conformidade através de sanções que podem ser severas: alvo para 18 setores críticos - Ex., energia, Bancam Saúde, Transportes, Infraestruturas de mercados financeiros, Água-Saneamento, Infraestruturas e provedores de suporte de digitalização de serviços (Redes, provedores Cloud, Provedores de comunicações, Prestadores TIC de serviços de segurança, Administração Pública, Indústria e Serviços Aero-Espaciais
EU GDPR: Regulamento EU 2016/679 PT RGPD: Lei n.º 58/2019, de 8/AGO	Base legal para uniformizar a proteção das pessoas singulares e tratamento de dados pessoais e asseguramento de livre circulação desses dados, garantindo aos cidadãos o controlo sobre esses dados e o controlo consentido para finalidade, uso e/ou contexto de retenção Complementaridades: LEI 59/2019 regula o tratamento de dados pessoais para efeitos de prevenção, deteção, investigação ou repressão de infrações penais
Padrões ISO/IEC 27000	É uma estrutura documental internacional para estabelecer boas práticas de conformidade e implementar, manter e melhorar continuamente os sistemas de Gestão de Segurança da Informação (SGSI) Ref. https://www.iso.org/standard/iso-iec-27000-family
OSI X.800	<i>Security Architecture for Open Systems Interconnection</i> : padrão desenvolvido pela ITU-T (União Internacional de Telecomunicações) que define propriedades, mecanismos e serviços numa arquitetura de segurança para o Modelo OSI de redes de computadores
IETF RFC 4949 e. RFCs de normas “de facto” (ex., Segurança na pilha TCP/IP, Segurança Internet) FIPS PUB e Standards tecnológicos	Padrões de normalização técnica de produtos e tecnologias de SW e HW

Frameworks: Padrões ISO/IEC 27000*

ISO/IEC 27000: Visão geral, conceitos fundamentais e o vocabulário de toda a família de normas

ISO/IEC 27001: Padrão central, sendo a **única normalização passível de certificação na prática de adoção**. Define os requisitos obrigatórios para planear, implementar, manter e melhorar continuamente um Sistema de Gestão de Segurança da Informação (SGSI)

ISO/IEC 27002: Padrão que estrutura um código de boas práticas e um conjunto de controlos de segurança da informação (estruturados para apoiar a ISO 27001)

ISO/IEC 27003: Orientações para boas práticas visando a implementação de um SGSI. [1]

ISO/IEC 27004: Orientações para monitorar, medir, analisar e avaliar/auditor o desempenho e conformidade de eficácia dos SGSI

ISO/IEC 27005: Padrão que aborda a **gestão de riscos de segurança da informação**, para orientação de linhas-guia para processo de identificação, análise e tratamento de riscos alinhado aos requisitos da ISO 27001.

* <https://www.iso.org/standard/iso-iec-27000-family>

Frameworks na Cibersegurança Organizacional

ALVO DE CONFORMIDADE:

Nível da Organização:
Gestão e Comunicação

EUA: NIST CSF 2.0

UE: Diretiva NIS2 (2022/2555)
RGPD

> Não Certificável

> Cumprimento Legal

> Cumprimento Legal

Instância-ALVO de CONFORMIDADE:

Nível da:
Gestão de Cibersegurança
Organizacional, Gestão IT

ISO 27000 Series

> Alinhamento Certificável

Instância-ALVO DE CONHECIMENTO:

Gestão IT, Engenharia,
Administração de
Sistemas e redes

OSI X.800

> Não Certificável

Instância ALVO de CONFORMIDADE:

Aquisição, Desenvolvimento de
Tecnologias, Base IT: SW/HW

IETF RFC 4949 e RFCs de normas
"de facto" (ex., Segurança na pilha
TCP/IP, Segurança Internet
FIPS PUB e Standards tecnológicos

> Certificável e verificável

**Alinhamento prático
exemplificativo de
diferentes
“frameworks” para
uma abordagem
multinível**

Critério / Aspeto	NIST CSF 2.0	Diretiva NIS2	ISO/IEC 27001	OSI X.800	FIPS PUB (ex: 140-2/3)
Tipo de Documento	Framework de resultados (guias e boas práticas).	Diretiva Legal / Regulamento da União Europeia.	Norma Técnica Internacional (Standard ISO/IEC).	Norma de Arquitetura de Redes (ITU-T / ISO).	Padrões Federais de Processamento de Informação (EUA).
Natureza Legal	Voluntária (aplicável globalmente a qualquer setor).	Obrigatória (entidades essenciais e importantes na UE).	Voluntária (adotada por decisão comercial/contratual).	Voluntária (referência técnica de engenharia).	Obrigatória para agências federais dos EUA e contratantes.
Foco Principal	Gestão e comunicação do risco de cibersegurança.	Resiliência cibernética nacional e de infraestruturas críticas.	Implementação de um Sistema de Gestão (SGSI/ISMS).	Segurança em comunicações e interconexão de sistemas (OSI).	Requisitos de segurança para módulos criptográficos e algoritmos.
Estrutura Core	6 Funções: Governar, Identificar, Proteger, Detetar, Responder e Recuperar.	Obrigações de governança, gestão de risco e reporte de incidentes.	Cláusulas de gestão (4 a 10) e 93 controlos de segurança (Anexo A).	Serviços de segurança (autenticação, integridade) e Mecanismos.	Níveis de segurança qualitativos (Nível 1 ao Nível 4).

**Ativos representados
no suporte de SISTEMAS COMPUTACIONAIS SEGUROS**

Como definir um Sistema Seguro ?

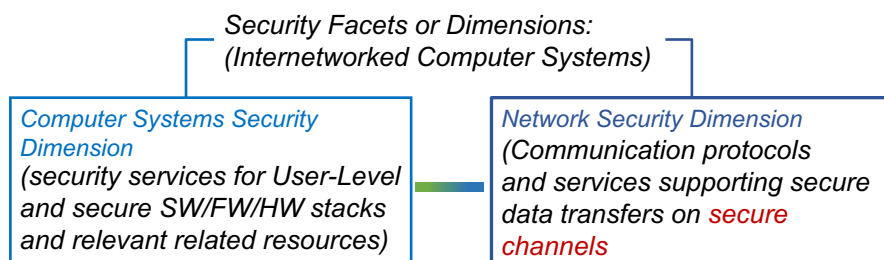
Como definir um Sistema Seguro ?

Um sistema concebido, implementado, operado e mantido de forma a proteger os seus ativos (recursos), assegurar o cumprimento dos requisitos de segurança para operação correta mantendo o risco de incorreção em níveis aceitáveis, mesmo perante falhas, erros, ameaças ou ataques

Um sistema computacional seguro é pois um sistema concebido, desenvolvido, configurado, operado e mantido de forma a proteger os seus recursos (processamento e comunicações) contra ações não autorizadas, preservando propriedades de segurança que asseguram contramedidas contra potenciais ataques definidos num modelo de adversário que esteve subjacente à sua conceção assegurando que apenas são executadas funções e operações autorizadas e gerir os riscos de segurança ao longo de todo o seu ciclo de vida.

Outra definição de “Sistema Seguro” (Def. a partir do NIST, NISTIR ref.

*A system with protections designed to achieve applicable **security objectives** by ensuring the **security properties** of its resources, including **data processing, software, firmware, hardware, and communications**, with protection provided through appropriate **security services** based on **security mechanisms** and **trusted computing bases**, which act as countermeasures to prevent threats and defend against types of attacks, within a **well-defined threat or adversary model**.*



Uma “equação” interessante

Risco de ataque = Vulnerabilidades x Potencial de Ameaça

Esta equação evolui com o tempo:

Risco (t) = Vulnerabilidades (t) x Potencial de Ameaça (t)

Uma “equação” interessante

Risco de ataque = Vulnerabilidades x Potencial de Ameaça

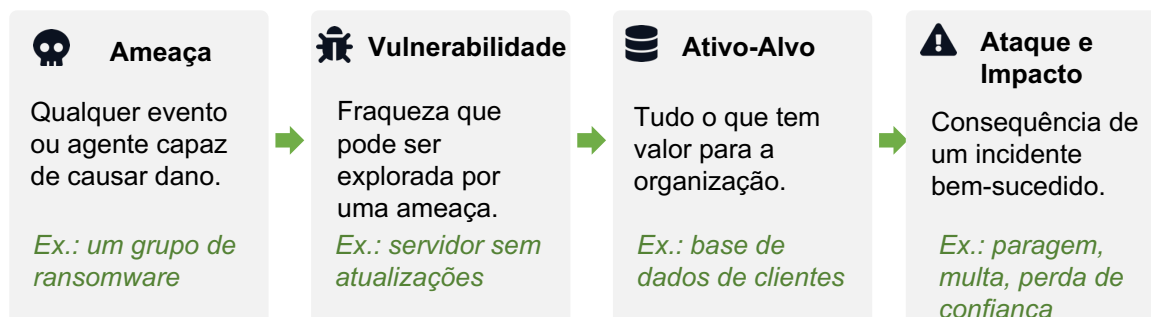
Esta equação evolui com o tempo:

Risco (t) = Vulnerabilidades (t) x Potencial de Ameaça (t)

Aspetos interessantes a considerar

- A definição do modelo de adversário que esteve na base de conceção de um sistema seguro pode vir a tornar-se obsoleto com o tempo. Porquê ?
- As bases de confiança podem deixar de o ser. Porquê ?
- Como se lida com a implicação da degradação progressiva dos sistemas com o passar do tempo?
- Redesenho para uma nova base de confiança. Possível ? Fácil ?
- Com base na revisão e atualização do modelo de adversário (é possível a reimplementação ou disponibilização de corretivos em componentes específicos afetados ?
- *Patching* (manutenção corretiva) ou Descontinuação do ciclo de vida de operação ?

Risco de Ataque



Risco = Probabilidade × Impacto

Os controlos de segurança reduzem a probabilidade, o impacto, ou ambos.

Aspetos interessantes na definição de “Sistema Seguro”

Ou sejam um sistema computacional seguro não é necessariamente um sistema sem vulnerabilidades, infinitamente seguro; é um sistema cuja segurança é concebida e gerida de forma sistemática, cujos comportamentos e recursos (computacionais e de comunicação) são controlados e cujos riscos são conhecidos e considerados aceitáveis, sendo concebidos a partir da definição:

- De um **modelo de adversário** que esteve na base da sua conceção e implementação correta
- De **propriedades de segurança** que concretiza face ao modelo de adversário definido
- Da **tipologia de ataques** subjacentes ao modelo de adversário e propriedades de segurança que concretiza
- Dos **controlos (ou serviços) de segurança** que garantem essas propriedades
- Dos **mecanismos de segurança** subjacentes aos controlos
- Da **base de confiança** da qual depende a sua conceção e operação

Objetivos e propriedades de segurança

A tríade CIA (*Confidentiality, Integrity, Availability*)



C

Confidencialidade

Só quem está autorizado acede à informação.

CONTROLOS

Cifra, controlo de acessos, classificação de dados

ATAQUES TÍPICOS

Fugas de dados, sniffing, roubo de credenciais, exfiltração de dados (ou Data-Disclosure – Divulgação Ilícita de dados que devem ser confidenciais)



I

Integridade

Os dados são exatos e não são alterados sem autorização, propiciando que possam ser processados corretamente

CONTROLOS

Hashes, assinaturas digitais, controlo de versões, integridade dos sistemas, proteção contra intrusões para quebras de integridade no processamento, isolamento

ATAQUES TÍPICOS

Adulteração de dados, *man-in-the-middle*, forjamento de dados, modificação ilícita,



A

Disponibilidade

Sistemas e dados acessíveis quando sempre quando são necessários.

CONTROLOS

Redundância, backups, proteção anti-DDoS

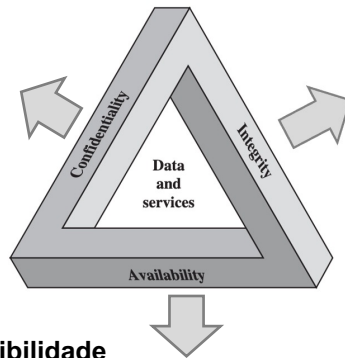
ATAQUES TÍPICOS

DDoS, ransomware, falhas de energia

Propriedades de Segurança: NIST TRIAD

Confidencialidade

- Dados
- Computações (ou processamento) (Computação confidencial)



Integridade

Integridade de dados (armazenados ou processados)

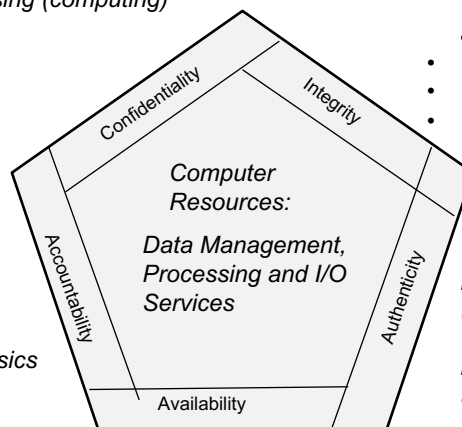
- Integridade do processamento
- Autenticação de sujeitos ou de componentes do sistema
- Autorização para controlo de acessos
- Não repudição
- Responsabilização (Accountability)

Disponibilidade

- Recuperabilidade
- Redundância com tolerância a falhas ou ataques

Extended representation of Security Goals and Properties

- Data confidentiality
- Confidential processing (computing)
- Non-Disclosure
- Data Privacy Controls



- Access control for authorized access (permissions and access management)
- Data Integrity
- System Integrity
- Trusted Execution

- Event-Registrations
- Logging
- Monitoring
- (Auditing and Forensics Analysis)
- Non-Repudiation

Peer-Authentication
(Principals, subjects, endpoints)
Data-Authentication
SW Attestation

- Timely and Reliable Access
- Available use (continuity)

Princípios gerais muito importantes (*mind-setting*)

- Saber como definir um sistema seguro (isso não significa um sistema “infinitamente” seguro)
- Estabelecer confiança ZERO (ZERO-TRUST)
- Isolar (contentorizar) o processamento e os dados críticos
- Segmentar a rede, e isolar segmentos (com microsegmentação) os sistemas críticos com as necessárias defesas de perímetro
- Diversificar (Segurança com Diversidade)
- Não esperar que não há adversários mas sim antecipar as suas possíveis ações e estar pronto para o receber e conter
- Um sistema não é seguro se não tiver sido construído com segurança por conceção e com base num modelo de adversário definido
- Ter em conta as dependências da base de confiança: reconhecer, identificar, minimizar e monitorizar as bases de confiança
- Ser paranoico (construtivo ;-)
- Saber comunicar e contribuir para a educação dos utilizadores em relação à cibersegurança
- Abordar a segurança com critérios de usabilidade

Que outros princípios ?

Princípios de segurança de sistemas e segurança por conceção (*Security Requirements By Design*)



Menor privilégio

Cada utilizador ou processo só tem o acesso estritamente necessário.



Defesas sobrepostas na profundidade

Várias camadas independentes: se uma falhar, outra protege.



Segurança por defeito

A configuração inicial já é a mais segura possível.



Separação de funções

Nenhuma pessoa controla sozinha um processo crítico.



Falhar de forma segura

Em caso de erro, o sistema nega o acesso em vez de o permitir.



Mediação completa

Cada acesso a cada recurso é verificado, sempre.



Zero Trust

Nunca confiar, verificar sempre — mesmo dentro da rede.



Simplicidade com usabilidade

Sistemas simples têm menos erros e menos superfície de ataque.

