Frequência 2

Aula 6 - Smartcards

Smartcards

- Definição: Um cartão com capacidades de processamento computacional, incluindo CPU, memória ROM, EEPROM, e RAM.
- Interfaces: Podem ter contacto (com chip visível) ou ser sem contacto (usando RFID ou NFC).

Why smartcards

- Interoperabilidade: Os cartões inteligentes são projetados para funcionar em diversas indústrias e sistemas, facilitando a sua adoção em múltiplos contextos.
- Suporte a multi-aplicações: Podem ser utilizados para várias finalidades, como identificação, autenticação, transações financeiras e acesso a serviços.
- Transações seguras: Oferecem um elevado nível de segurança em transações devido à criptografia avançada.
- Aceitação pelo utilizador: A simplicidade e conveniência dos cartões inteligentes aumentam a aceitação e confiança do utilizador.
- Acessibilidade: Facilidade de uso e acessibilidade aumentada, permitindo a inclusão de diferentes grupos de utilizadores.

Smartcard applications: Communication protocol stack

- Protocolos de Comunicação:
  - APDU (Application Protocol Data Unit): Conjunto de comandos e respostas usados para comunicação entre o cartão e o leitor.
  - T=0 e T=1: Protocolos de transmissão de dados; T=0 transmite byte a byte, enquanto T=1 transmite blocos de bytes, sendo mais rápido.

T=0 and T=1

- T=0: Transmissão de cada byte separadamente, o que resulta numa comunicação mais lenta.
- T=1: Transmissão de blocos de bytes, permitindo uma comunicação mais rápida.
- ATR (Answer to Reset): A resposta do cartão a uma operação de reset, indicando o protocolo esperado pelo cartão.

APDU (ISO 7816-4)

- Comando APDU: Estrutura de comando e resposta definida pela norma ISO 7816-4.
  - CLA (Class): Indica a classe da instrução.
  - INS (Instruction): Especifica o comando a ser executado.
  - P1 e P2 (Parameters): Parâmetros específicos do comando.
  - Lc: Comprimento dos dados opcionais do comando.
  - Le: Comprimento dos dados esperados na resposta.
- Resposta APDU: Estrutura de resposta com dados e códigos de estado.

    - SW1 e SW2 (Status Words): Bytes de status que indicam o resultado da operação (0x9000 significa sucesso).

Smartcard: File system

- Sistema de Ficheiros: Estrutura de armazenamento de dados no cartão inteligente, organizada em diferentes tipos de ficheiros.
    - Master File (MF): Raiz do sistema de ficheiros, identificada por ID 0x3F00.
    - Dedicated File (DF): Similar a um diretório, pode conter outros DFs ou EFs.
    - Elementary File (EF): Ficheiro de dados comum, com tamanho fixo determinado na criação.

Java cards

- Cartões Java: Cartões inteligentes que executam applets Java, utilizando o Java Card Runtime Environment (JCRE).
    - JCRE: Inclui a Máquina Virtual Java, Card Executive (gestão e comunicações) e Java Card Framework (funções de biblioteca).

Cryptographic services

- Serviços Criptográficos:
    - Cifras: Algoritmos para cifragem e decifragem de dados.
    - Funções de Digestão: Algoritmos que produzem um resumo (hash) dos dados.
    - Geração e Gestão de Chaves: Criação e manutenção de chaves criptográficas.
    - Assinaturas Digitais: Criação e verificação de assinaturas eletrónicas.
    - Certificados de Chave Pública: Gestão de certificados que associam uma chave pública a uma entidade.

Cryptographic services: Middleware

- Middleware Criptográfico: Software que fornece uma interface entre aplicações e dispositivos criptográficos, permitindo a utilização de serviços criptográficos.
    - PKCS #11: Interface de Token Criptográfico (Cryptoki), definida pela RSA Security Inc.
    - PKCS #15: Formato de Informação de Token Criptográfico, também definido pela RSA Security Inc.
    - CAPI CSP: Provedor de Serviço Criptográfico do CryptoAPI, definido pela Microsoft para sistemas Windows.
    - PC/SC: Estrutura padrão para acesso a cartões inteligentes em sistemas Windows.

