Fundamentos de Segurança Informática (FSI)

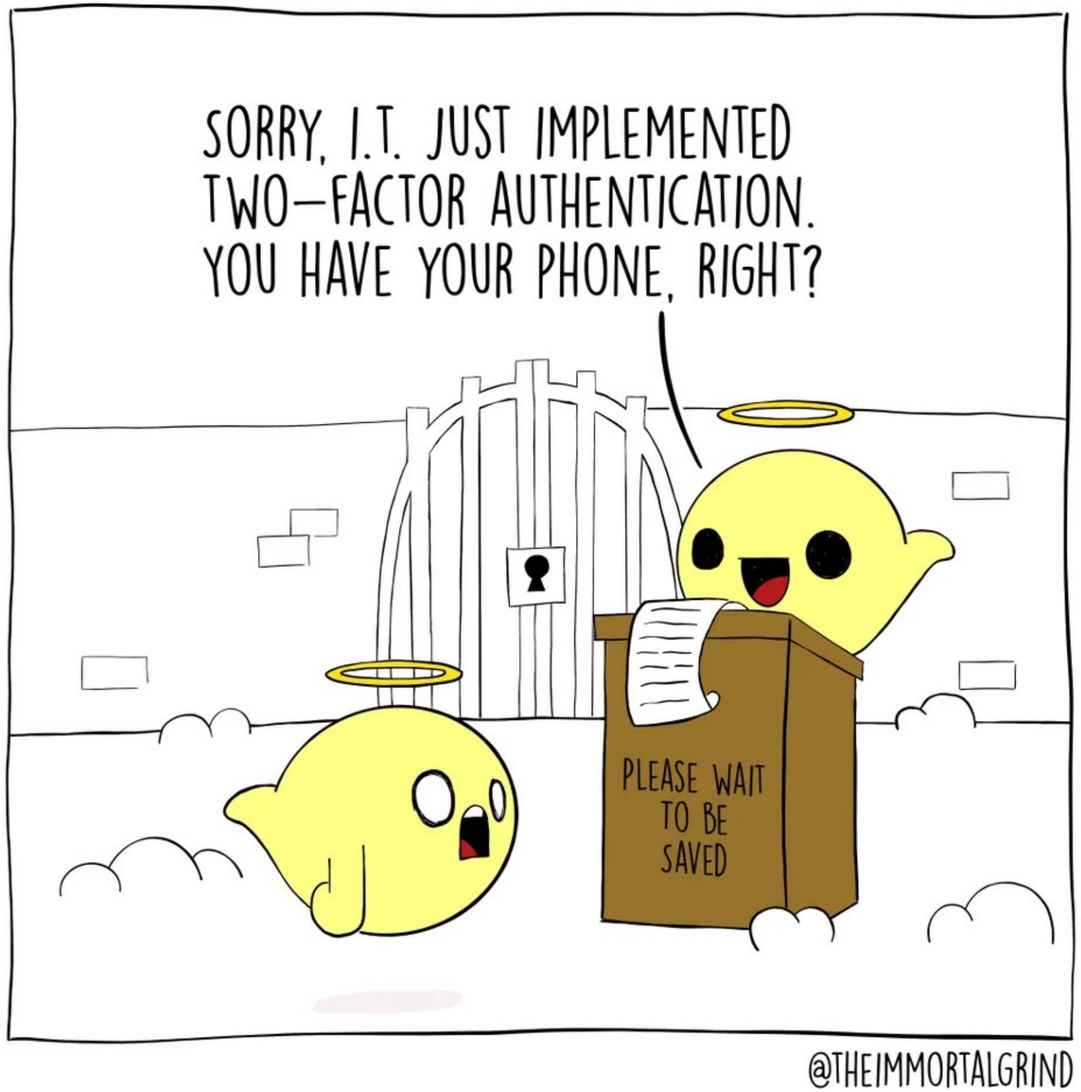
2021/2022 - LEIC

Manuel Barbosa
mbb@fc.up.pt

Aula 20
Autenticação 2

Multi-factor

- Autenticação multi-factor
  - Defesa em profundidade
  - Password vai ser quebrada
  - Preparar sistema para utilizações suspeitas/operações críticas
  - Utilizar fatores adicionais para confirmação e/ou deteção de problemas



https://www.reddit.com/r/comics/comments/elejb2/twofactor_authentication_oc/

Algo que se possui

- Utiliza-se um dispositivo físico que apenas a Alice possui:
  - chave, cartão de códigos, smartcard, RFID, token específico
- Utilizado frequentemente como segundo fator (além da password)
  - two-factor authentication token
  - estritamente: prova de posse de token <> prova de identidade
  - mas reforça a evidencia de identidade