Outros princípios de segurança por concepção*

- *Open design (Transparency, No Security by Obscurity)*
- *Economy of mechanisms*
- *Fail-safe defaults*
- *Complete mediation*
- *Separation of privilege*
- *Least privilege*
- *Least common mechanism*
- *Psychological acceptability*
- *Isolation (Ex., by Virtualization, Containment, Compartmentalization)*
- *Encapsulation*
- *Modularity*
- *Layering*
- *Least astonishment (or least-surprise, no-disturbance)*

*) Computer Security: Principles and Practica, 5th Ed., Prentice Hall/Pearson Ed., 2024

Ameaças, vulnerabilidades, ataques e incidentes

- Potencial de ameaça (ou **THREAT**): ameaça no risco identificado, mas não necessariamente concretizado sob a forma de um ataque
- Ataque (**ATTACK**): uma ação deliberada que concretiza uma ameaça, usando um vetor (**ATTACK VECTOR** para a concretizar
Um vetor de ataque pode ser baseado numa ferramenta de ataque (**EXPLOIT TOOL**)
- Os Ataques exploram vulnerabilidades expostas pelos sistemas (**VULNERABILITIES**)
- Os ataques podem explorar vulnerabilidades não conhecidas por quem defende ou por quem desenhou e concebeu o sistema, quem o implementou e quem o operacionalizou (**ZERO DAY VULNERABILITIES**)
- Os sistemas podem conter vulnerabilidades por diferentes razões: deficiências de concepção, erros de concepção ou porque não se assumiu essa vulnerabilidade na definição do modelo de adversário ou a probabilidade disso ser explorado por vetores de ataque no modelo de risco
- Os ataques podem ser **passivos** ou **ativos**, podem ser **silenciosos** ou **detetáveis** e podem ser **determinísticos** (ex., *modelo fail-stop*) ou **arbitrários** (modelo de *arbitrary failures*)

Bibliografia

- [CS] W. Stallings, L. Brown, Computer Security – Principles and Practice, Person, 5h Ed., 2024. Ch.1 - Overview
- [NSE] W. Stallings, Network Security Essentials – Applications and Standards, Ch.1 – Introduction
- Paul C. Van Oorschot, Chap.1 – Security Concepts and Principles
<https://people.scs.carleton.ca/~paulv/toolsjewels/TJrev1/ch1-rev1.pdf>

Outras referências e recursos

NIST CSF 2.0

Cybersecurity Framework:
estrutura de gestão de
risco.

nist.gov/cyberframework

OWASP

Top 10, ASVS e Web
Security Testing Guide.

owasp.org

MITRE ATT&CK

Táticas e técnicas de
adversários reais.

attack.mitre.org

CIS Controls

Conjunto priorizado de
boas práticas defensivas.

cisecurity.org

NVD · CVE · CISA KEV

Bases de dados públicas
de vulnerabilidades.

nvd.nist.gov

CNCS e ENISA

Alertas, guias e boas
práticas nacionais e

cncs.gov.pt ·
enisa.europa.eu

Ataques a servidores (Endpoints)

Categoria	Exemplos	Propriedade afetada
Reconhecimento	Port scanning, fingerprinting de serviços	Confidencialidade
Negação de serviço	SYN flood, DDoS, amplificação UDP	Disponibilidade
Exploração de vulnerabilidades	Buffer overflow, SQL injection, execução remota de código	Integridade
Acesso não autorizado	Força bruta, credential stuffing, escalada de privilégios	Confidencialidade
Malware	Ransomware, backdoors, rootkits	Todas
Interceção	Sniffing, ARP/DNS spoofing, man-in-the-middle	Confidencialidade

Contramedidas

Ataque	Contramedidas primárias
Reconhecimento	Firewall, desativar serviços desnecessários, IDS/IPS
Negação de serviço	Limitação de taxa, SYN cookies, filtragem anti-DoS/DDoS – <i>Denial of Service</i> , <i>Distributed Denial of Service</i> e CDN, redundância, replicação
Vulnerabilidades	Atualizações (<i>patches</i>), validação de entradas, <i>queries</i> parametrizadas, WAF / Application-Level Firewalls (H-WAF),
Acesso não autorizado	Palavras-passe fortes, autenticação multifatorial, bloqueio de contas, mínimo privilégio, separação de privilégios, mediação completa no controlo de acesso
Malware	Antivírus/EDR, backups offline, lista de aplicações autorizadas
Interceção	Cifragem (TLS, SSH, VPN), DNSSEC, segmentação de rede

Conjugação de defesas de profundidade

Rede

Firewalls, DMZ, segmentação, IDS/IPS e filtragem anti-DDoS

Sistema

Hardening, patches, antimalware/EDR e controlo de acessos

Aplicação

Validação de entradas, WAF, TLS e gestão segura de sessões

Processos

Logs e monitorização, backups, resposta a incidentes

Pessoas / Utilizadores

Autenticação forte, Autenticação Multifatorial, Protocolos de Autenticação com Segurança e Sistemas Robustos de autenticação contínua, Formação

Software malicioso ou *malware*

Tipo	Comportamento	Contramedidas
Vírus	Anexa-se a programas ou ficheiros e propaga-se quando são executados	Antivírus atualizado, verificação de ficheiros
Worm	Propaga-se sozinho pela rede, explorando vulnerabilidades	Patches, firewall, segmentação de rede
Trojan e backdoor	Fingem ser software legítimo e abrem acesso remoto ao atacante	Software de fontes fiáveis, lista de aplicações autorizadas, EDR
Ransomware	Cifra os dados e exige resgate	Backups offline testados, mínimo privilégio
Spyware e keylogger	Recolhem em segredo dados e credenciais do utilizador	Antimalware, autenticação multifator

Software malicioso ou *malware*

Tipo	Comportamento	Contramedidas
Vírus	Anexa-se a programas ou ficheiros e propaga-se quando são executados	Antivírus atualizado, verificação de ficheiros
Worm	Propaga-se sozinho pela rede, explorando vulnerabilidades	Patches, firewall, segmentação de rede
Trojan e backdoor	Fingem ser software legítimo e abrem acesso remoto ao atacante	Software de fontes fiáveis, lista de aplicações autorizadas, EDR
Ransomware	Cifra os dados e exige resgate	Backups offline testados, mínimo privilégio
Spyware e keylogger	Recolhem em segredo dados e credenciais do utilizador	Antimalware, autenticação multifator

Ataques a comunicações (Rede)

Ataques passivos

O atacante observa ou recolhe informação, sem alterar dados nem o funcionamento da rede.

Alvo: confidencialidade

Deteção: Pode ser difícil

Foco: prevenção.

Ataques ativos

O atacante modifica, cria, repete ou interrompe dados e serviços.

Alvo: integridade, autenticidade e disponibilidade

Deteção: possível por deteção do estado incorreto induzido

Foco: deteção e resposta.

Tipos de ataques passivos a comunicações (Rede)

Ataques passivos	Descrição	Contramedidas primárias
Interceção (e <i>packet sniffing</i>)	Captura de pacotes e leitura do conteúdo das comunicações	Cifragem (TLS, VPN, SSH), switches em vez de hubs
Análise de tráfego (<i>traffic-analysis</i>)	Inferência a partir de origens, destinos, volumes e horários, mesmo com dados cifrados	Padding, tráfego de cobertura, anonimização
Reconhecimento (<i>Scanning of Endpoints</i>)	Port scanning e mapeamento da rede e dos serviços	Firewall, IDS/IPS, redução de serviços expostos

Tipos de ataques ativos

Ataque	Descrição	Contramedidas primárias
Disfarce (<i>Spoofing. Masquerading</i>) <i>IP Spoofing, MAC Spoofing</i> <i>Sybil-Attacks</i>	Falsificação ou mascaramento de identidade: IP, MAC, DNS	Filtragem de endereços, Controlo de Acesso na Rede,, DHCP <i>snooping</i> , DNSSEC (DNS <i>Security</i>)
Repetição (replay)	Reenvio de mensagens válidas capturadas	Números de sequência, <i>timestamps</i> , <i>nonces</i> protegidos em confidencialidade
Modificação / <i>Man-in-the-middle</i> (<i>Message-Tampering</i>), Forjamento de tráfego/pacotes falsos	Alteração de mensagens ou sequestro de sessões em trânsito	Autenticação mútua, certificados digitais, <i>Hashing</i> de mensagens, MAC e assinaturas digitais
Negação de serviço (DoS/DDoS)	Saturação de recursos para tornar o serviço indisponível	Limitação de taxa,, filtragem anti-DoS/DDoS no perímetro (NIDS, HIDS), redundância

Outras tipologias de ataques

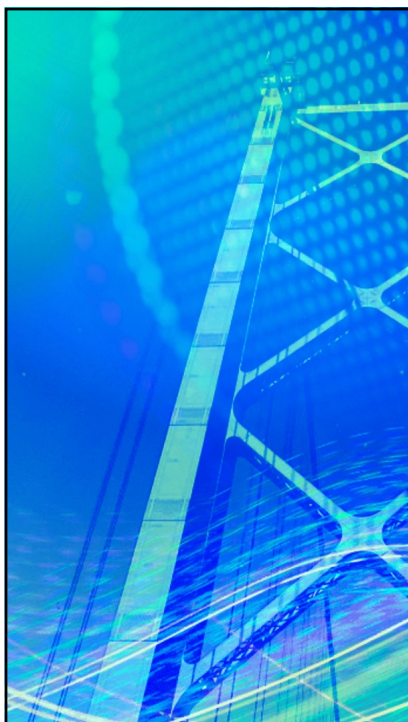
Ataques ao SOFTWARE

Ataques aos Utilizadores / Fator Humano, Pessoas

Ataques de IA

Noções Fundamentais de Segurança Informática

Questões sobre a PARTE I



2. Segurança de Servidores e Suporte de Comunicações Cliente/Servidor

Segurança de Servidores

Tópicos neste bloco



**Arquitetura cliente-
servidor**



**Segurança em
servidores**



Gestão de serviços



**Riscos e níveis de
exposição**

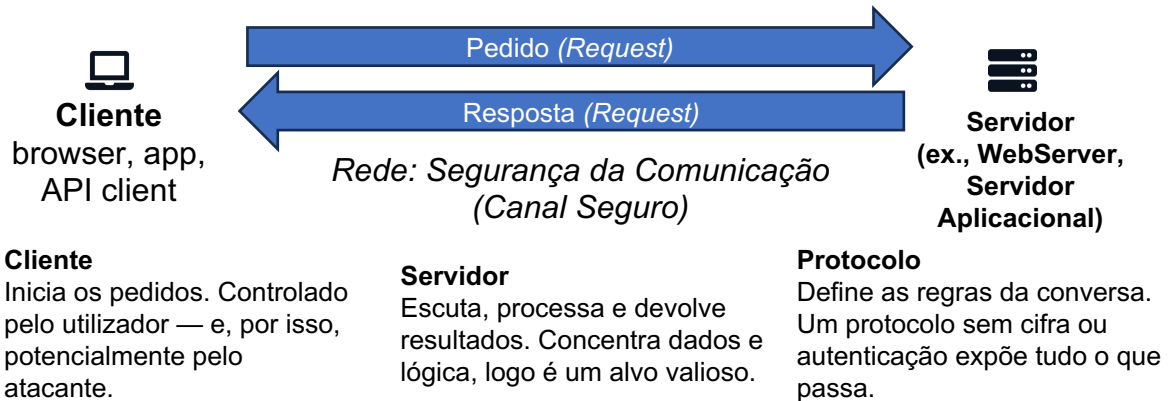


Contramedidas



**Desenvolvimento
seguro**

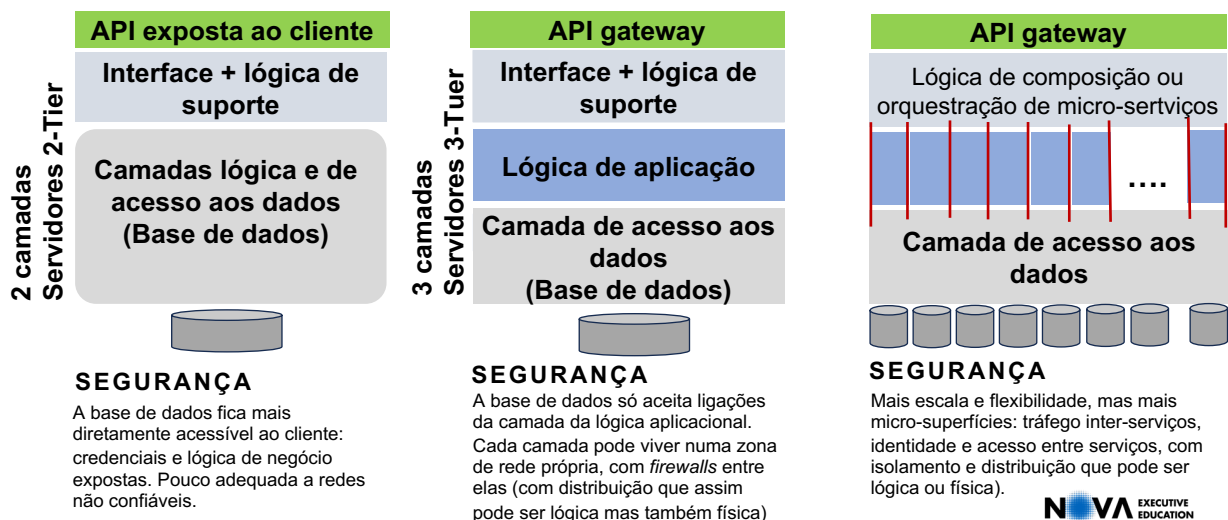
Modelo cliente-servidor



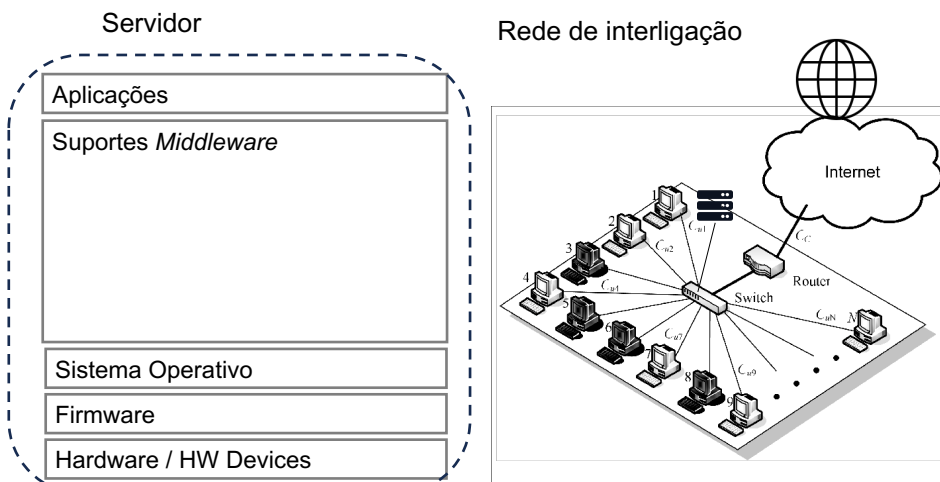
Fronteira de confiança: ZERO TRUS:

O servidor nunca deve confiar no cliente — tudo o que chega deve de ser validado.