PKCS #11

- Integração Middleware Cryptoki: Interface de programação de aplicações (API) para acesso a dispositivos criptográficos, como cartões inteligentes.
    - Hierarquia de Objetos: Estrutura que define diferentes tipos de objetos como dados, chaves e certificados.
    - Sessões Cryptoki: Conexões lógicas entre aplicações e tokens, com sessões de leitura e escrita, geridas por diferentes proprietários (público, utilizador, oficial de segurança).
    - Operações de Sessão: Incluem login/logout, gestão de objetos e operações criptográficas.

Cartão de Cidadão: Middleware

- Middleware para Unix (Linux/MacOS): Componentes de software específicos para permitir o uso do Cartão de Cidadão nestes sistemas operativos.
- Middleware para Windows: Componentes de software para uso do Cartão de Cidadão em sistemas Windows, integrando-se com CryptoAPI e outros padrões.

Authentication with the PTEID

- Autenticação com PTEID: Utiliza um NONCE enviado ao Cartão de Cidadão para ser assinado com a chave privada. Enfrenta desafios como a falta de acesso direto do navegador ao cartão, exigindo o uso de plugins ou applets.

PT Authentication Plugin

- Plugin de Autenticação PT: Solução que permite acesso ao cartão através de um servidor web local instalado no computador do utilizador. Este plugin lida com solicitações autenticadas.

Mobile Digital Key (CMD)/Virtual Smart Card

- Chave Digital Móvel (CMD): Permite autenticação e assinatura sem necessidade do cartão físico, mas mantendo um nível de segurança semelhante. Utiliza autenticação de dois fatores (2FA), combinando um código PIN e um código enviado por outro canal, como SMS.

Aula 7 - FIDO

FIDO (Fast Identity Online) Alliance

- **Aliança FIDO**: Uma associação da indústria aberta com a missão de desenvolver padrões de autenticação e promover a sua adoção para reduzir o uso de senhas.
- **Abordagem**: Autenticação forte baseada em chaves públicas, resistência ao phishing e boa usabilidade.

FIDO token-based authentication

- **Autenticação baseada em tokens**: As chaves de autenticação são armazenadas em tokens. A autenticação é feita por meio de assinaturas, que são difíceis de copiar manualmente.
- **Enrolamento de dispositivos**: Adicionar dispositivos aos perfis dos utilizadores é responsabilidade dos autenticadores, incluindo o processo de recuperação em caso de perda de um token.

FIDO certification

- **Certificação FIDO**: Valida a qualidade dos produtos FIDO.
    - **Programas de certificação**: Funcionais (conformidade e interoperabilidade), autenticadores (proteção de segredos de nível L1 a L3+), e biométricos (taxas de falsos aceites e rejeições, IAPMR).

Universal 2nd Factor (U2F) protocol

- **Protocolo U2F**: O utilizador possui um dispositivo U2F que cria um par de chaves único por serviço (baseado em URL). Cada serviço regista a chave pública na conta do utilizador.

- **Interação**: Usado por APIs JavaScript em navegadores e APIs nativas do sistema operativo.

U2F devices

- **Dispositivos U2F**:
  - **USB, NFC e Bluetooth LE**: Dispositivos com interface HID reconhecível.
  - **Aplicações de software**: Podem ser suportadas por dispositivos de segurança de hardware.
  - **Presença do utilizador**: Necessário para prevenir o uso sem consentimento. O consentimento é normalmente dado através de toque num botão, impressão digital ou código PIN.

U2F protocols

- **Camada superior**: Protocolo criptográfico central que define a semântica e os dados trocados.
- **Camada inferior**: Protocolo de transporte host-dispositivo, conhecido como CTAP (Client To Authenticator Protocol).