Smartcards

- Processador embutido em cartão de plástico
  - Alice traz cartão consigo
  - cartão pode armazenar e processar chaves criptográficas
  - processador do cartão interage com o exterior através de NFC ou contactos
- Muitas utilizações para além da identificação:
  - cartões SIM, descodificadores de satélite, multibanco, etc.
- Exemplo de autenticação: challenge-response de uma assinatura digital (FIDO)
- O próprio smartcard frequentemente inclui outro fator de autenticação: PIN

Smartcards


JAPAN AIRLINES
JAL MILEAGE BANK
JAL CARD
CLUB-A
JAL GLOBAL CLUB
0000 0000 0000 0000
3333
有効期限
MONTH(月)/YEAR(年)
MEMBER SINCE 00
GOOD THRU - 12/11
MEMBERSHIP NBR
お得意様番号 00 000 0663
AAAAAA AAAAAAAAA
VISA


894412
00501
0236
894412
10906
0617
16K
880037
894412
11542
7371
GEMPLUS
16k
894412
00704
0393
894412
20922
0367

One-Time Tokens

- Ideia semelhante ao smartcard, mas com design específico para autenticação (baixo custo, alta segurança, etc.)
- Geralmente sem interface eletrónico
  - não é possível inserir desafio, ou muito limitado
  - resposta visualizada num pequeno ecrã

https://ergonomics-europe.com/products-solutions/one-time-password-token-otp-token/

https://www.tokenguard.com/RSA-SecurID-SID700.asp

One-Time Tokens

- Protocolo típico não interativo:
  - criptografia simétrica: segredo pré-partilhado com servidor
  - periodicamente (e.g. 1m) token gera MAC de hora actual
  - servidor pede password e MAC recente (dois factores)
- Vantagens:
  - defesa em profundidade: conhecer password não é suficiente
  - códigos MAC são de uso único (não se copiam, roubam)
  - códigos MAC futuros são imprevisíveis

One-Time Tokens

- Protocolo típico não interativo:
  - criptografia simétrica: segredo pré-partilhado com servidor
  - periodicamente (e.g. 1m) token gera MAC de hora actual
  - servidor pede password e MAC recente (dois factores)
- Desvantagens:
  - mesmo canal: vulnerável a Man-in-the-Middle e phishing (como password)
  - servidor precisa de armazenar chaves secretas (escalabilidade, ponto único de falha)

One-Time Passcode

- Soluções mais recentes utilizam simplesmente um dispositivo existente:
  - utilizar App noutro dispositivo para gerar códigos one-time, ou
  - utilizar App noutro dispositivo para fazer challenge-response
  - aka *token virtual*, e.g., no telefone, tablet, relógio.
- Vantagens: menos um dispositivo, chaves mais fáceis de gerir
- Desvantagens: menor garantia de independencia entre fatores (e.g. telefone roubado)

One-Time Passcode

- Caso mais simples: código enviado por SMS e introduzido com password
- Muitas vezes utiliza-se para aumentar garantias de segurança:
  - e.g., transação bancária que implica transferência de valores
  - e.g., confirmação de identidade em caso de comportamento suspeito
  - e.g., confirmação de identidade em caso de alteração de password
- Nestes mecanismos são também utilizados outros canais: e.g. email
  - independência entre canais: garantia de se tratar da pessoa correta

One-Time Passcode

- Conclusão:
  - muito utilizado, adequado a casos de uso comuns
    - usabilidade vs segurança
  - insuficiente para cenários *security-critical*:
    - e.g., confirmar lançamento de míssil com SMS
    - e.g., confirmar transferência de 1MEUR com SMS
    - etc.