Servidores e Arquiteturas de SW



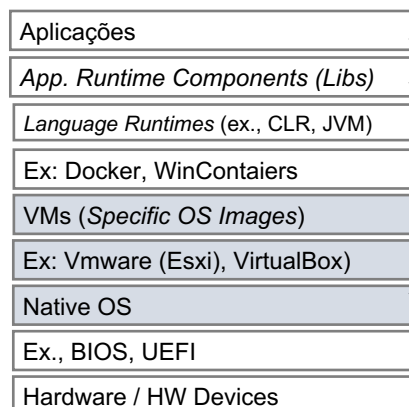
Níveis de suporte (SW) em servidores



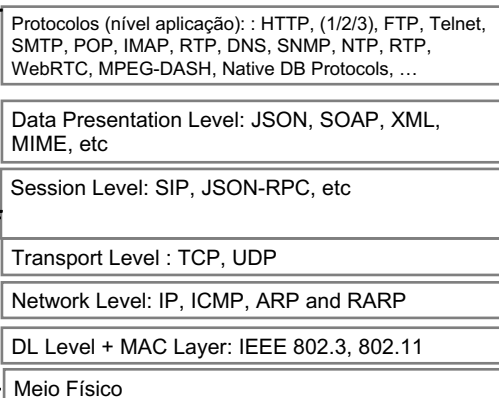
63

Níveis de serviços de SW num servidor (Exemplificação)

SERVIDOR

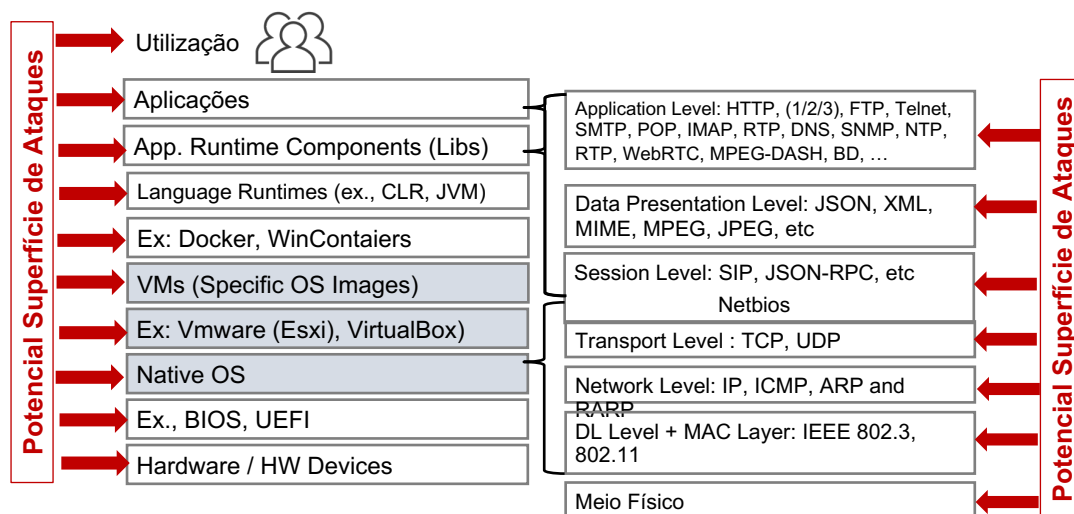


Níveis de protocolos na rede de acesso

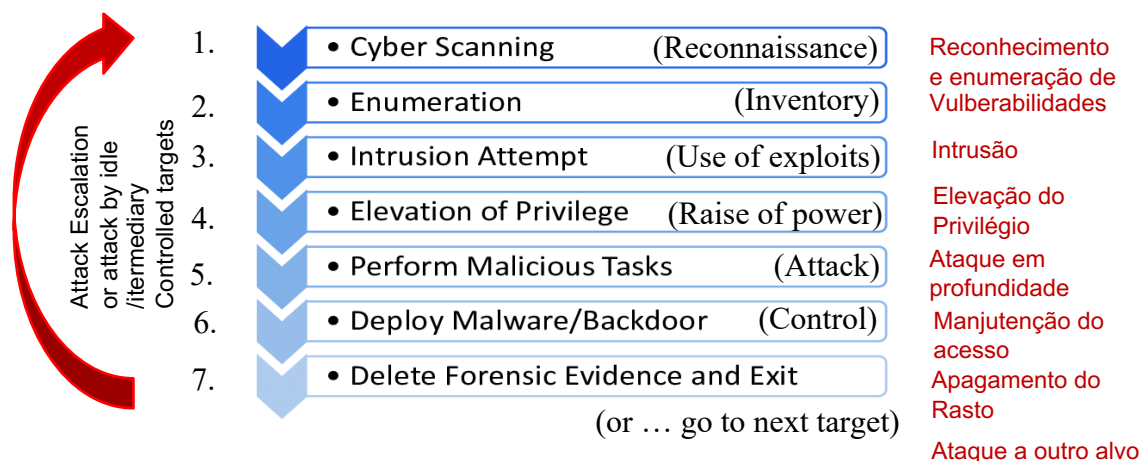


64

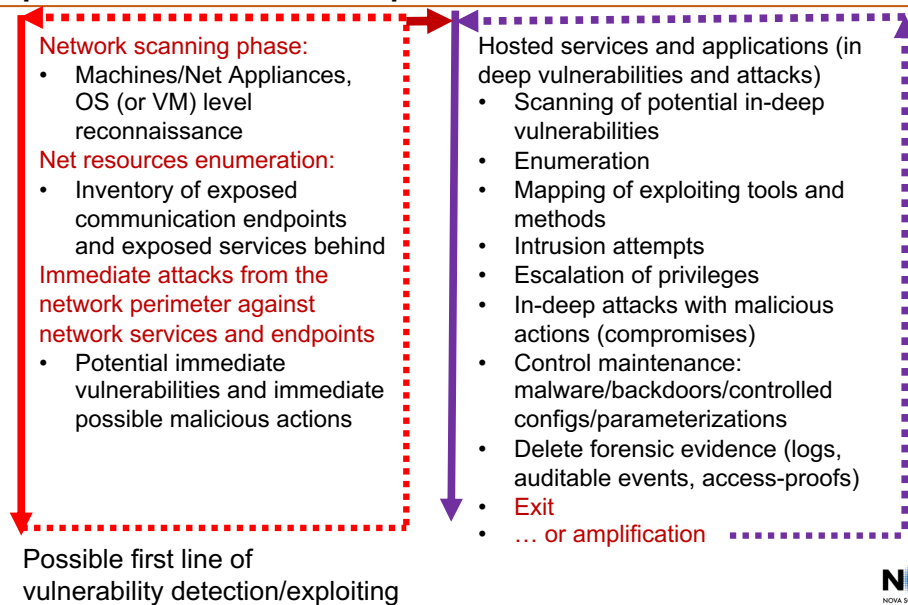
Níveis de ataques nas superfícies de ataque



Anatomia de um ataque típico



Exemplo: fases do ataque



67

A ter em conta ...

- A deteção de ataques pode não funcionar para ataques silenciosos ou ataques de exploração de vulnerabilidades ZERO DAY
- A recuperação de ataques pode ser de diferentes tipos:
 - **Recuperação reativa:** reposição do sistema após deteção do ataque, com uma certa métrica de disponibilidade e correção de vulnerabilidade
 - **Recuperação pró-ativa:** reposição periódica, independentemente da deteção
 - **Tolerante ao ataque:** quando o sistema se torna imune e pode recuperar sem paragem, mantendo o funcionamento correto

Defesas com redundância ativa e replicação contra ataques arbitrários

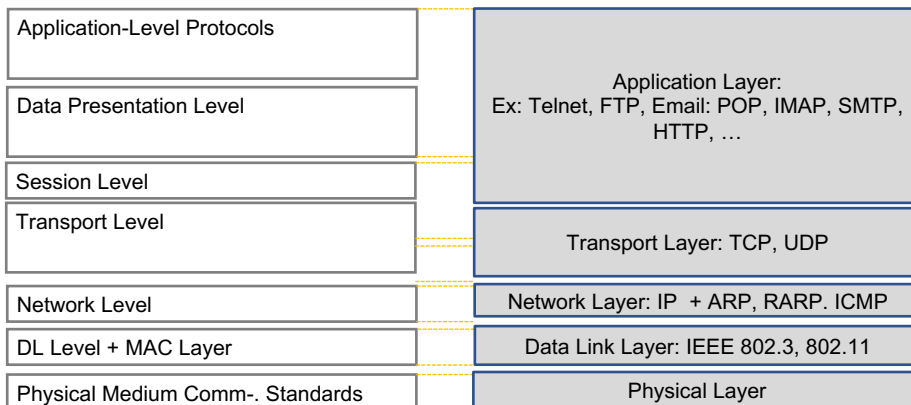
Relevância dos modelos de replicação e tolerância a intrusões com tolerância a falhas bizantinas

68

Redes de acesso: *TCP/IP Stack* sem segurança

Níveis de protocolos
na pilha TCP/IP

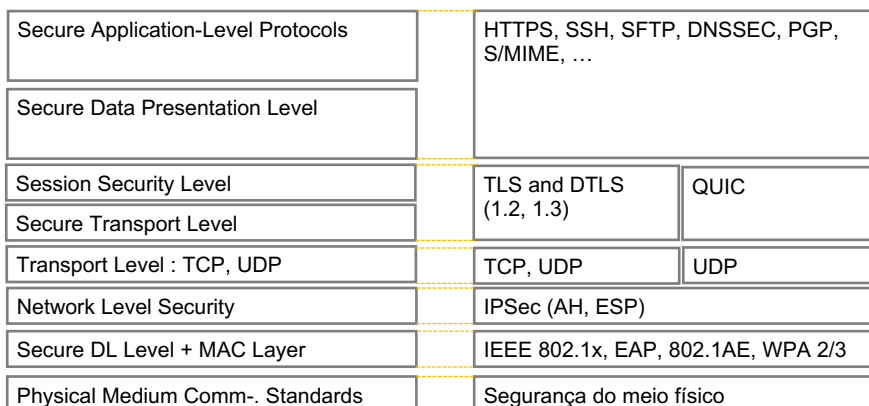
Tipologia de protocolos não seguros na
pilha TCP/IP



Redes de Acesso: *TCP/IP Security Stack*

Níveis de protocolos
na pilha TCP/IP

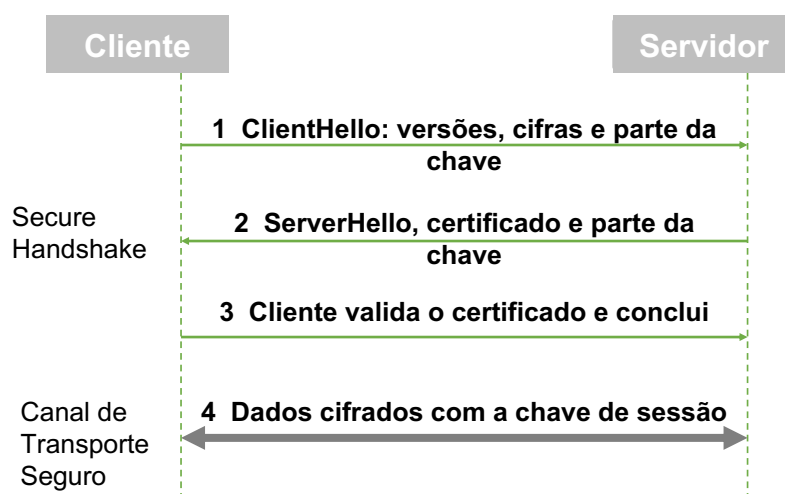
Pilha de protocolos de segurança
na pilha TCP/IP



Protocolos e acesso a serviços comuns (na *suite* de protocolos TCP/IP)

Protocolo	Porta(s)	Função	Risco e boa prática
HTTP / HTTPS	80 / 443	Web e APIs	Forçar HTTPS e HSTS (suportado em TLS 1.2 ou 1.3)
SSH	22	Administração remota	Chaves criptográficas em vez de passwords; limitar origens
DNS	53	Resolução de nomes	<i>Poisoning</i> e amplificação; DNSSEC, rate limiting
SMTP / IMAP	25, 587 / 993	Correio eletrónico	<i>Spoofing</i> ; SPF, DKIM e DMARC
RDP	3389	Ambiente de trabalho remoto	Nunca expor à internet; VPN + MFA
SMB	445	Partilha de ficheiros	Alvo de <i>worms</i> ; bloquear no perímetro
FTP / Telnet	21 / 23	Legado, inseguro	Substituir por SFTP / SSH

Comunicação segura com TLS



O que o TLS garante em geral:

confidencialidade, integridade e autenticação do servidor.

Boas práticas

TLS 1.2 ou superior (preferir 1.3), HSTS, renovação automática de certificados.

Segurança de servidores

Objetivo: redução da Superfície de Ataque



Instalação mínima

Apenas o software e os serviços estritamente necessários.



Patching regular

Sistema operativo e aplicações atualizados e testados.



Contas e privilégios

Sem contas por defeito; sudo e acessos mínimos.



Configuração segura

Seguir benchmarks (CIS) e remover defaults inseguros.



Firewall local

Permitir apenas as portas e as origens necessárias.



Cifra e registos

Discos e segredos cifrados; logs enviados para fora do servidor.

Acessos, autenticação e isolamento de segredos



SSH com chaves

Autenticação por chave pública; desativar login de root e por password.



MFA e acesso privilegiado

MFA na administração; bastion host e PAM com sessões registadas.



Menor privilégio

sudo com comandos limitados; contas de serviço sem login interativo.



Gestão de segredos

Cofre central, rotação automática e segredos fora do código.

Segredos nunca vão para o código nem para o repositório.

Usar cofres digitais (por exemplo Vault, Azure Key Vault, AWS Secrets Manager).

Segurança e ameaças típicas por tipo de servidor

Servidor	Ameaças típicas	Medidas-chave
Web / aplicação	Injeção, XSS, falhas de autenticação, DoS	WAF, TLS, validação de input, patching, reverse proxy
Base de dados	SQL injection, acesso direto, fuga de dados	Rede isolada, contas mínimas, cifra, auditoria
E-mail	Phishing, spoofing, relay aberto, malware	SPF, DKIM e DMARC, anti-spam, TLS, filtragem de anexos
DNS	Cache poisoning, amplificação DDoS, hijacking	DNSSEC, rate limiting, recursão restrita
Acesso a Ficheiros	Ransomware, movimento lateral, partilhas abertas	Permissões mínimas
Diretórios (AD/LDAP)	Roubo de credenciais, escalada de privilégios	Segregação de contas e permissões, MFA, <i>hardening</i> dos DC, monitorização

75

Virtualização, contentores e cloud



Máquinas virtuais

Isolamento forte via hypervisor.

- Manter o hypervisor atualizado
- Controlar snapshots com dados
- Evitar VM esquecidas



Contentores

Partilham o kernel do host anfitrião.

- Imagens mínimas e de origem fiável
- Não executar como root
- Scan de imagens e limites de recursos



Cloud

Infraestrutura gerida por terceiros.

- IAM com menor privilégio
- Sem buckets públicos por engano
- Registos e alertas ativos

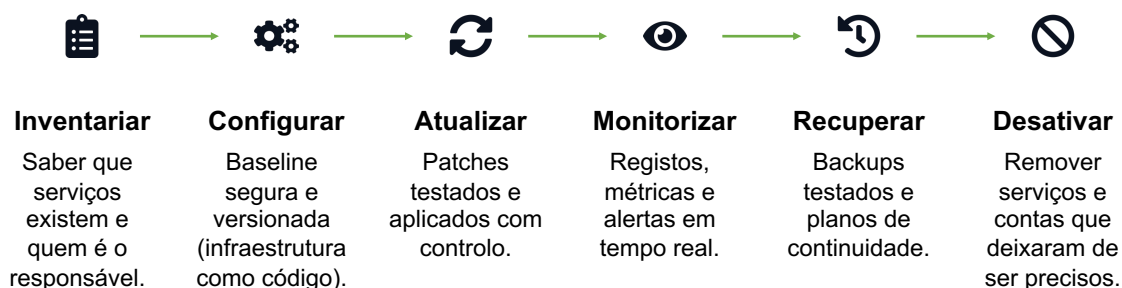
Responsabilidade é partilhada no modelo bipartido cloud

Fornecedor: hardware, rede física, virtualização (modelo IaaS) podendo ir até oferta de “suportes *runtime* específicos para *deployment* de aplicações (modelo PaaS) ou mesmo de fornecimento de aplicações finais como serviço (modelo SaaS)

Cliente: dados, identidades, configuração e uso de aplicações.

76

Ciclo de vida de serviços suportados em servidores

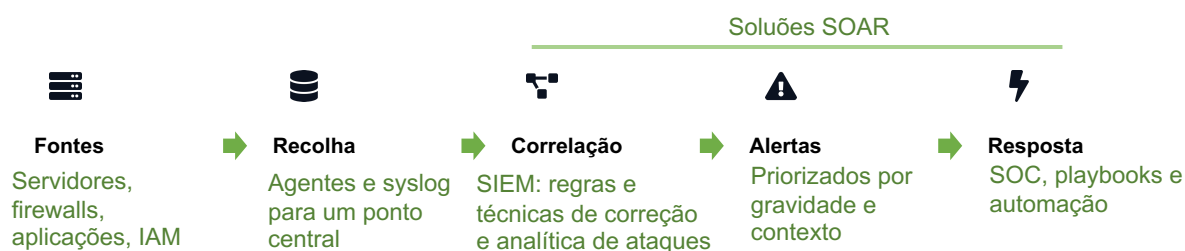


Gestão de alterações (change management)

Pedido e aprovação Teste prévio Plano de reversão Registo e revisão

77

Registos, monitorização e deteção



O que registar

- Autenticações, com sucesso e com falha
- Alterações de privilégios e de configuração
- Acessos a dados sensíveis
- Erros de aplicação e eventos de rede relevantes

Boas práticas

- Sincronizar relógios (NTP)
- Enviar logs para fora do servidor
- Proteger a integridade e definir a retenção
- Nunca registar passwords nem dados sensíveis

SOAR: Security Orchestration, Automation, and Response

78

Backups e DR para continuidade

3-2-1

- 3 cópias dos dados
- 2 suportes diferentes
- 1 cópia offline ou imutável

RPO *Recovery Point Objective*

Quantidade máxima de dados que se aceita perder, medida em tempo desde o último backup.