U2F upper layer protocol: User registration

- **Registo do utilizador**: O dispositivo U2F gera um par de chaves específico para o serviço, identificando o serviço com um hash da sua identidade (protocolo, hostname, porta). O dispositivo retorna um Handle de Chave e a chave pública ao serviço.

U2F upper layer protocol: User authentication (1)

- **Autenticação do utilizador**: O utilizador fornece o seu identificador no serviço. O serviço retorna o Handle de Chave do utilizador e um desafio aleatório. A aplicação cliente do utilizador utiliza um dispositivo local para assinar os dados fornecidos (Handle de Chave, hash da identidade do serviço e hash dos dados do cliente).

U2F upper layer protocol: User authentication (2)

- **Processo de autenticação**: O dispositivo verifica se o hash da identidade do serviço é válido para o Handle de Chave. Em caso de sucesso, procura a chave privada correspondente e assina os dados do cliente, retornando a assinatura para validação pelo serviço.

Certification of U2F devices

- **Certificação de dispositivos U2F**: Os provedores de serviços precisam garantir a qualidade dos dispositivos U2F, que devem ter um par de chaves de certificação com certificado de chave pública emitido pelo fabricante. Os fabricantes precisam ser certificados pela FIDO.

Anonymity of attestation key pairs

- **Anonimato dos pares de chaves de certificação**: Os dispositivos U2F não podem ter pares de chaves de certificação únicos para evitar *tracking* de utilizadores. Em vez disso, os pares de chaves de certificação são partilhados por lotes.

dispositivos

Uncertified U2F devices

- **Dispositivos U2F não certificados**: Podem existir e ser usados, dependendo do serviço, mas os serviços precisam ter a sua própria cadeia de confiança para esses dispositivos.

FIDO2 and U2F

- **Compatibilidade**: O FIDO2 é compatível retroativamente com dispositivos U2F.

U2F JS / MessagePort API

- **API JavaScript U2F**: Interface JavaScript usada por páginas web para interagir com dispositivos U2F, utilizando uma API MessagePort.

WebAuthn

- **WebAuthn**: Parte do framework FIDO2, uma evolução da API U2F, especificada pela W3C e FIDO. Implementada por navegadores, permite lidar com o registo e autenticação de dispositivos U2F.

Client to Authenticator Protocol (CTAP)

- **CTAP**: Protocolo para interoperabilidade entre uma plataforma de utilizador (e.g., um laptop) e um autenticador criptográfico controlado pelo utilizador. Baseado no padrão de autenticação U2F.

CTAP variants

- **Variantes do CTAP**:
  - **CTAP1/U2F**: Também conhecido como FIDO U2F, utiliza formato de mensagem bruta.
  - **CTAP2**: Para autenticadores FIDO2 (também conhecidos como WebAuthn), utiliza o formato de serialização de dados CBOR (Concise Binary Object Representation).

Use case: Passkeys

- **Chaves de Acesso (Passkeys)**:
  - Surgiram para evitar problemas comuns de autenticação, como senhas fracas, phishing, roubo de senhas/cookies, falta de um segundo fator, ataques MITM ou leaks, e custos com o segundo fator.
  - Melhoram a usabilidade, evitando a necessidade de gerar, memorizar e gerir múltiplas senhas.
  - **Definição**: Passkeys são uma tecnologia de autenticação que utiliza material de autenticação do utilizador diretamente no dispositivo, sem ser exposto a terceiros. Gera um par de chaves (pública e privada) para autenticação, onde a chave pública é armazenada no serviço.

Use case: Passkeys Functionality

- **Funcionamento das Passkeys**:
  - Utilizam material de autenticação do utilizador diretamente no dispositivo, que nunca é exposto a terceiros.
  - Geram um par de chaves, sendo a chave pública armazenada no serviço.
  - A autenticação considera o serviço, dispositivo, chaves e utilizador, implicitamente usando 2FA.
  - **Certificação**: Capacidade de assegurar a proveniência do autenticador, garantindo que ele está realmente a fornecer os dados de autenticação. A chave pública é incluída num objeto de certificação, assinado por uma chave privada.

Use case: Passkeys Limitations

- **Limitações das Passkeys**:
  - **Suporte de dispositivos**: Tecnologia ainda nova.
  - **Dependência do dispositivo**: As chaves de acesso são específicas do dispositivo e podem não ser funcionais entre diferentes ecossistemas.
  - **Segurança biométrica**: Biometria pode não ser segura contra ataques locais, mas é melhor que apenas senhas.

Aula 8 - Kerberos e SAML

Authentication with Trusted Third Parties / KDCs

- Introdução à autenticação com Terceiros de Confiança (TTP) e Centros de Distribuição de Chaves (KDC).

Shared-key authentication

- **Autenticação com chave partilhada**:
  - **Orientada à conexão** e **sem conexão**.
  - **Problema**: Como distribuir a chave Kab para todos os pares A-B possíveis?

Authentication with Trusted Third Party: Key Distribution Center (KDC) concept

- **Conceito de KDC**:
  - TTP é responsável por intermediar a comunicação entre pares.
  - A e B não possuem informações partilhadas, mas ambos possuem informações partilhadas com a TTP.

Why KDC?

- **Porquê KDC?**:
  - Um TTP pode distribuir uma chave de sessão (KAB) para A e B, provando a identidade de cada um.
  - A chave de sessão KAB é temporária (apenas para uma sessão).
  - A e B usam KAB para provar a identidade um do outro de diferentes formas, no início ou durante cada interação da sessão.

Session key distribution

- **Distribuição de chave de sessão**:
  - A TTP (Terceira Parte Confiável) gera e distribui a chave de sessão KAB.
  - **Processo**:
    1. **A TTP** distribui a chave KAB para A e B.
    2. **B** recebe a chave KAB juntamente com um timestamp TB da TTP.
    3. **B** solicita uma prova de identidade usando a chave KAB.
    4. **A** e **B** utilizam a chave KAB para autenticar a comunicação.
- **Exemplo Visual**:
  - **Passo 1**: A TTP envia a chave KAB para A e B.
  - **Passo 2**: A e B usam KAB para autenticar e proteger suas comunicações.

- **Detalhes Adicionais**: A imagem no slide ilustra os fluxos de mensagens e como a chave é utilizada para estabelecer confiança entre A e B.

Example: SAML Web Browser SSO Profile

- **Perfil de SSO (Single Sign-On) do SAML para navegadores**:
  - Processo de autenticação de utilizador com afirmação de identidade SAML, usando cookies de autenticação para manter a sessão.

Kerberos: Goals

- **Objetivos do Kerberos**:
  - Autenticar pares num ambiente distribuído, inicialmente desenvolvido para o projeto Athena no MIT.
  - Distribuir chaves de sessão para adicionar segurança às sessões entre pares.
  - Objetivos incluem autenticação, confidencialidade (opcional) e SSO (Single Sign-On), permitindo ao utilizador lembrar-se de uma única senha.

Kerberos background: Needham-Schroeder (1978)

- **Antecedentes do Kerberos**:
  - Baseado no protocolo de Needham-Schroeder.
  - A e B confiam num KDC comum.
  - KDC partilha uma chave com cada A e B, atuando como uma autoridade central de autenticação.

Kerberos: Architecture and base concepts

- **Arquitetura e conceitos básicos do Kerberos**:
  - Serviços KDC do Kerberos: Serviço de Autenticação (AS) e Servidor de Concessão de Tickets (TGS).
  - Entidades (principais): Todas têm uma chave secreta partilhada com Kerberos (AS ou TGS).
  - Requisitos: Relógios bem sincronizados.
  - Elementos de autenticação: Ticket (necessário para solicitar um serviço) e autenticador (prova da identidade do solicitante).