RTO *Recovery Time Objective*

Tempo máximo aceitável de indisponibilidade até o serviço estar de novo operacional.

Um backup que nunca foi restaurado (em teste) não é um backup seguro. Testar restauros Recuperar com regularidade e manter cópias imutáveis (muito importante, por exemplo contra ransomware).

Risco de exposição: zonas e níveis de exposição

Internet

Exposição máxima

- Clientes legítimos
- Parceiros e APIs externas
- Atacantes e bots

DMZ

Exposição alta

- Servidor web / proxy
- Gateway de e-mail
- DNS público

Rede interna

Exposição média

- Servidores de aplicação
- Servidores de ficheiros
- Postos de trabalho

Zona crítica

Exposição baixa

- Bases de dados
- Diretório (AD)
- Backups e gestão

Confiança e controlo aumentam para o interior

Regra: o tráfego só atravessa zonas através de firewall e com sistemas de deteção de intrusões, com regras mínimas (deny by default) e control de eventos anómalos

Alguns riscos comuns em servidores



Serviços expostos sem necessidade

RDP, SSH, bases de dados ou painéis de administração acessíveis na internet.



Credenciais fracas ou por defeito

Força bruta e *credential stuffing* contra serviços de autenticação.



Software por atualizar

Exploração de CVEs conhecidos, por vezes poucos dias após a divulgação.



Má configuração

Permissões excessivas, TLS fraco (ou versão insegura), ficheiros em buckets públicos sem dados cifrados



Negação de serviço

Esgotar a largura de banda, as ligações ou os recursos do servidor.



Ataques em cadeia

Um servidor comprometido serve de ponte para os restantes.

CVE – Common Vulnerability Exposure

Credential Stuffing: quando se utilizam listas de nomes de utilizador e passwords que foram expostos em fugas de dados anteriores

Tecnologias de proteção para acesso na rede

Tecnologia	Onde atua	Protege contra
Firewall (<i>stateful Inspection</i>) / Tecnologias NGFW)	Perímetro e entre zonas	Tráfego não autorizado e varrimentos de portas
WAF	À frente de aplicações web	Injeção, XSS, bots e abuso de APIs
<i>Reverse proxy / load balancer</i>	Entrada dos serviços	Exposição direta dos servidores e picos de carga
IDS / IPS	Tráfego de rede	Padrões de ataque conhecidos e anomalias
VPN	Acesso remoto	Exposição de serviços de administração
Anti-DDoS	No perímetro de acesso de rede	Negação de serviço volumétrica
Segmentação de VLANs (com proteção de acesso)	Rede interna	Acesso na comunicação “lateral”

Tecnologias a ter em conta para verificação



**Anti-virus,
Anti-malware**



**Integridade de
ficheiros (HIDS)**



**Controlo de acesso
obrigatório:
Papel das soluções
para IAM com
autenticação
multifatorial**



**PAM e cofre de
segredos**

**Pode envolver
isolamento
cruográfico**



**Gestão de
configuração e
*patching***



**Testar/Uso de
Scanners de
vulnerabilidades**

83

Um Pipeline para DevSecOps



Commit

Hooks de
pré-commit e
detecção de
segredos
(Gitleaks).



Build

SAST
(Semgrep,
CodeQL) e
SCA
(Dependabot
, Trivy).



Imagem

Scan de
imagens
(Trivy) e
assinatura de
artefactos.



Teste

DAST
(OWASP ZAP)
e testes de
segurança
automatizados.



Deploy

Verificação
de IaC e
políticas
(Checkov,
OPA).



Operação

Monitorizaçã
o, WAF e
gestão de
vulnerabilida
des.

Quality gates

O pipeline falha se houver
vulnerabilidades críticas.

Segredos

Num cofre, nunca em
variáveis expostas nem
no repositório.

CVE-LISTS

Lista de componentes
para responder depressa
a novos CVE.

84

Uma verificação final interessante: Exposição de Servidores WEB HTTPS e nível de Segurança exposto

Análises

<https://internet.nl/>

<https://www.ssllabs.com/ssltest/>

Pontos Chave a Reter

- 1 O servidor nunca deve confiar no cliente: validar tudo na fronteira.
- 2 Hardening, patching e menor privilégio reduzem a superfície do servidor.
- 3 Gerir serviços é um ciclo: inventariar, configurar, atualizar, monitorizar, recuperar.
- 4 Exposição e criticidade determinam as prioridades de proteção.
- 5 A segurança constrói-se desde o desenho

Pilares fundamentais

- 1 **Autenticação e Autorização:** Garantir que apenas clientes legítimos acedem ao servidor e que têm apenas as permissões necessárias (princípio do privilégio mínimo).
- 2 **Cifragem de Dados e** Utilização de protocolos seguros no acesso a servidores para proteger os dados em trânsito entre o cliente e o servidor.
- 3 **Isolamento (com micro-segmentação nas redes de acesso) e uso de Firewalls e Sistemas de Detecção de intrusões (IDS/IPS):** Proteção do perímetro do servidor contra ataques externos, acessos internos não autorizados e malware e reduzir os acessos aos protocolos mínimos necessários
- 4 **Gestão de Patches:** Aplicação regular de atualizações do sistema operativo e do software do servidor para corrigir falhas de segurança com planeamento de paragens técnicas em horários de menor impacto para os utilizadores (clientes).
- 5 **Cópias de Segurança (Backups):** Automatização de backups periódicos e testes de recuperação de dados.

Pilares fundamentais da segurança de servidores

- 6 **Redundância:** Utilização de sistemas de armazenamento redundante de dados (como RAID) e fontes de alimentação duplicadas para evitar pontos únicos de falha.
- 7 **Logs e Auditoria:** Gerir históricos de eventos de todas as atividades do servidor para diagnóstico de problemas e auditorias de segurança e registar os eventos externamente em plataformas de gestão de eventos (ex., SIEM – *Security Information and Event Management*).
- 8 **Controlo de Limites (Rate Limiting):** Restringir o número de pedidos que um único cliente pode fazer num determinado intervalo de tempo para mitigar ataques de força bruta e negação de serviço (DoS).
- 9 **Identificação Mútua robusta do lado cliente:** O servidor deve autenticar o cliente (através de credenciais e provas multifatoriais ou usando certificados digitais e métodos criptográficos e o cliente deve validar a autenticação do acesso ao servidor para evitar ataques de *spoofing*.
- 10 **Gestão de Sessões segura:** Utilizar *tokens* de sessão criptográficos (como JWT) com tempos de expiração curtos e mecanismos de revogação imediata e garantir o suporte do **Princípio do Menor Privilégio:** Limitar as permissões de cada utilizador ou processo do cliente estritamente ao necessário para executar as suas funções no servidor.

Sugestões de Bibliografia

W. Stallings, L. Brown, Computer Security – Principles and Practice, 5th ED.
Person, 2024

Outras referências e recursos

NIST SP 800-123

Guia de segurança geral
para servidores.

csrc.nist.gov

NIST SP 800-218

Secure Software
Development Framework
(SSDF).

csrc.nist.gov

CIS Benchmarks

Guias de configuração
segura por sistema.

cisecurity.org

Mozilla TLS

Gerador de configurações
TLS recomendadas.

ssl-config.mozilla.org

OWASP

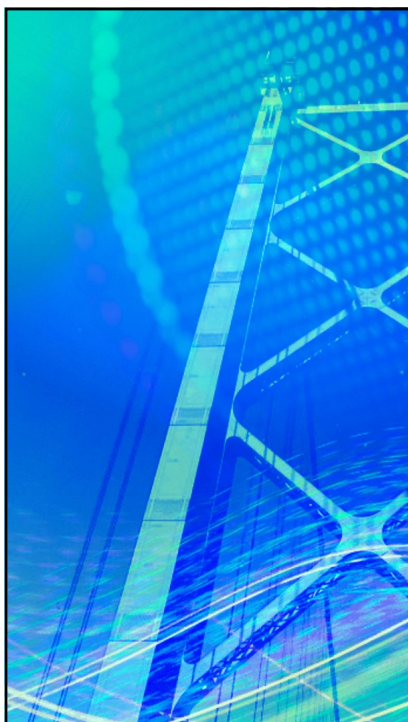
Top 10, ASVS, API
Security e Cheat Sheets.

owasp.org

CNCS e ENISA

Boas práticas e alertas
nacionais e europeus.

cncs.gov.pt
enisa.europa.eu



3. Criptografia

Visão Resumo

Criptografia e aspetos a reter

- 1 A criptografia é um mecanismo muito importante para confidencialidade, integridade, autenticidade e não-repúdio
- 2 Na prática combinam-se simétrica (dados), assimétrica (chaves e assinaturas) e hash (integridade) bem como MACs (para autenticidade e integridade de dados)
- 3 Usar algoritmos e modos recomendados e gerir bem as chaves: a maioria das falhas está na implementação.
- 4 O computador quântico ameaça RSA, DH e ECC; AES-256 e SHA-2/SHA-3 mantêm-se.
- 5 Para preparar o futuro: inventariar, ganhar cripto-agilidade e adotar o modo híbrido (ML-KEM) para proteção de dados de longa duração.

Referências e recursos



NIST: criptografia pós-quântica

Projeto PQC e normas FIPS 203, 204 e 205.

csrc.nist.gov/projects/post-quantum-cryptography



NIST SP 800-57 e 800-131A

Gestão de chaves e transição de algoritmos.

csrc.nist.gov



OWASP Cheat Sheets

Password Storage, Cryptographic Storage e outros.

cheatsheetseries.owasp.org



ENISA e roteiro da UE

Transição coordenada para a criptografia pós-quântica.
enisa.europa.eu



Open Quantum Safe

Biblioteca liboqs e protótipos para experimentar PQC.
openquantumsafe.org



CNCS

Boas práticas e orientações nacionais de cibersegurança.
cncs.gov.pt

Tópicos sobre criptografia nesta parte



Relevância e Conceitos base

Objetivos, primitivas e força das chaves



Criptografia Simétrica e assimétrica

AES, ChaCha20
RSA, ECC
e troca de chaves (Diffie Hellman)



Outros métodos

Hash (SHA-2, SHA-3)
MAC: HMAC, CMAC
Assinaturas Digitais
PKI e KDF



Algoritmos relevantes

O que usar e o que evitar



Aplicações práticas

Dados em Trânsito,
Dados em repouso,
identidade e erros



Era pós-quântica

Ameaça, normas NIST e migração para criptografia pós-quântica

Relevância da criptografia

Os métodos e algoritmos criptográficos são mecanismos fundamentais de comprovação formal para serviços de segurança amplamente utilizados, ex:

- Suporte de proteção de dados em trânsito: HTTPS/TLS, VPN, SSH, e-mail cifrado.
- Proteção dados em repouso: cifragem de discos, bases de dados ou backups.
- Proteção de dados em uso (que estão a ser processados): ainda um desafio (computação confidencial).
- Assinaturas e certificados: software, documentos, provas de identidade digital.

O que protegé a criptografia?



Confidencialidade

Só o destinatário consegue ler.



Integridade

Qualquer alteração é detetada.



Autenticidade

Prova de quem enviou.



Não-repúdio

O autor não pode negar a ação.

Princípio de Kerckhoffs: a segurança deve depender apenas do segredo da chave, não do segredo do algoritmo (ou seja, a segurança criptográfica não deve ser obtida por obscurantismo dos algoritmos criptográficos)

Primitivas criptográficas



Cifra simétrica

Confidencialidade rápida com uma chave partilhada.



Cifra assimétrica

Troca de chaves sem segredo prévio, com par público/privado.



Função de hash

Impressão digital de dados: integridade.



MAC / HMAC

Integridade e autenticidade com chave partilhada.



Assinatura digital

Autenticidade e não-repúdio com chave privada.



KDF

Deriva chaves a partir de passwords ou segredos.



Gerador aleatório

Chaves e nonces imprevisíveis (CSPRNG).



Protocolos que usam criptografia

TLS, SSH, Signal, WhatsUp. combinam várias primitivas.

Força das chaves e boas regras

Segurança	Simétrica	RSA / DH	Curvas elípticas
112 bits	3DES (obsoleto)	2048 bits	224 a 255 bits
128 bits	AES-128	3072 bits	256 a 383 bits
192 bits	AES-192	7680 bits	384 a 511 bits
256 bits	AES-256	15360 bits	512 bits ou mais

*Equivalência aproximada de segurança
(Ref. NIST SP 800-57).*

Algoritmos públicos

Usar normas abertas e escrutinadas por anos.

Nunca criar a própria criptografia (ou modificar um algoritmo existente)

Usar bibliotecas reconhecidas, não código caseiro.

A chave é o ativo crítico

A segurança vale o que valer a geração e a proteção da chave.

Cifra simétrica



Algoritmos criptográficos simétricos

Cifras de bloco (ou Block Ciphers)
Ex: AES

Cifras em cadeia (ou Stream Ciphers)
Ex., ChaCha20

101

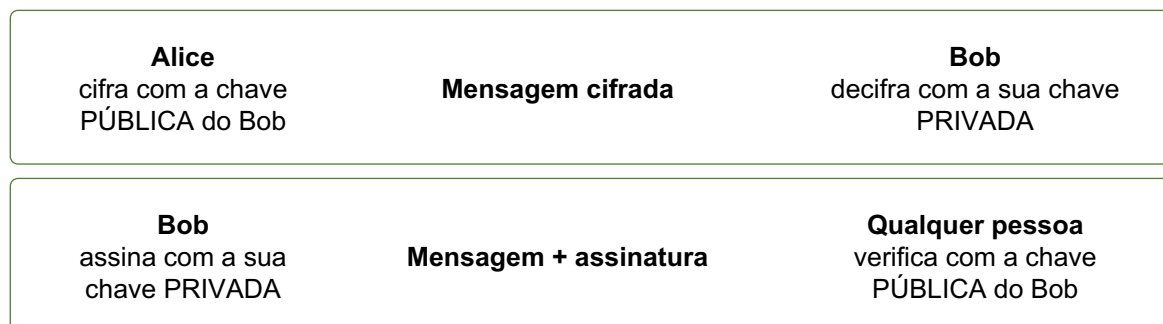
Modos de operação de cifras de bloco

Modo	Funcionamento	Garantias	Veredicto
ECB	Cada bloco é cifrado de forma independente	Revela padrões; sem integridade	Nunca usar
CBC	Encadeia blocos com um IV aleatório	Só confidencialidade; risco de padding oracle	Legado; juntar MAC
CTR	Transforma a cifra de bloco numa cifra de fluxo	Só confidencialidade; nonce único	Bom, com MAC
Modos AEAD GCM / ChaCha20- Poly1305	Cifra e autentica numa só operação (AEAD)	Confidencialidade + integridade	Recomendado

AEAD – Authenticated Encryption w/ Associated Data

102

Encriptação assimétrica e (alguns) modos de operação



RSA

Assenta na dificuldade de fatorizar. Chaves de 3072 bits ou mais.

Curvas elípticas (ECC)

Chaves curtas, mais rápidas: X25519, Ed25519, P-256.

Diffie-Hellman

Permite acordar um segredo partilhado em canal público.

Simétrica vs. assimétrica

Característica	Simétrica	Assimétrica
Chaves	Uma chave partilhada	Par de chaves: pública e privada
Velocidade	Muito rápida	Muito mais lenta
Tamanho da chave	128 a 256 bits	256 bits (ECC) a 3072 ou mais (RSA)
Distribuição da chave	Difícil: exige canal seguro	Simples: a chave pública pode ser divulgada
Usos principais	Cifrar dados em volume	Troca de chaves e assinaturas
Exemplos	AES, ChaCha20	RSA, ECC, X25519

Criptografia híbrida: a assimétrica estabelece uma chave de sessão; a simétrica cifra os dados. É assim que o TLS funciona.

Troca de chaves e *forward secrecy*

1 Gerar

Cada parte gera um par de chaves efêmero (privada e pública).

2 Trocar

As chaves públicas são enviadas em claro: todos as podem ver.

3 Combinar

Cada um junta a sua privada com a pública do outro.

4 Derivar

Ambos obtêm o mesmo segredo e derivam chaves de sessão (HKDF).

Forward secrecy (ECDHE)

Chaves efêmeras por sessão: se a chave do servidor for roubada mais tarde, o tráfego passado continua protegido.

Autenticar a troca

Diffie-Hellman sem autenticação é vulnerável a man-in-the-middle. Por isso combina-se com assinaturas e certificados.