Kerberos: Tickets and authenticators

- **Tickets e autenticadores no Kerberos**:
  - **Ticket**: Dado que não pode ser forjado, interpretado apenas pelo serviço alvo, contendo a identidade do cliente e uma chave de sessão.
  - **Autenticador**: Contém um carimbo de tempo do pedido, a identidade do cliente e prova que o cliente conhece a chave de sessão.

Overview of Kerberos SSO: 1st step: Login

- **Primeiro passo do SSO do Kerberos**:
  - Localização dos servidores Kerberos do domínio.
  - Autenticação do utilizador pelo Kerberos (AS).

- O utilizador recebe um Ticket Granting Ticket (TGT) e uma chave de sessão (KTGT) para interagir com outro serviço Kerberos (TGS).

Overview of Kerberos SSO: 2nd step: Authenticated access to servers

- **Segundo passo do SSO do Kerberos**:
  - O utilizador solicita ao Kerberos (TGS) um ticket para acessar um serviço (S).
  - O utilizador usa o TGT no pedido e deve provar que é o proprietário do TGT.
  - Recebe uma chave de sessão (KUS) e um ticket para o serviço (TUS), que é usado para fazer pedidos autenticados ao serviço.

Kerberos: Protocol (of version V5)

- **Protocolo do Kerberos (versão V5)**:
  - Descreve o fluxo de mensagens e interações entre cliente (C), servidor (S), AS e TGS.

Kerberos: Pre-authentication alternative

- **Alternativa de pré-autenticação no Kerberos**:
  - Previne ataques de dicionário proativos (Kerberoasting), filtrando pedidos ilegítimos.

Kerberos: Scalability

- **Escalabilidade do Kerberos**:
  - Domínios de autenticação (realms).
  - Cada domínio tem um servidor Kerberos, com cooperação entre domínios para permitir acesso entre diferentes realms.
  - Chaves secretas partilhadas entre servidores TGS de diferentes domínios, associadas a um caminho de confiança.

Kerberos V5: Security politics and mechanisms

- **Políticas e mecanismos de segurança do Kerberos V5**:
  - Autenticação de entidades usando chaves secretas, nomes e endereços de rede.
  - Períodos de validade com carimbos de tempo em tickets e autenticadores.
  - Proteções contra repetição usando nonces e carimbos de tempo/números de sequência.
  - Delegação e opções de autorização em tickets.
  - Autenticação entre domínios usando chaves secretas partilhadas e emissão de tickets de um TGS para outro.

Kerberos: Security issues

- **Questões de segurança do Kerberos**:
  - O KDC pode se fazer passar por qualquer pessoa, necessitando de máxima segurança na sua administração.
  - Pode ser um ponto único de falha, mas a replicação é uma opção.
  - Senha roubada de um utilizador permite a outros fazerem-se passar pela vítima em todos os serviços do domínio.

    - Credenciais TGS roubadas são menos arriscadas, pois sua validade é limitada (geralmente um dia).

Kerberos V5: Actual availability

- **Disponibilidade atual do Kerberos V5**:
    - Lançamentos do MIT disponíveis em web.mit.edu/kerberos.
    - Versões para Windows, com componentes modificados para acomodar credenciais do Windows.
    - Componentes incluem servidores/daemons Kerberos, bibliotecas para "kerberizar" aplicações, e aplicações de suporte como klogin, kpasswd, kadmin.
    - Aplicações kerberizadas (clientes e servidores).

Aula 9 - Gestão de identidade

Identidade Digital

- **Definição**: Um conjunto arbitrário de atributos de uma entidade, segregados em diferentes contextos.
- **Identificadores Contextuais**: Subconjuntos desses atributos usados para reconhecer a entidade em cada contexto.

Identity Manager (IdM)

- **Função**: Gerir perfis de identidade em cada contexto.
- **Tarefas**:
    - Criar e eliminar perfis de identidade.
    - Recolher e atualizar atributos nos perfis.
- **Objetivos**: Identificação, autenticação, autorização, controlo de acesso, contabilização, vigilância e rastreamento.

Identity Provider (IdP)

- **Função**: Fornecer atributos de identidade de um sujeito como afirmações (assertions).
- **Exemplo**: GitHub, Microsoft Azure AD.
- **Princípio**: "Need-to-know", ou seja, fornecer apenas os atributos necessários para cada solicitante, respeitando questões de privacidade.

Fonte Autorizada

- **Definição**: A principal IdM responsável por fornecer um dado atributo de identidade de um sujeito.
- **Função**: Garantir que há apenas uma fonte autorizada, mesmo que possa ser replicada para redundância.

Declaração de Identidade

- **Definição**: Uma afirmação que alguém faz sobre a identidade de si próprio ou de outro sujeito.
- **Função das IdMs/IdPs**: Prover conjuntos de declarações de identidade embaladas em afirmações confiáveis.

Abordagem 1: IdM Isolado ou Orientado a Silo

- **Características**:
  - IdM por serviço, sem relação entre serviços na organização.
  - Atributos de identidade não são partilhados entre serviços, resultando em duplicação.
  - Cada pessoa possui um perfil de identidade em cada serviço.
  - Desafios na integração e remoção de identidades entre serviços.
- **Exemplo Prático**: Uma empresa onde cada aplicação interna (e.g., email, ERP, CRM) requer login separado e não partilha informações de identidade.

Abordagem 2: IdM Agregado

- **Características**:
  - Um IdM para vários serviços, com um único perfil para cada entidade.
  - Mais eficiente na gestão, integração e remoção de identidades.
  - Utiliza um IdP central para autenticação e fornecimento de afirmações de identidade.
  - Serviços confiam no IdP.
- **Exemplo Prático**: Microsoft Office 365, onde um único login dá acesso a vários serviços (Outlook, Teams, SharePoint) usando Azure AD.

Abordagem 3: Identidade Federada

- **Conceito**: Políticas, práticas e protocolos comuns para gerir identidade entre organizações.
- **Objetivo**: Permitir que uma entidade aceda a um serviço de outra organização com um conjunto de afirmações de identidade fornecidas por IdMs confiáveis.
- **Exemplo Prático**: eduGAIN, que permite que utilizadores de uma universidade acedam a recursos de outra instituição participante.

Abordagem 4: Gestão de Identidade Baseada em Declarações

- **Características**:
  - Provisionamento de declarações de identidade por múltiplos IdPs.
  - O fornecedor de serviços solicita vários atributos de identidade como declarações e propõe IdMs alternativos.
  - O cliente do serviço utiliza um ou mais IdMs para obter todas as declarações necessárias.
- **Exemplo Prático**: OpenID Connect, onde utilizadores podem escolher autenticar-se através de Google, Facebook, ou outro provedor OpenID.

Credenciais

- **Definição**: Conjunto de declarações de identidade de um sujeito, afirmadas por um IdM.
- **Metadados**: Data de emissão, períodos de validade, atributos de identidade do emissor, e propósito da emissão.
- **Exemplo**: Cartões de identidade, certificados digitais.

Problemas de Privacidade

- **Questões**: Rastreamento, controle da apresentação das credenciais, e necessidade de auditoria.
- **Requisitos**: O proprietário da credencial deve provar a sua posse e controlar a sua apresentação.

Credenciais Verificáveis (VC)

- **Definição**: Credenciais seladas criptograficamente fornecidas a um detentor, que pode ser verificadas quanto à sua autenticidade, validade e origem.
- **Exemplo**: Certificados de vacinação armazenados em uma blockchain.