Funções de hash

Determinística

A mesma entrada dá sempre o mesmo resultado.

Unidirecional

Não se recupera a entrada a partir do hash.

Resistente a colisões

É inviável encontrar duas entradas com o mesmo hash.

Efeito de avalanche

Uma pequena alteração muda o hash por completo.

Hash não é cifra: não tem chave nem se reverte. Usos: integridade de ficheiros, assinaturas, blockchain e (com KDF) armazenamento de passwords. Recomendados: SHA-256, SHA-384, SHA-3.

EFEITO DE AVALANCHE · SHA-256

entrada: "cibersegurança"
8a952c1ab4f3797a053220b8b2de1685
160651bc6576d57704f42034773c002b

entrada: "Cibersegurança"
4d20bfdb5a9a51657dcadc737d11841e
7c7491a7ba5045163b92d836ce4f6004

Armazenar passwords com segurança

EVITAR

- Texto limpo ou cifra reversível
- MD5, SHA-1 ou SHA-256 simples
- **Hashes rápidos e sem salt: A EVITAR**

Um hash rápido deixa o atacante testar milhares de milhões de palpites por segundo (por ex, pode usar potência para 10^{12} h/s)

USAR

- Argon2id (primeira escolha)
- scrypt, bcrypt ou PBKDF2
- Salt único e custo ajustável

Lento e com memória, de propósito: trava ataques por força bruta.



Salt

Valor aleatório único por password: impede *rainbow tables* e *hashes* iguais para passwords iguais.



Pepper

Segredo adicional guardado fora da base de dados, e isolados, por exemplo com tecnologia HSM ou de isolamento Seguro de armazenamento e processamento criptográfico



Custo

Tempo e memória ajustados ao hardware; rever os parâmetros periodicamente.

107

107

Hash, MAC e assinatura: diferenças

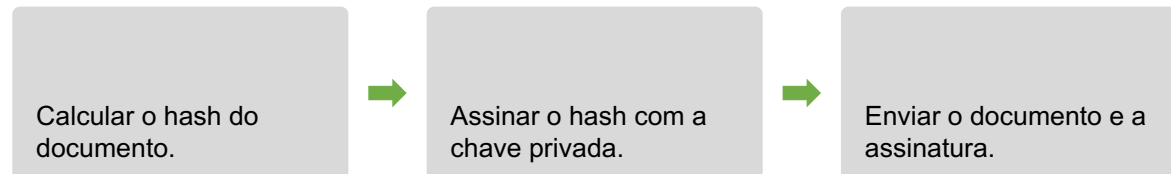
Mecanismo	Chave	Garante	Exemplos
Hash	Nenhuma	Integridade (qualquer pessoa recalcula)	Famílias SHA-2, SHA-3 Ex: SHA-256,
MAC / HMAC	Segredo partilhado	Integridade e autenticidade entre as partes	HMAC-SHA-256
Assinatura digital	Par público/privado	Integridade, autenticidade e não-repúdio	Ed25519, ECDSA, RSA-PSS
AEAD	Segredo partilhado	Confidencialidade e integridade	AES-GCM, ChaCha20-Poly1305

Um MAC não garante não-repúdio: os dois lados conhecem a chave: qualquer um poderia ter produzido a etiqueta.

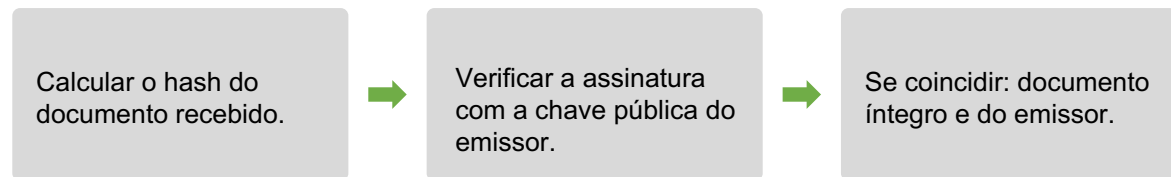
108

Assinaturas digitais

Assinar (emissor)



Verificar (destinatário)



PKI e certificação de chave pública

CA raiz
offline, protegida num HSM

↓ *assina*

CA intermédia
emite os certificados do dia a dia

↓ *assina*

Certificado do servidor
ex.: www.exemplo.pt

Confiar numa raiz é confiar em tudo o que ela assina e sucessivamente isso permite validar a cadeia de certificação.

O que contém um certificado X.509

Nome (domínio), chave pública, emissor, validade e assinatura da CA.

Como é validado

Cadeia até uma raiz confiável, validade, nome e revogação (CRL ou OCSP).

- CRL - *Certificate Revocation List*
- OCSP - OnLine Certificate Status Protocol

Boas práticas

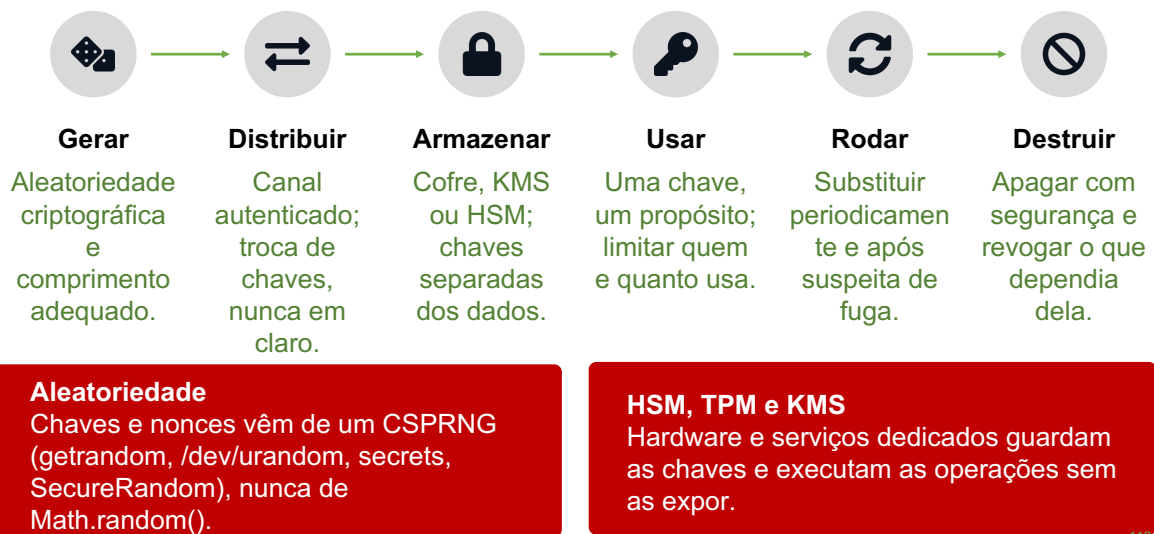
Uso de CAs de confiança

Que hoje possam gerir certificados de forma transparente e de revogação/actualizações para curta duração.

Outros aspetos complementares de detalhe no uso de métodos e algoritmos criptográficos

111

Ciclo de vida das chaves criptográficas



112

Em resumo (1) ...

Objetivos fundamentais

- **Confidencialidade:** só quem está autorizado consegue ler a informação.
- **Integridade:** garante que os dados não foram alterados.
- **Autenticidade:** confirma a identidade de quem envia ou acede.
- **Não repúdio:** impede que alguém negue ter praticado uma ação (assinaturas digitais)

Tipos principais e utilização

- **Criptografia Simétrica** (ex.: AES-256): uma única chave para cifrar e decifrar. É rápida e ideal para grandes volumes de dados, mas obriga a partilhar a chave de forma segura.
- **Criptografia Assimétrica** (ex.: RSA, ECC): par de chaves, pública e privada. Resolve a troca de chaves e permite assinaturas digitais, mas é mais lenta.
- **Funções de hash** (ex.: SHA-256, bcrypt, Argon2): transformação unidirecional, usada para verificar integridade e guardar palavras-passe.
- **MACs (HMACs e CMACs):** uso para provas de autenticidade rápida e integridade de dados
- **Híbrida:** combina as anteriores (é o que o TLS/HTTPS faz): assimétrica para trocar a chave, simétrica para os dados.

Em resumo (2) ...

Boas práticas essenciais

- Usar **algoritmos padrão, testados e escrutinados cientificamente**; nunca inventar ou modificar algoritmos.
- Evitar algoritmos obsoletos: MD5, SHA-1, DES, RC4
- Proteger as **chaves**: a segurança depende mais delas já que o algoritmo é conhecido quanto à sua especificação. Guardá-las em cofres digitais (HSM, KMS), nunca no código nem em repositórios e evitar a sua exposição em memória quando não sejam necessárias
- Fazer **rotação ou refrescamento periódico** de chaves e ter plano de revogação.
- Usar chaves com comprimento adequado para serem seguras (ex.: AES-256, RSA ≥ 2048 bits, preferencialmente 3072).
- Para palavras-passe, usar hash **com salt** e algoritmos lentos (Argon2, bcrypt, scrypt), e não de cifra reversível.
- Usar bibliotecas criptográficas reconhecidas como tendo implementações seguras dos algoritmos criptográficos disponibilizados

Em resumo (3) ...

Erros a evitar

- Confundir **codificação em formas de representação de dados** (ex., Base64) ou **ofuscação** com formas de transformação criptográfica
- Reutilizar chaves ou vetores de inicialização (IV/nonce) nas operações criptográficas
- Evitar formas de padding inseguras
 - Ex., em RSA usar RSA/OAEP, RSA/PSS) ou algoritmos que não precisam de utilizar padding
 - Usar criptografia simétrica com modos de processamento AEAD: AES/GCM ou ChaCha20/Poly1305
- Usar algoritmos de hash hoje considerados fracos (ex., MD5, SHA-1)
 - Adotar SHA-256, SHA-384, SHA-512 (nas famílias SHA-2 e SHA-3)
- Guardar a chave junto aos dados cifrados.
- Ignorar a validação adequada de certificados de chave pública
- Não ter cópia de segurança das chaves (perder a chave = perder os dados).

115

Técnicas criptográficas avançadas



Provas de conhecimento nulo

Provar que se sabe ou cumpre algo sem o revelar. Ex.: identidade e blockchains privadas.



Cifras homomórficas

Calcular sobre dados cifrados sem os decifrar. Ex.: análise de dados sensíveis na cloud.



MPC e partilha de segredos

Várias partes calculam em conjunto sem revelar dados; Shamir divide uma chave em partes.



Computação confidencial

Hardware isola os dados em uso (enclaves, TEE) e os dados em claro nunca são expostos for a dos enclaves (estando sempre cifrados em memória)



Tokenização, FPE

Substituir dados sensíveis por tokens de anonimização ou cifrar mantendo o formato. Ex.: cartões de pagamento.



Timestamping e cadeias de hash

Provar que um dado existia numa data e não foi alterado. Ex.: registos e blockchain.

116

Algoritmos aplicados e relevantes atualmente, seu propósito e exemplo de utilização

Função	Algoritmo(s)	Parâmetros	Usos típicos
Cifra simétrica	AES, ChaCha20	AES-128 ou 256, modo GCM	Cifra de Discos, TLS, VPN, bases de dados
Cifra autenticada	AES-GCM, ChaCha20-Poly1305	Nonce único por mensagem	TLS 1.3, SSH, WireGuard
Troca de chaves	ECDH (X25519, P-256), DH	Chaves efêmeras (ECDHE)	TLS, SSH, Signal
Cifra assimétrica	RSA-OAEP	3072 bits ou mais	Transporte de chaves (legado)
Assinaturas	Ed25519, ECDSA, RSA-PSS	P-256 ou superior; RSA $\geq$ 3072	Certificados, SSH, código
Hash	SHA-256, SHA-384, SHA-3, BLAKE2	256 bits ou mais	Integridade, assinaturas
MAC	HMAC-SHA-256, Poly1305	Chave de 128 bits ou mais	APIs, tokens, cookies
Passwords (KDF)	Argon2id, scrypt, bcrypt, PBKDF2	Custo ajustado ao hardware	Armazenar credenciais ¹¹⁷

117

“Recomendado” vs. “a evitar”

RECOMENDADO

- AES-256 ou AES-128 em modo GCM
- ChaCha20-Poly1305
- SHA-256, SHA-384, SHA-3
- X25519 ou ECDHE P-256
- Ed25519 ou ECDSA P-256+
- RSA $\geq$ 3072 (OAEP e PSS)
- Argon2id ou bcrypt
- TLS 1.3 (ou 1.2 bem configurado)

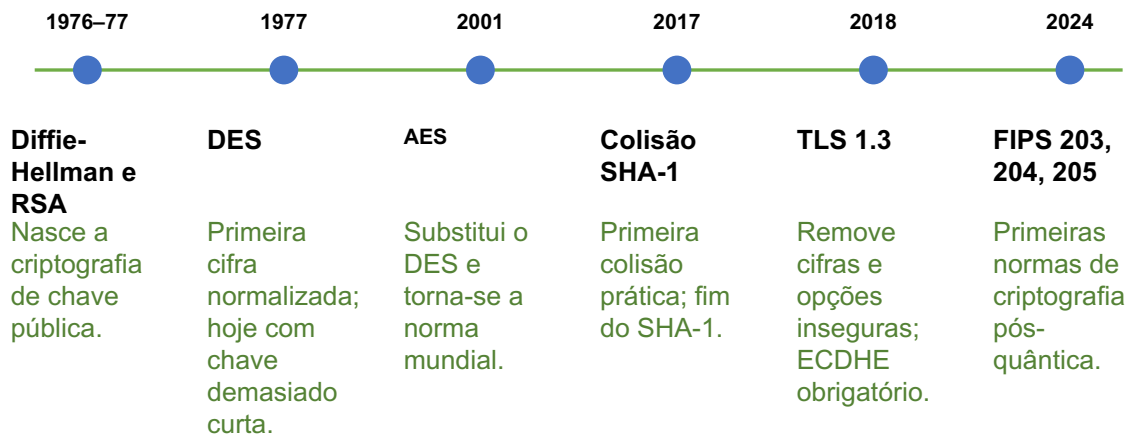
EVITAR (NÃO USAR)

- DES e 3DES
- RC4
- MD5 e SHA-1 (assinaturas e certificados)
- RSA ou DH com menos de 2048 bits
- ECB e CBC sem autenticação
- PKCS#1 v1.5 para cifra
- Versões SSL, TLS 1.0 e TLS 1.1
- Algoritmos caseiros (sem escrutínio científico)

As recomendações evoluem com a evolução das soluções e protocolos que os utilizam

118

Evolução...



Cada passo respondeu a estudos de criptanálise, a uma limitação ou a uma nova necessidade.

119

Utilizações relevantes? (Exemplos)



120

Exemplos de utilização de métodos criptográficos para proteção de dados em trânsito

Tecnologia	Para quê	Criptografia usada
TLS 1.3 (HTTPS)	Web, APIs e e-mail	ECDHE, AES-GCM ou ChaCha20, assinaturas de certificado
SSH	Administração remota	X25519, Ed25519, ChaCha20-Poly1305 ou AES-GCM
IPsec / IKEv2	VPN site-to-site e remota	(EC)DH, AES-GCM, SHA-2
WireGuard	VPN moderna e simples	Curve25519, ChaCha20-Poly1305, BLAKE2s
WPA3	Redes Wi-Fi	SAE (Dragonfly), AES-CCMP ou GCMP
Signal Protocol	Mensagens end-to-end	X3DH ou PQXDH, Double Ratchet, AES-256, HMAC

121

Proteção de Dados Armazenados

Dados
em claro

cifra com a DEK (AES-256-GCM)

DEK
chave de dados

cifrada pela KEK (no KMS, com
isolamento num HSM – Hardware
Security Module)

Disco completo
BitLocker,
FileVault

Base de dados
TDE: cifra
transparente

Aplicação
Cifra de campos e
colunas

**Backups e
objetos**
Cifra com chaves
do KMS

122

Erros comuns de criptografia



Modos Fracos ou Padding Fraco

Revela padrões ou permite recuperar texto limpo.



Chaves no código

Segredos em repositórios e imagens de contentores.



Criptografia fraca

Algoritmos próprios ou combinações improvisadas.



Aleatoriedade fraca

Induz previsibilidade



Hash rápido em passwords

MD5 ou SHA-256 simples permitem testes em massa.



Certificados sem control adequado de validação



Sem rotação nem revogação

Chaves eternas aumentam o impacto de uma fuga.