Provas de Conhecimento Zero (ZKP)

- **Definição**: Método que permite a uma entidade A provar a identidade de uma entidade B sem revelar informações adicionais.
- **Exemplo**: Provar que se sabe a localização de "Onde está o Wally?" sem revelar a localização exata.

Identidade Auto-Soberana (SSI)

- **Definição**: Identidade descentralizada que requer uma carteira digital para guardar credenciais digitais verificáveis.
- **Tipos de Credenciais**:
  - **Credenciais Atestadas por Terceiros**: Verificadas criptograficamente e confiáveis pelo receptor.
  - **Credenciais Auto-Atestadas**: Declarações que um indivíduo faz sobre si próprio.
- **Exemplo Prático**: Soluções de identidade digital como a European Digital Identity Wallet.

Interoperabilidade

- **Definição**: Capacidade de diferentes sistemas de cooperar, comunicando e entendendo-se mutuamente.
- **Tipos**:
  - **Interoperabilidade Sintática**: Capacidade de comunicar e analisar itens de comunicação.
  - **Interoperabilidade Semântica**: Capacidade de entender a informação trocada corretamente.

eIDAS

- **Definição**: Regulamento da UE sobre sistemas de identificação eletrónica e serviços de confiança para transações eletrónicas no mercado interno.
- **Objetivo**: Estabelecer normas para assinaturas eletrónicas, certificados, e serviços de confiança.
- **Exemplo**: Utilização de identidades eletrónicas nacionais para aceder a serviços públicos em diferentes países da UE.

Aula 10 - Anonimato e Privacidade

Privacidade

- **Definição**: O direito de alguém manter os seus assuntos pessoais e relações em segredo. Inclui o estado de estar sozinho ou de manter informações pessoais conhecidas apenas por um grupo restrito.

Tipos de Privacidade

- **Física**: Relacionada às propriedades do próprio corpo.
- **Vigilância**: Relacionada ao estado de ser observado.
- **Informação**: Relacionada à forma como as informações sobre si mesmo são manuseadas.

Modelo de Privacidade Digital IEEE

- **Expectativas de Privacidade (EoP)**: Quando e como os utilizadores esperam privacidade, englobando seis características: Identidades, Comportamentos, Inferências, Transações, Confidencialidade & Integridade, Acesso & Observabilidade.
- **Influências na Privacidade (IoP)**: Influências que moldam a infraestrutura de privacidade digital, incluindo influências técnicas, regulatórias, económicas, legislativas, legais, individuais, sociais e culturais.

Privacidade e Tecnologia

- **Problemas**: O uso da tecnologia cria informações (dados e metadados), processadas rapidamente e em grande volume, que podem ser vazadas.
- **Dilema**: É possível usar tecnologia sem fornecer dados pessoais?

Privacidade e Empresas

- **Necessidade de Dados**: As empresas precisam de dados para modelos de negócio (financeiros, marketing, transacionais).
- **Compromisso da Privacidade**: A privacidade é frequentemente comprometida para fornecer produtos, causando potencial impacto legal e na perceção da marca pelos utilizadores.

Privacidade e IAA (Identificação, Autenticação e Autorização)

- **Conexão**: A privacidade está fortemente ligada aos métodos e tecnologias de IAA, que lidam com utilizadores, seus atributos e relações.

Identificação

- **Relevância da Privacidade**: Os dados de identificação podem revelar informações adicionais, facilitando rastreamento, roubo de identidade e outras ameaças.
- **Melhores Práticas**: Identificadores devem ser específicos do serviço, únicos dentro dos limites de usabilidade, e não devem revelar informações adicionais.

Autenticação

- **Relevância da Privacidade**: Os dados de autenticação podem revelar informações pessoais adicionais e comportamentais, que podem ser usados para rastreamento e outras ameaças.
- **Melhores Práticas**: Dados de autenticação devem ser manuseados de acordo com o GDPR, com práticas claras de processamento de dados e consentimento do utilizador.

Anonimato

- **Definição**: O estado de não ser observável dentro de um conjunto de sujeitos, definido pelo conjunto de anonimato.
- **Conceito Dependente do Contexto**: O contexto define o conjunto de anonimato em relação a uma ação particular.

Problemas de Privacidade de Microdados

- **Microdados**: Informações ao nível de respondentes individuais.
- **Problemas**: Como compartilhar microdados sem expor a identidade das pessoas que os forneceram?

Melhorias na Privacidade de Microdados

- **Remoção de IDs Potencialmente Únicos**: Remover identificadores únicos (nomes, IDs nacionais, números de telefone, etc.) para evitar a ligação de microdados entre várias bases de dados.
- **Adição de Ruído**: Adicionar ruído aos dados armazenados ou aos resultados das consultas para aumentar a privacidade, embora comprometa a integridade.
- **Truncamento**: Não fornecer dados completos, limitando a precisão para aumentar a privacidade.

K-Anonimato

- **Definição**: Nenhuma consulta pode devolver um conjunto de anonimato com menos de k entradas.
- **Atributos Críticos**: Identificadores únicos, quase-identificadores (quando combinados produzem tuplas únicas) e atributos sensíveis.
- **Exemplo Prático**:
  - **Dados Originais**:

Nome	Idade	Sexo	Código Postal	Doença
Sam	29	M	43102	Diabetes
Gloria	38	F	43102	Cancro da Mama
Adam	51	M	43102	Cancro do Cólon
Eric	29	M	43102	Diabetes
Tanisha	34	F	43102	HIV
Don	51	M	43102	Doença Cardíaca

  - **Após Remoção de Identificadores Únicos**:

Idade	Sexo	Código Postal	Doença
29	M	43102	Diabetes
38	F	43102	Cancro da Mama
51	M	43102	Cancro do Cólon
29	M	43102	Diabetes
34	F	43102	HIV
51	M	43102	Doença Cardíaca

  - **Generalização para 2-anonimato**:

Idade	Sexo	Código Postal	Doença
30	M	43102	Diabetes
40	F	43102	Cancro da Mama
50	M	43102	Cancro do Cólon
30	M	43102	Diabetes
30	F	43102	HIV

Idade	Sexo	Código Postal	Doença
50	M	43102	Doença Cardíaca

L-Diversidade

- **Problema com K-Anonimato**: Vulnerável a ataques de homogeneidade e conhecimento prévio.
- **Solução**: L-diversidade, onde os resultados de uma consulta devem conter pelo menos l valores diferentes para cada atributo sensível.
- **Exemplo**:
  - **2-anonimidade com 1-diversidade**:

Idade	Sexo	Código Postal	Doença
30	M	43102	Diabetes
40	F	43102	Cancro da Mama
50	M	43102	Cancro do Cólon
30	M	43102	Diabetes
30	F	43102	HIV
50	M	43102	Doença Cardíaca

  - **2-anonimidade com 2-diversidade**:

Idade	Sexo	Código Postal	Doença
30	M	43102	Diabetes
40	F	43102	Cancro da Mama
50	M	43102	Cancro do Cólon
30	M	43102	Diabetes
30	F	43102	HIV
50	M	43102	Doença Cardíaca

Conclusão

- **Limitações**: Tanto k-anonimidade quanto l-diversidade têm falhas, sendo vulneráveis a vários tipos de ataques, como homogeneidade, conhecimento prévio, enviesamento e similaridade.
- **Exemplo de Ataque de Homogeneidade**: Se um atacante souber que Bob tem 27 anos e mora no código postal 47678, pode concluir que Bob tem doença cardíaca.
- **Exemplo de Ataque de Similaridade**: Se um atacante souber que Bob tem um salário baixo (3k-5k), pode inferir que ele tem uma doença relacionada ao estômago.