Cifrar sem autenticar

Sem integridade, o atacante altera o texto cifrado.

123

Criptografia atual em dados críticos



Em trânsito

TLS 1.3, VPN, SSH:
ECDHE, AES-GCM,
ChaCha20



Em repouso

AES-256 a executar com
KMS e em HSM.



Identidade

Assinaturas ECDSA,
Ed25519 e RSA;
certificados X.509.



Integridade

SHA-256, SHA-3, HMAC
e AEAD.

EM QUE ASSENTA A SEGURANÇA

Simétrica e hash

Força bruta: o custo cresce com o tamanho da chave.

Assimétrica

Problemas matemáticos difíceis: fatorização (RSA) e logaritmo discreto (DH, ECC).

É esta segunda base que um “futuro computador quântico” ameaça.

124

E o que esperar para a era pós-quântica ?

125

Resistência à ameaça quântica

Algoritmo	Tipo	Ataque quântico	Impacto	Resposta
RSA, DH, ECC	Assimétrica	Algoritmo de Shor	Quebrados	Substituir por PQC
AES-128	Simétrica	Algoritmo de Grover	≈ 64 bits efetivos	Preferir AES-256
AES-256	Simétrica	Algoritmo de Grover	≈ 128 bits: seguro	Manter
SHA-256, SHA-3	Hash	Grover (pré-imagem)	Margem reduzida	SHA-384+ a longo prazo

Ok, na prática ainda não existe um computador quântico capaz de quebrar RSA, DH, DSA ou ECC (Os estudos baseiam-se em análises projetadas de algoritmos ótimos, que possam quebrar a segurança (nas *hardness assumptions*) desses métodos quando executarem com suporte esperado de computação quântica)

Mas o movimento na concepção de novos algoritmos que os substituam já foi posto em marcha incerta, mas os dados de hoje podem ser interceptados já.

126

A transição para a criptografia quântica: *Harvest now, decrypt later*



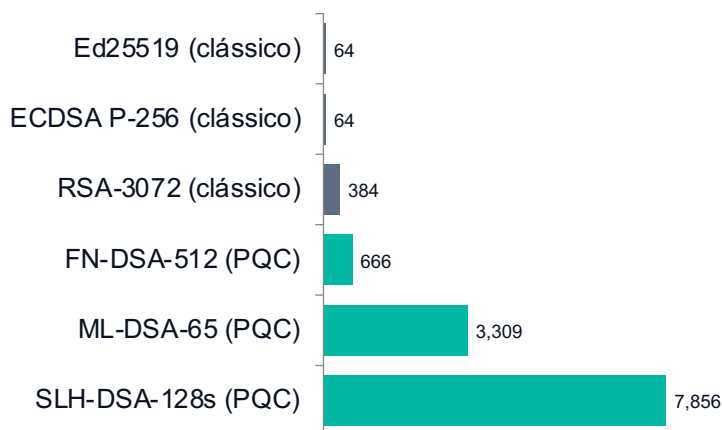
Normas pós-quânticas do NIST para algoritmos criptográficos pós-quânticos

Algoritmo (nome anterior)	Norma	Função	Família matemática	Estado
ML-KEM (CRYSTALS-Kyber)	FIPS 203	Troca de chaves (KEM)	Reticulados	Final (ago. 2024)
ML-DSA (CRYSTALS-Dilithium)	FIPS 204	Assinaturas	Reticulados	Final (ago. 2024)
SLH-DSA (SPHINCS+)	FIPS 205	Assinaturas	Funções de hash	Final (ago. 2024)
FN-DSA (FALCON)	FIPS 206	Assinaturas compactas	Reticulados (NTRU)	Em rascunho
HQC	A atribuir	KEM de reserva	Códigos corretores de erros	Selecionado (mar. 2025)

Estado em outubro de 2026. FN-DSA e HQC ainda não têm norma final.

O “custo” computacional dos algoritmos pós-quânticos

Tamanho típico da assinatura (bytes)



Reticulados

ML-KEM e ML-DSA: rápidos, com chaves de 1 a 2 KB e assinaturas de cerca de 3 KB.

Baseadas em hash

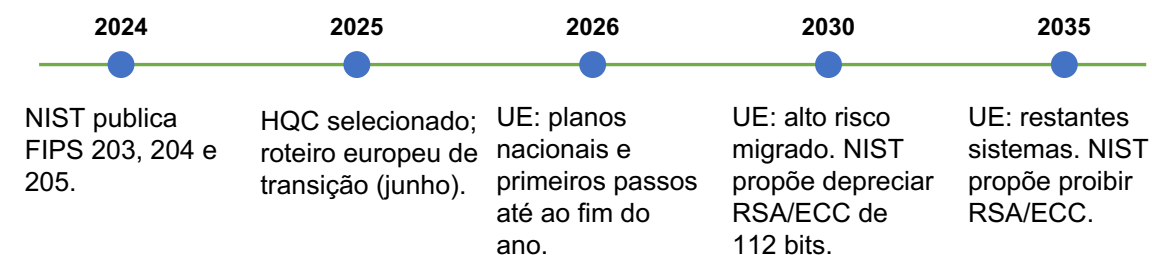
SLH-DSA: hipóteses de segurança muito conservadoras, mas assinaturas maiores.

Impacto prático

Handshakes TLS e cadeias de certificados maiores; atenção a dispositivos limitados.

129

Migração para a era pós-quântica



Datas NIST: proposta do IR 8547 (versão preliminar). Datas UE: roteiro do Grupo de Cooperação NIS (junho de 2025).



Híbrido

Combinar clássico e PQC (ex.: X25519MLKEM768 em TLS 1.3): seguro se um dos dois resistir.



Cripto-agilidade

Poder trocar algoritmos sem reescrever sistemas.



Inventário (CBOM)

Descobrir onde se usa RSA e ECC: bibliotecas, certificados, HSM e firmware.



Já em produção

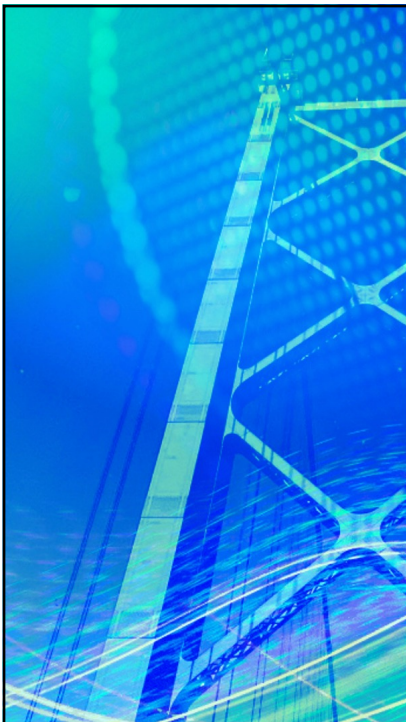
Browsers e CDN (híbrido), OpenSSH, Signal (PQXDH) e iMessage (PQ3).

130

Sugestões de Bibliografia

W. Stallings, L. Brown, Computer Security – Principles and Practice, Person

W. Stallings, Cryptography and Network Security – Prentice Hall, Person



4. Segurança do Software e Bases de Dados

Visão Resumo

Pontos chave a reter neste bloco

- 1** Segurança do software começa no desenho (Security by Design): modelar ameaças, codificar com segurança, testar e controlar as dependências.
- 2** As bases de dados concentram o valor: proteger por camadas, cifrar, auditar e garantir a integridade. Servidores de Bases de Dados como Servidores Críticos
- 3** Operar a BD com segurança: hardening, rede privada, backups testados e registos protegidos.
- 4** Decidir acessos com menor privilégio, combinando RBAC como base com ABAC para regras contextuais (modelos que são hoje suportados em SGBDs)
- 5** Identidade forte: MFA resistente a phishing, passkeys, SSO com OIDC e acessos privilegiados controlados.

Tópicos nesta parte



Segurança do software

Vulnerabilidades, OWASP e cadeia de fornecimento



Desenvolvimento de Software seguro

SSDF, ameaças, codificação e testes



Segurança de Bases de dados

Ameaças, camadas, proteção e integridade



Operação Segura de BD

Hardening, rede, auditoria e backups



Controlo de acessos

Modelos DAC, MAC, RBAC, ABAC e BD



Gestão de Identidades e autenticação

IAM, MFA, passkeys, SSO e tecnologias

Objetivo da segurança do software: evitar vulnerabilidades em SW (por concepção)



Defeito

Erro de código ou de desenho.



Vulnerabilidade

Defeito que pode ser explorado com impacto na segurança.



Exploit

Técnica ou código que explora a vulnerabilidade.



Incidente

Dano real: fuga de dados, indisponibilidade, fraude.

Dificuldades a endereçar

Complexidade, código legado, dependências de terceiros, pressão de prazos e linguagens sem memória segura.

Objetivo: *security by design*

Software que resiste a ataques, deteta abusos e tem mecanismos de recuperação com segurança estabelecida desde a concepção

Exemplificação de classes de vulnerabilidades

Classe	Exemplo (CWE)	Como acontece	Defesa principal
Memória	Escrita fora de limites da memória dos processos (CWE-787)	Escrever para lá do fim de um buffer na dimensão estabelecida	Linguagens com memória segura
Injeção	SQL injection (CWE-89)	Input interpretado como código	Querys “sanitizadas”
Scripting entre sites	XSS (CWE-79)	Input não codificado numa página	Codificar o output, CSP
Controlo de acessos	Autorização em falta (CWE-862)	O servidor não verifica permissões	Autorização em cada pedido
Falsificação de pedidos	CSRF (CWE-352)	O browser envia um pedido não pretendido	Tokens anti-CSRF, SameSite
Travessia de diretórios	Path traversal (CWE-22)	Caminhos manipulados para acesso indevido a ficheiros	Normalizar e restringir caminhos

<https://cwe.mitre.org/>

137

Ciclo de desenvolvimento Seguro de serviços

Requisitos	Desenho	Implementação	Testes	Entrega	Operação
- Requisitos de segurança - Classificação dos dados - Casos de abuso	- Threat modeling (STRIDE) - Revisão de arquitetura - Princípios de design seguro	- Codificação segura - Revisão de código - Deteção de segredos	- SAST e SCA - DAST e pentest - Testes automáticos	- Configuração segura - Artefactos assinados - Aprovação de segurança	- Monitorização - Correção de falhas - Revisão pós-incidente

Shift left: quanto mais cedo se corrige uma falha ou vulnerabilidade melhor ... e mais barato é.

138

Princípios de codificação segura

✓ Validar todo o input

Nunca confiar no cliente: validar tipo, tamanho e formato no servidor.

⌘ Codificar o output

Escapar os dados consoante o contexto (HTML, SQL, shell).

🔑 Autenticação e sessões fortes

Hash de passwords (Argon2, bcrypt), cookies seguros e expiração.

👤 Controlo de acessos no servidor

Verificar autorização em cada pedido, por objeto e por função.

🔑 Segredos fora do código

Chaves e passwords em cofres, nunca em repositórios.

🔒 Criptografia padrão

Bibliotecas e algoritmos reconhecidos; nunca inventar a própria.

⚠ Erros sem fugas de informação

Mensagens genéricas ao utilizador; detalhes só nos registos.

🔗 Dependências sob controlo

Inventário de versões fixas e atualização regular.

139

OWASP Top 10:2025 em foco

A01 **Controlo de acessos quebrado**
Utilizadores a aceder ao que não deviam

A02 **Má configuração de segurança**
Defaults inseguros e permissões a mais

A03 **Cadeia de fornecimento de software**
Dependências e builds comprometidos **NOVO**

A04 **Falhas criptográficas**
Cifra fraca ou mal aplicada

A05 **Injeção**
SQL, comandos, XSS: dados tratados como código

A06 **Design inseguro**
Falhas de conceção, não só de código

A07 **Falhas de autenticação**
Passwords fracas e sessões mal geridas

A08 **Integridade de software e dados**
Atualizações e dados não verificados

A09 **Falhas de registo e alertas**
Ataques sem deteção nem resposta

A10 **Condições excecionais mal tratadas**
Erros mal tratados que abrem falhas **NOVO**

140

OWASP Top 10: 2025 e defesas

A01	Controlo de acessos quebrado Autorização no servidor em cada pedido		A06	Design inseguro Modelação de ameaças e padrões seguros	
A02	Má configuração de segurança Baselines, hardening e revisão automática		A07	Falhas de autenticação MFA, passkeys e gestão de sessões	
A03	Cadeia de fornecimento SBOM, SCA e artefactos assinados	NOVO	A08	Integridade de software e dados Assinaturas e verificação de atualizações	
A04	Falhas criptográficas Algoritmos padrão, TLS e gestão de chaves		A09	Falhas de registo e alertas Registos úteis, alertas e resposta	
A05	Injeção Consultas parametrizadas e codificação		A10	Condições excecionais Falhar de forma segura; tratar todos os erros	NOVO

141

Segurança de APIs



Autenticação forte

OAuth 2.0, FIDO2, ou OpenID Connect, com tokens de curta duração.



Autorização por objeto

Validar a propriedade do recurso em cada pedido.



Limitar e validar

Rate limiting, quotas, validação de schema (OpenAPI) e tamanhos máximos.



TLS (1.2 ou se possível 1.3) em todo o lado

HTTPS obrigatório; TLS memo entre serviços internos.



API gateway

Ponto único de entrada: autenticação, limites, registo e WAF.



Inventário e versões

Conhecer todas as APIs e desativar as versões obsoletas.

142

Linguagens de programação e gestão segura de memória

Linguagens com memória segura

Rust, Go, Java, C#, Python ou Swift gerem a memória de forma automática ou verificada. Eliminam classes inteiras de erros: buffer overflow, use-after-free ou leituras fora dos limites.

C e C++

Desempenho e controlo total, mas o programador gere a memória. Reduzir o risco com “sanitizers”, “fuzzing”, revisão de código e mitigações na defesas do sistema operativo.

ASLR

Aleatoriza os endereços de memória.

DEP / NX

Memória isolada de dados não executável.

Stack canaries

Deteta a corrupção da pilha.

CFI

Restringe o fluxo de execução.

Recomendação: usar código novo e linguagens com memória segura sempre que possível.

Aspetos complementares

As técnicas (1)

ASLR - Address Space Layout Randomization

Aleatoriza os endereços de memória, oferece **imprevisibilidade dos endereços de memória**

Endereços diferentes a cada execução: a stack, a heap, as bibliotecas e o próprio executável são carregados em posições aleatórias

Aplicada pelo sistema operativo, com impacto de desempenho praticamente nulo.

DEP / NX - Data Execution Prevention / **No-eXecute**

Memória isolada de dados não executável.

páginas de memória como a stack e a heap são marcadas como não executáveis, e o CPU recusa-se a executar instruções que lá estejam
roteção ao nível do hardware

CFI - Control-Flow Integrity

Restringe o fluxo de execução.

funciona em duas fases: primeiro **define o que é permitido (com base num grafo de fluxo de execução correta)** e depois **verifica em tempo de execução** que o programa cumpre essas regras.

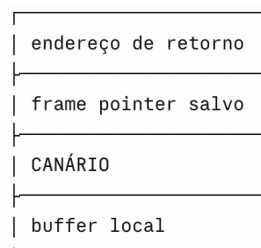
As técnicas (2)

STACK CANARY: Um **stack canary** (canário de stack) é um **valor secreto colocado na stack, entre os buffers locais e os dados de controlo** (o endereço de retorno e o frame pointer guardado). Se o canário for alterado, há um overflow e provoca-se uma exceção

Início da função (prólogo): o compilador insere código que copia o valor do canário (um valor global aleatório, gerado no arranque do processo) para a stack, logo abaixo do endereço de retorno.

1.Durante a função: o código executa normalmente. Se houver um buffer overflow, os bytes escritos para além do buffer **têm de atravessar o canário** antes de chegar ao endereço de retorno.

2.Fim da função (epílogo): antes de executar o `ret`, o código compara o canário na stack com o valor original.



Gestão da cadeia de fornecimento de software



Dependências vulneráveis

Uma biblioteca com falha afeta todos os que a usam. Exemplo: Log4Shell.



Pacotes maliciosos

Typosquatting e dependency confusion levam a instalar código do atacante.



Build ou distribuição comprometidos

O código é alterado antes de chegar ao cliente. Exemplos: SolarWinds e xz Utils.

DEFESAS

- SBOM: Inventário dos componentes de cada aplicação na análise de código
- SCA (Software Composition Analysis) e versões fixadas (lockfiles)
- Artefactos assinados e proveniência verificável
- Repositório interno mantido em segurança e revisão de novas dependências
- MFA e contas protegidas de publicação de pacotes de SW para instalação e uso

SBOM; a formal, machine-readable inventory list that acts like a "nutrition label" or recipe for an application. It details all the building blocks inside a software package

Desenvolvimento seguro: o NIST SSDF SP 800-218. Secure Software Development Framework

PO

Prepare the Organization

Preparar a organização

Políticas, papéis, formação e ferramentas de segurança.

PS

Protect the Software

Proteger o software

Controlar o acesso ao código e garantir a integridade dos artefactos.

PW

Produce Well-Secured Software

Produzir software seguro

Desenho, revisão de código, testes e codificação segura.

RV

Respond to Vulnerabilities

Responder a vulnerabilidades

Identificar, corrigir e analisar a causa de cada falha.

Modelos de referência a ter em conta:

NIST SSDF (SP 800-218), OWASP SAMM e Microsoft SDL.

Modelação de ameaças: STRIDE

Ameaça	Propriedade violada	Exemplo	Mitigação
Spoofing (falsificação)	Autenticação	Usar credenciais roubadas	MFA, passkeys
Tampering (adulteração)	Integridade	Alterar dados em trânsito ou na BD	Assinaturas, TLS, controlo de acessos
Repudiation (repúdio)	Não-repúdio	Negar uma operação realizada	Registos auditáveis, assinaturas
Information disclosure	Confidencialidade	Expor dados numa API	Cifra, autorização, minimização
Denial of service	Disponibilidade	Sobrecarregar um serviço	Rate limiting, escalabilidade
Elevation of privilege	Autorização	Utilizador comum passa a admin	Menor privilégio, validação no servidor

Modelar ameaças: desenhar o fluxo de dados, marcar as fronteiras de confiança e aplicar o STRIDE a cada elemento.

149

Testes de segurança do SW

Técnica	Quando no ciclo de desenvolvimento ?	O que encontra ?	Limites
SAST	Codificação e CI	Padrões inseguros no código-fonte	Falsos positivos; sem contexto de execução
SCA	Build	Dependências com CVE e licenças	Só conhece vulnerabilidades já publicadas
DAST	Teste e staging	Falhas na aplicação em execução	Cobertura limitada; mais lento
IAST	Teste	Instrumenta a aplicação durante os testes	Exige agente; depende da cobertura dos testes
Fuzzing	Teste	Entradas inesperadas que causam falhas	Precisa de tempo e de bons casos de teste
Pentest e revisão manual	Antes do lançamento	Falhas de lógica e cadeias de ataque	Pontual e dispendioso

150

Termos das técnicas e tipologias de ferramentas

SCA - Software Composition Analysis, <https://jfrog.com/learn/sdlc/sca/>

SAST: [Static Application Security Testing](#)

Analisa o código-fonte com métodos de análise estática antes da aplicação ser executada (de dentro para fora, no início do desenvolvimento)

DAST . Dynamic Application Security Testing

DAST (Dynamic Application Security Testing):

Testa a aplicação em execução (de fora para dentro, sem código-fonte), <https://www.opentext.com/what-is/dast>

IAST – Interactive Application Security Testing., <https://www.harness.io/blog/what-is-interactive-application-security-testing-iaast>

FUZZING: (ou teste de fuzz) é uma técnica de teste de software automatizada ou semiautomatizada que consiste em enviar uma grande quantidade de dados inválidos, inesperados ou aleatórios para as entradas de um programa, <https://pt.wikipedia.org/wiki/Fuzzing>

Ameaças e vulnerabilidades a defender no acesso a bases de dados

Injeção SQL

Manipular consultas para ler ou alterar dados.

Credenciais fracas

Passwords simples ou por defeito em contas da BD.

Privilégios excessivos

Contas com mais acesso do que precisam.

Exposição à internet

Portas da BD acessíveis publicamente, por vezes sem autenticação.

Má configuração

Defaults inseguros e funcionalidades desnecessárias.

Backups desprotegidos

Cópias sem cifra em locais acessíveis.

Ameaças internas

Utilizadores legítimos que abusam do acesso.

Ransomware e exfiltração

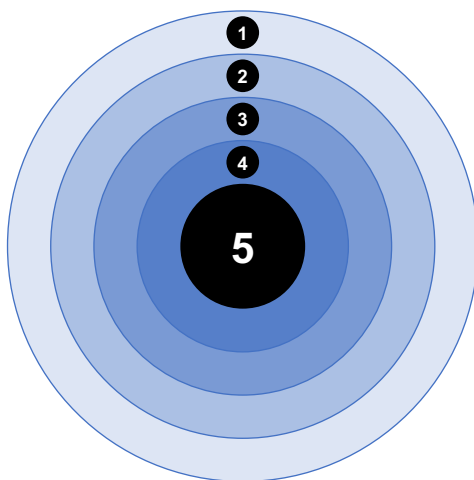
Cifrar ou roubar dados para extorsão.

Bases de Dados

Servidores de Bases de Dados:

- Podem ter que ser vistos como servidores críticos (devido à relevância dos dados que gerem)
- Aplicar desde logo todas as preocupações de segurança como no ponto em que se abordou a segurança de servidores

Defesa em profundidade para bases de dados



- 1 Rede e perímetro**
Sub-rede privada, firewall, sem exposição pública
- 2 Servidor e sistema operativo**
Hardening, patches, cifra de disco
- 3 Instância da base de dados**
Configuração segura, contas, auditoria
- 4 Objetos e permissões**
Privilégios mínimos, RLS, mascaramento
- 5 Dados**
Cifra, tokenização, backups protegidos

Proteção dos dados mantidos em BD

Técnica	O que faz	Protege contra	Limites
Cifra de dados armazenados (ou em repouso) - TDE	Cifra os ficheiros e backups da BD	Roubo de discos e backups	Não protege de utilizadores com acesso à BD e ações incorretas
Cifra de coluna ou na aplicação	Cifra campos sensíveis com chaves próprias	Acesso de DBAs e fugas na BD	Pesquisa e indexação limitadas
Cifra em trânsito (TLS)	Protege as ligações cliente-BD	Interceção e man-in-the-middle	Só protege o canal de comunicação no acesso
Mascaramento	Ocultar valores a quem não precisa	Visualização indevida	Estático (cópias) ou dinâmico (consultas)
Tokenização	Substitui dados por tokens que podem ter transformações criptográficas	Fuga de dados como cartões	Requer um cofre de tokens
Pseudonimização e anonimização	Remove ou substitui identificadores	Reidentificação (RGPD)	A anonimização verdadeira é irreversível

Ambientes de teste: usar dados mascarados ou sintéticos, nunca dados reais.

155

155

Integridade dos dados no modelo ACID

A Atomicidade

A transação executa-se por inteiro ou não se executa.



Restrições de integridade

Chaves primárias e estrangeiras, CHECK NOT NULL e tipos de dados corretos.

C Consistência

Os dados respeitam sempre as regras definidas.



Deteção de alterações

Triggers, registos de alterações, tabelas só de acréscimo (ledger) e hashes.

I Isolamento

Transações concorrentes não interferem entre si.



Backups verificados

Checksums, restauros testados e separação entre quem altera e quem audita.

D Durabilidade

Depois de confirmados, os dados persistem.

Integridade: dados corretos, completos e só alterados por quem tem autorização.

ECUTIVE
UCATION
156

SCIENCE & TECHNOLOGY

156

Hardening de servidores de BD



Versão suportada

Usar versões com suporte e aplicar os patches de segurança.



Remover o que não é preciso

Contas, bases de exemplo e funcionalidades sem uso.



Desativar funções perigosas

Execução de comandos do SO, leitura de ficheiros locais e extensões.



TLS (1.2, 1.3) e autenticação obrigatórios

Sem ligações em claro nem acesso anónimo.



Contas de serviço mínimas

O processo da BD corre sem privilégios administrativos no SO.



Benchmarks CIS

Baselines de configuração por SGBD, verificadas automaticamente.

157

Exposição de rede da base de dados (Exemplos, Portas Expostas)

Porta	Serviço	Nota
3306	MySQL / MariaDB	Nunca exposto à internet
5432	PostgreSQL	Limitar origens (pg_hba.conf)
1433	SQL Server	Alvo frequente de força bruta
1521	Oracle	Listener com acesso restrito
27017	MongoDB	Ativar autenticação e TLS
6379	Redis	Exigir palavra-passe e rede privada
9200	Elasticsearch	Exposição pública causou fugas

Sub-rede privada

A BD só aceita ligações da camada de aplicação.

Administração como sistemas bastião e/ou com acesso VPN

Nunca diretamente da internet ou sem defesa de perímetro

Firewall por origem

Regras mínimas: deny by default.

158

Ataques a Bases de Dados

SQLi (SQL Injection)

Ataques por Inferência

Ataques à confidencialidade dos dados

Aspetos complementares de detalhe sobre segurança de Bases de Dados

Ataques a Bases de Dados: SQLi

SQLi (SQL Injection)

Uma das ameaças de segurança baseadas em rede mais prevalentes e perigosas

Concebida para explorar a natureza das páginas de aplicações Web

Envia comandos SQL maliciosos para o servidor de bases de dados

Objetivo do ataque

O objetivo de ataque mais comum é a extração em massa de dados

Dependendo do ambiente, a injeção de SQL também pode ser explorada para:

- modificar ou eliminar dados
- Executar comandos arbitrários do sistema operativo
- Lançar ataques de negação de serviço (DoS)

Como funciona o ataque SQLi

O ataque de SQLi funciona tipicamente através da terminação prematura de uma cadeia de texto e da adição de um novo comando

Como o comando inserido pode ter strings adicionais acrescentadas antes de ser executado, o atacante termina a string injetada com uma marca de comentário “- -”



O texto subsequente é ignorado em tempo de execução

Tipologias SQLi

Entrada do utilizador - Os atacantes injetam comandos SQL fornecendo dados de entrada do utilizador devidamente construídos

Variáveis do servidor - Os atacantes podem forjar os valores colocados nos cabeçalhos HTTP e de rede e explorar esta vulnerabilidade inserindo dados diretamente nos cabeçalhos

Injeção de segunda ordem - Um utilizador malicioso pode aproveitar dados já presentes no sistema ou na base de dados para desencadear um ataque de injeção de SQL, de modo que, quando o ataque ocorre, a entrada que modifica a consulta para causar o ataque não provém do utilizador, mas do próprio sistema

Cookies - Um atacante pode alterar cookies de forma a que, quando o servidor de aplicações constrói uma consulta SQL com base no conteúdo do cookie, a estrutura e a função da consulta sejam modificadas

Entrada física do utilizador - Aplicação de dados de entrada do utilizador que constroem um ataque fora do âmbito dos pedidos web

Ataques InBand vs. Outband

InBand: Utiliza o mesmo canal de comunicação para injetar código SQL e obter os resultados. Os dados obtidos são apresentados diretamente na página Web da aplicação.

OutBand: Utiliza um canal diferente para obtenção do resultado (ex., EMAIL)

Inclui diferentes tipos

Tautologias

Esta forma de ataque injeta código numa ou mais instruções condicionais, de modo a que sejam sempre avaliadas como verdadeiras

End-of-line comments

Depois de injetar código num determinado campo, o código legítimo que se segue é anulado através da utilização de comentários de fim de linha

Piggybacked queries

O atacante acrescenta consultas adicionais para além da consulta pretendida, aproveitando um pedido legítimo para transportar o ataque

Contramedidas contra SQLi

Práticas de codificação defensivas: Validação de Inputs (INPUT TYPE CHECKING)

Inserções de *queries* parametrizadas para filtragem de *queries* incorretas

SQL-DOM: uso de classes para automação de controlo de tipos de dados

Deteção com SIGNATURE_BASED: de formatos de *queries* específicas consideradas incorretas

Deteção anómala: de formatos de *queries* não consideradas corretas

Análise de código: usando testes de injeção de SQLi e verificação de respostas obtidas

Controlo de acesso em Bases de Dados

Dois comandos para gerir direitos de acesso:

Grant

Utilizado para conceder um ou mais direitos de acesso ou para atribuir um utilizador a uma função

Revoke

Revoga os direitos de acesso

Os direitos de acesso típicos são:

Select Insert Update Delete References

Ex: GRANT SELECT ON ANY TABLE TO ricflair

Ex: REVOKE SELECT ON ANY TABLE FROM ricflair

Controlo de acesso

GRANT { privileges | role }
[ON table]
TO { user | role | PUBLIC }
[IDENTIFIED BY password]
[WITH GRANT OPTION]

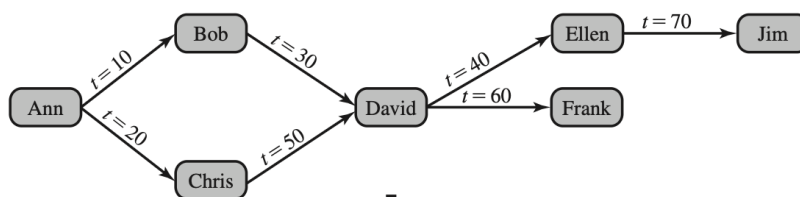
Ex: GRANT SELECT ON ANY TABLE TO ricflair

REVOKE { privileges | role }
[ON table]
FROM { user | role | PUBLIC }

Ex: REVOKE SELECT ON ANY TABLE FROM ricflair

167

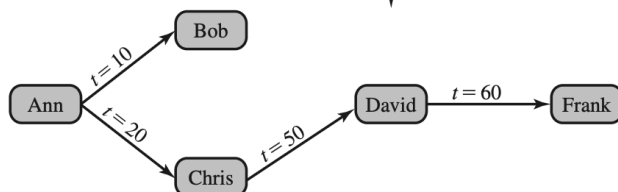
Controlo de acesso nas cadeias de autorizações ou revogações



Exemplo de Autorizações de Acesso



Exemplo de Revogação feita por Bob



168

Controlo de acessos com modelo RBAC (Ex.)

Role	Permissions
Fixed Server Roles	
sysadmin	Can perform any activity in SQL Server and have complete control over all database functions
serveradmin	Can set server-wide configuration options, shut down the server
setupadmin	Can manage linked servers and startup procedures
securityadmin	Can manage logins and CREATE DATABASE permissions, also read error logs and change passwords
processadmin	Can manage processes running in SQL Server
dbcreator	Can create, alter, and drop databases
diskadmin	Can manage disk files
bulkadmin	Can execute BULK INSERT statements
Fixed Database Roles	
db_owner	Has all permissions in the database
db_accessadmin	Can add or remove user IDs
db_datareader	Can select all data from any user table in the database
db_datawriter	Can modify any data in any user table in the database
db_ddladmin	Can issue all Data Definition Language (DDL) statements
db_securityadmin	Can manage all permissions, object ownerships, roles and role memberships
db_backupoperator	Can issue DBCC, CHECKPOINT, and BACKUP statements
db_denydatareader	Can deny permission to select data in the database
db_denydatawriter	Can deny permission to change data in the database

Fixed Roles:
Microsoft SQL Server

169

Ataques de inferência

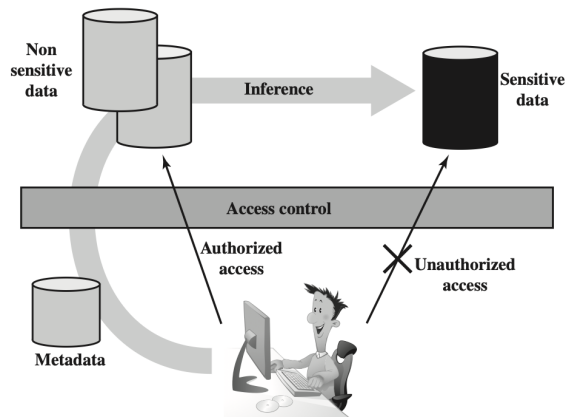
Não há uma transferência real de dados, mas o atacante consegue reconstruir a informação enviando pedidos específicos e observando o comportamento resultante do servidor do website/base de dados

Inclui: Consultas ilegais/logicamente incorretas

- Este ataque permite a um atacante recolher informações importantes sobre o tipo e a estrutura da base de dados de back-end de uma aplicação Web
- O ataque é considerado um passo preliminar de recolha de informação para outros ataques SQL injection cega (Blind SQL injection)
- Permite aos atacantes inferir os dados presentes num sistema de base de dados, mesmo quando o sistema é suficientemente seguro para não devolver ao atacante qualquer informação de erro

170

Ataques de Inferência



Item	Availability	Cost (\$)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware
Cake pan	online only	12.99	housewares
Shower/tub cleaner	in-store/online	11.99	housewares
Rolling pin	in-store/online	10.99	housewares

(a) Inventory table

Availability	Cost (\$)
in-store/online	7.99
online only	5.49
in-store/online	104.99

Item	Department
Shelf support	hardware
Lid support	hardware
Decorative chain	hardware

(b) Two views

Item	Availability	Cost (\$)	Department
Shelf support	in-store/online	7.99	hardware
Lid support	online only	5.49	hardware
Decorative chain	in-store/online	104.99	hardware

(c) Table derived from combining query answers

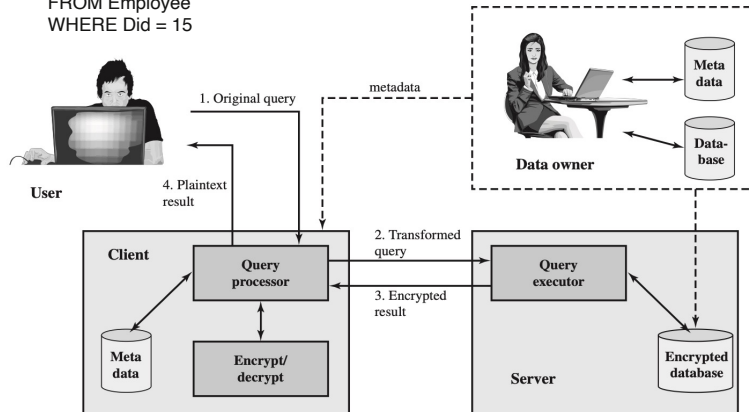
Contramedidas contra ataques por inferência

Deteção na modelação para conceção da Base de Dados

Deteção em tempo de *querying* (com análise de sequências de queries)

Bases de dados cifradas

```
SELECT Ename, Eid, Ephone
FROM Employee
WHERE Did = 15
```



```
SELECT Ename, Eid, Ephone
FROM Employee
WHERE Did = 1000110111001110
```

- O utilizador emite uma consulta SQL para obter campos de um ou mais registos com um valor específico da chave primária.
- O processador de consultas no cliente cifra a chave primária, modifica a consulta SQL em conformidade e transmite a consulta ao servidor.
- O servidor processa a consulta utilizando o valor cifrado da chave primária e devolve o registo ou registos correspondentes.
- O processador de consultas decifra os dados e devolve os resultados.

Auditoria e monitorização na Gestão de Bases de Dados

O que auditar

- Autenticações, com sucesso e com falha
- Alterações de privilégios e papéis
- Alterações de esquema (DDL)
- Acessos a dados sensíveis

Como

- Auditoria nativa: pgAudit, SQL Server Audit, Oracle Unified Auditing, MySQL Enterprise Audit
- Registos enviados para um SIEM

Deteção

- Database Activity Monitoring (DAM)
- Consultas em massa ou fora de horas
- Novos privilégios e contas
- Alertas e resposta

Proteger os registos de operações:

enviados para fora da BD e inalteráveis por quem a administra.

Backups e recuperação



Regra 3-2-1

Três cópias, dois suportes e uma offline ou imutável.



Backups cifrados

Cifrar as cópias e guardar as chaves separadamente.



Point-in-time recovery

Restaurar a um instante exato, antes do erro ou ataque.



Restausos testados

Testar regularmente e medir RPO e RTO reais.

Atenção

Replicação não é backup: por exemplo, o apagamento de dados ou a sua cifra por *ransomware* replicam-se de imediato para as réplicas. Devem manter-se cópias independentes e imutáveis.

Controlo de acesso: conceitos e princípios



Identificação

Controlar a identificação de quem diz ser.



Autenticação

Ter prova que é quem diz.



Autorização

O que pode fazer.



Auditoria

Registo do que fez. (como autorizado)

Menor privilégio

Só o acesso estritamente necessário.

Need-to-know

Só a informação necessária à função.

Separação de funções

Ninguém controla sozinho um processo crítico.

Deny by default

Tudo o que não é permitido é negado.

Revisão periódica

Recertificar os acessos com regularidade.

Modelos de controlo de acessos e considerar a relevância dos modelos

Modelo	Quem decide	Como funciona	Exemplos
DAC (discricionário)	O dono do recurso	O dono concede permissões a outros (ACL)	chmod, GRANT em SQL, partilhas
MAC (obrigatório)	A política do sistema	Rótulos de classificação e autorizações	SELinux, sistemas classificados
RBAC (papéis)	Os papéis atribuídos	Permissões agrupadas em papéis	Papéis em BD, Active Directory
ABAC (atributos)	Políticas sobre atributos	Utilizador, recurso e contexto	OPA, AWS Cedar, XACML
ReBAC (relações)	Relações entre entidades	Quem se relaciona com o quê	OpenFGA, Google Zanzibar

Na prática combinam-se: RBAC como base e ABAC para regras contextuais.

Controlo de acessos em bases de dados no modelo RBAC

PAPÉIS E PRIVILÉGIOS (RBAC)

```
CREATE ROLE analista;
GRANT SELECT ON vendas_resumo
TO analista;
REVOKE ALL ON clientes
FROM analista;
-- Conta da aplicação: mínimo
GRANT SELECT, INSERT
ON encomendas TO app_loja;
```

Princípios

Contas nominais, papéis em vez de privilégios diretos, contas de aplicação separadas e revisão periódica.

SEGURANÇA AO NÍVEL DA LINHA (POSTGRESQL)

```
ALTER TABLE doentes
ENABLE ROW LEVEL SECURITY;

CREATE POLICY ver_proprios
ON doentes
USING (medico = current_user);

-- Cada médico só vê os seus doentes
```

Outros mecanismos

Privilégios por coluna, views, procedimentos armazenados, mascaramento dinâmico e VPD

Exemplificação do modelo RBAC detalhe

Utilizadores	Papéis	Permissões
Ana	Médico	Ler e escrever processo
Rui	Enfermeiro	Ler processo
Marta	Administrativo	Marcar consultas

Vantagens Administração simples e auditável: mudar de função é mudar de papel.	Cuidados Explosão de papéis e papéis demasiado amplos acabam por violar o menor privilégio.	Separação de funções Impedir papéis incompatíveis, como quem cria e quem aprova a mesma operação.
--	---	---

179

ABAC e políticas dinâmicas

POLÍTICA (PSEUDOCÓDIGO)

```

permitir se:
  utilizador.departamento ==
    recurso.departamento
  e utilizador.nivel >=
    recurso.classificacao
  e dispositivo.gerido == verdadeiro
  e hora entre 08:00 e 20:00
senão: negar

```

RBAC vs. ABAC

O RBAC é simples e previsível.
O ABAC é flexível e decide em tempo real com atributos e contexto, mas as políticas são mais complexas de escrever e de testar.

PEP

Intercepta o pedido e aplica a decisão.

PDP

Avalia as políticas e decide.

PIP

Fornecer atributos do utilizador, recurso e contexto.

PAP

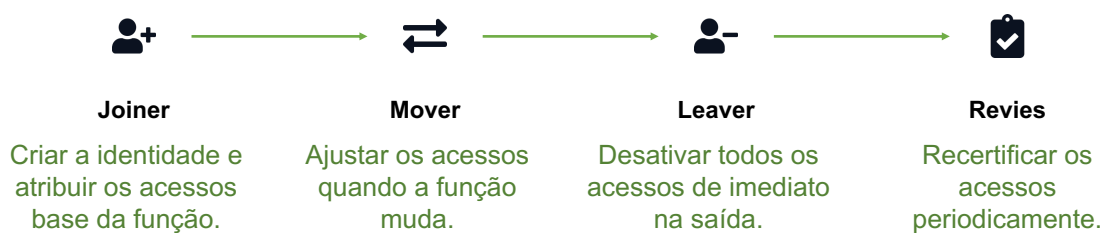
Onde as políticas são escritas e geridas.

180

Gestão IAM (Identity and Access Management)

181

Ciclo de Vida na Gestão de Identidades Digitais



Fonte de verdade e diretórios

RH como origem; diretório ou IdP (Active Directory, Entra ID, LDAP) e provisionamento automático com SCIM.

Governança e identidades de máquinas

Pedidos e aprovações, recertificação e detecção de acessos órfãos. Contas de serviço e chaves de API também têm dono e ciclo de vida.

182

Fatores de autenticação

Passwords (Passwords Fracas, Ataques a Passwords e Segurança de Passwords)

Autenticação *Passwordless* (ou sem passwords)

Autenticação com dispositivos usados em protocolos de desafio/resposta

Autenticação Biométrica

Autenticação com fatores comportamentais (ou “biometria comportamental”)

Autenticação com fatores criptográficos (provas de assinaturas digitais)

Autenticação multifatorial (duplo fator ou mais fatores)

Autenticação e Autenticação Multi-Fatorial (MFA)

Fatores: algo que sabes (segredo partilhado) · algo que tens (dispositivo) · algo que és (biometria)

Método	Resistente a phishing?	Notas
Palavra-passe	Não	Valorizar o comprimento; verificar contra listas de fugas; sem rotação forçada
OTP por SMS	Não	Vulnerável a SIM swap e interceção
TOTP (app autenticadora)	Não	Melhor do que SMS; pode sofrer phishing em tempo real
Notificação push	Parcial	Usar number matching; atenção à fadiga de MFA
Chave FIDO2 / passkey	Sim	Ligada ao domínio; sem segredo partilhado
Cartão inteligente / certificado	Sim	Cartão de Cidadão, PKI empresarial

Passkeys e FIDO2/WebAuthn

1 Registo

O dispositivo gera um par de chaves para este site (crypto assimétrica)

Resistente a phishing
A chave só funciona no domínio legítimo.

2 Registo

O servidor guarda apenas a chave pública.

Sem segredo partilhado
Uma fuga no servidor não expõe credenciais utilizáveis.

3 Login

O servidor envia um desafio aleatório.

Boa experiência
Biometria ou PIN local, e sincronização entre dispositivos.

4 Login

O dispositivo assina o desafio, após biometria ou PIN local, e o servidor verifica.

185

Protocolos de gestão de identidade e autenticação SSO (Single Sign On)

Protocolo	Para quê	Como funciona	Típico
Kerberos	SSO em redes de domínio	Autenticação com base em Tickets emitidos num modelo distribuído: AS + KDC	Pode usar Active Directory
LDAP	Consultar e autenticar em diretórios	Protocolo de acesso a diretório	OpenLDAP, AD
SAML 2.0	SSO para web empresarial	Assertions XML assinadas	Aplicações SaaS
OAuth 2.0	Autorização delegada	Access tokens com âmbitos (scopes)	Acesso a APIs
OpenID Connect	Login federado	ID token (JWT) sobre OAuth 2.0	Entrar com conta Microsoft ou Google
SCIM	Provisionamento de contas	API REST normalizada	Sincronizar utilizadores

186

Fluxo OAuth 2.0 / OpenID Connect

1

A aplicação redireciona o utilizador para o servidor de autorização, com PKCE.

Usar

Authorization Code com PKCE: web, móvel e SPA.

2

O utilizador autentica-se e dá o seu consentimento.

3

O servidor de autorização devolve um código de autorização.

Evitar

Fluxo implícito e fluxo de palavra-passe do proprietário.

4

A aplicação troca o código, com o verificador PKCE, por tokens.

Validar

iss, aud, exp e assinatura dos tokens; âmbitos mínimos.

5

A aplicação chama a API com o access token.

Tokens e sessões autenticadas

ESTRUTURA DE UM JWT

```
CABEÇALHO
{"alg":"ES256","typ":"JWT"}

PAYLOAD
{"sub":"u123","aud":"api-loja",
"exp":1760000000}

ASSINATURA (do emissor)
```

Assinado, não cifrado

Qualquer pessoa consegue ler o payload.
Não incluir dados sensíveis no token.

Validar sempre

Assinatura com o algoritmo esperado (rejeitar «none»), emissor, audiência e expiração.

Armazenar com cuidado

Cookies HttpOnly, Secure e SameSite; evitar localStorage para tokens sensíveis.

Revogar e rodar

Tokens de curta duração, refresh tokens rotativos e revogação central.

Acesso privilegiado e Zero Trust



Verificar explicitamente

Autenticar e autorizar cada pedido com identidade, dispositivo e contexto.



Menor privilégio

Acesso mínimo, limitado no tempo e no âmbito (just-in-time).



Assumir a violação

Segmentar, monitorizar e limitar o impacto de uma conta comprometida.

Gestão de acessos privilegiados (PAM)

Isolamento em “Cofres digitais” de credenciais

Acesso just-in-time

Sessões gravadas

Rotação automática

Tecnologias de gestão de identidades

Categoria	Função	Exemplos
IdP e SSO	Autenticação centralizada e federação	Microsoft Entra ID, Okta, Keycloak
Diretórios	Contas, grupos e políticas	Active Directory, OpenLDAP
MFA e FIDO2	Fatores fortes e passkeys	Chaves YubiKey, Windows Hello, apps autenticadoras
PAM e segredos	Acessos privilegiados e credenciais	CyberArk, BeyondTrust, HashiCorp Vault
Governança (IGA)	Ciclo de vida e recertificação	SailPoint, Microsoft Entra ID Governance
Autorização como código	Políticas de acesso centralizadas	OPA, AWS Cedar, OpenFGA
Identidade digital PT e UE	Identificação e assinatura qualificada	Cartão de Cidadão, Chave Móvel Digital, carteira europeia

Nota: xemplos ilustrativos: a escolha depende dos requisitos, da arquitetura e do custo.

Pontos chave a reter neste bloco

- 1 Segurança do software começa no desenho (*Security by Design*): modelar ameaças, codificar com segurança, testar e controlar as dependências.
- 2 As bases de dados concentram o valor: proteger por camadas, cifrar, auditar e garantir a integridade. Servidores de Bases de Dados como Servidores Críticos
- 3 Operar a BD com segurança: hardening, rede privada, backups testados e registos protegidos.
- 4 Decidir acessos com menor privilégio, combinando RBAC como base com ABAC para regras contextuais (modelos que são hoje suportados em SGBDs)
- 5 Identidade forte: MFA resistente a *phishing*, *passkeys*, SSO com OIDC e acessos privilegiados controlados.

Sugestões de Bibliografia

W. Stallings, L. Brown, Computer Security – Principles and Practice, Person
Chap 5 – Database and Data Center Security

Referências e recursos



OWASP

Top 10, ASVS, SAMM e
Cheat Sheets.

owasp.org



NIST SP 800-218 (SSDF)

Framework de
desenvolvimento seguro.

csrc.nist.gov



NIST SP 800-63

Orientações de identidade
digital e autenticação.

pages.nist.gov/800-63-4



NIST SP 800-207

Arquitetura Zero Trust.

csrc.nist.gov



CIS Benchmarks

Configuração segura de
SGBD e sistemas.

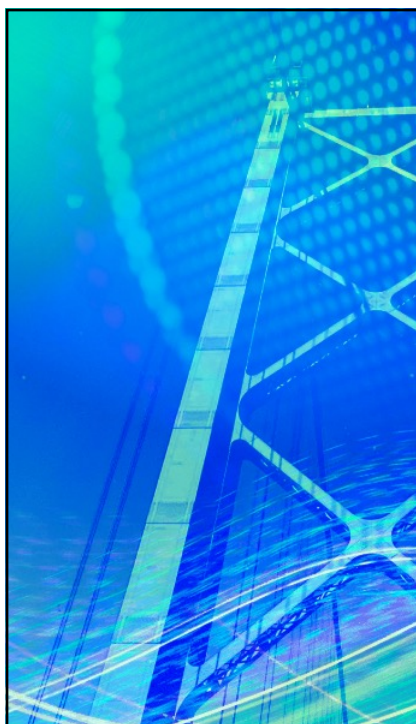
cisecurity.org



CNCS, CNPD e FIDO Alliance

Boas práticas, proteção de
dados e passkeys.

cncs.gov.pt · cnpd.pt ·
fidoalliance.org



Obrigado
Questões Finais

**Keep Learning.
Keep in touch.**

✉ info.executive@fct.unl.pt