Biometria

Biometria

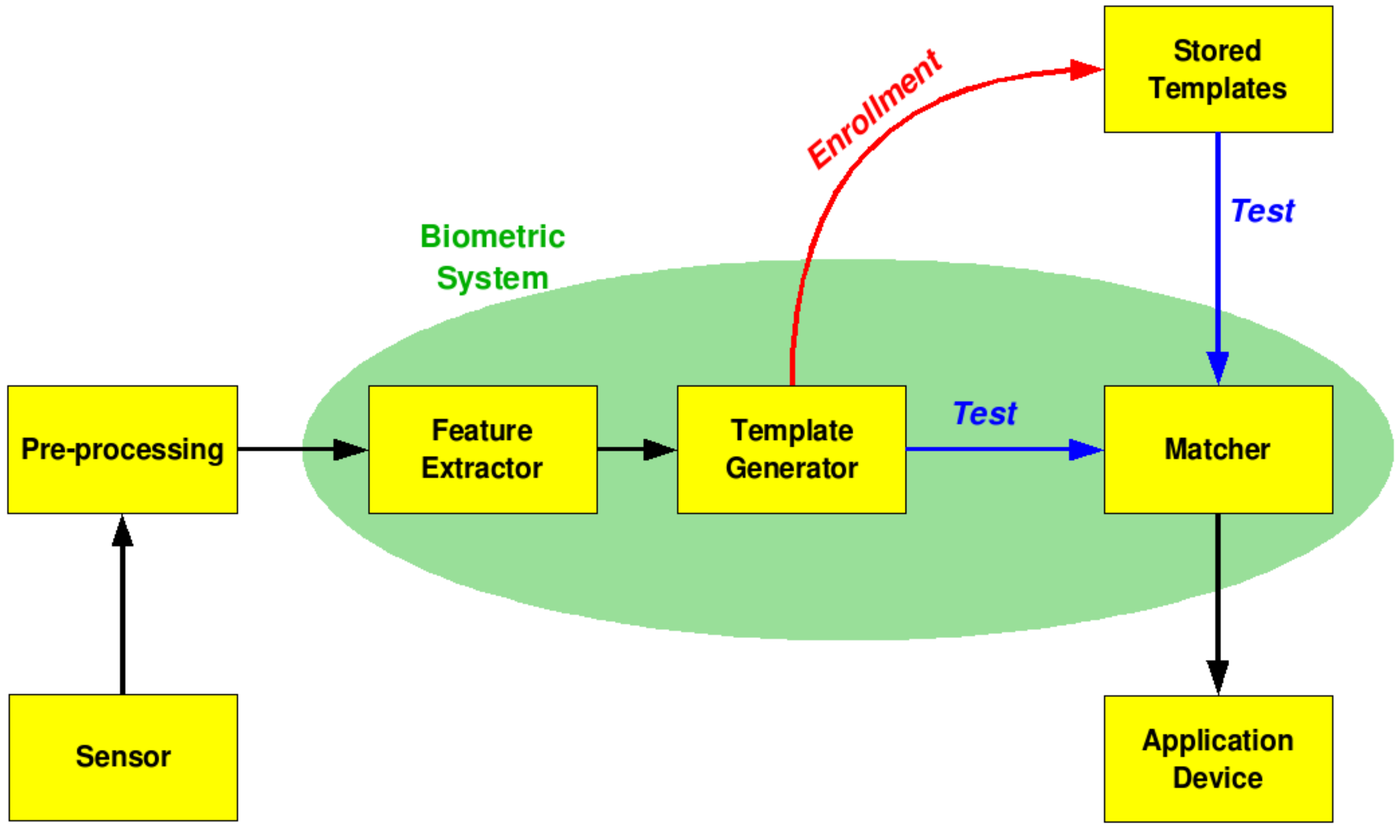
- Prova de identidade por característica intrínseca da pessoa
  - Característica física: impressão digital, forma da iris, face
  - Característica comportamental: caligrafia, uso do teclado
  - Combinação de ambas: voz, forma de andar (gait)
- Transposição para o meio digital da forma como nos identificamos em sociedade:
  - vantagens: não é transferível, usabilidade ideal, pode dar garantias fortes
  - desvantagens: problemas de privacidade, direito ao esquecimento

Biometria

- Impressão digital
- Impressão da palma da mão
- Reconhecimento da retina
- Reconhecimento da iris
- Reconhecimento facial
- Padrão vascular (e.g., mão)
- Reconhecimento de voz
- Reconhecimento de caligrafia
- Padrão de utilização de teclado
- Forma de caminhar
- Batimento cardíaco
- ADN

Biometria: processo

- Enrollment => Registo
  - recolha de amostras
  - extração de “templates”
- Autenticação:
  - recolha de amostra
  - match com templates



Biometria: processo

- Em autenticação remota temos várias hipóteses:
  - servidor recebe amostra(s) em bruto
  - servidor recebe “templates”/características já processadas para match
  - servidor recebe apenas resultado de match
- Ordem crescente de necessidade de confiança servidor => cliente
- Ordem decrescente de necessidade de confiança cliente => servidor

Biometria: processo

- Caso #1: servidor recebe amostras em bruto
  - servidor confia no computador da Alice para recolher amostras
  - todas a informação biométrica está armazenada num servidor central!

Biometria: processo

- Caso #2: servidor recebe template/características para match
  - servidor confia no computador da Alice para recolher amostras e calcular/extrair templates
  - servidor (ainda) tem acesso a um conjunto de templates/características
  - informação biométrica ainda armazenada no servidor

Biometria: processo

- Caso #3: servidor recebe apenas resultado
  - servidor confia no computador da Alice para processo de autenticação => implica alguma forma de atestação/hardware confiável
  - servidor não tem acesso a informação biométrica

Biometria: desafios

- Técnicos:
  - precisão
  - usabilidade, nomeadamente no registo
  - estabilidade dos templates e características
  - armazenamento de informação sensível/dados pessoais
  - frescura: evitar replay attacks, integridade, robustez, etc.
- Não técnicos:
  - aceitação pelos utilizadores

Biometria: registo

- O registo de um utilizador muito mais elaborado do que para uma password
- Obriga a fazer um conjunto de medidas até que uma precisão adequada possa ser garantida
- Usual existirem indivíduos para os quais é difícil obter a precisão desejada

Biometria: precisão

- Dois eixos (pode ser definido para um indivíduo ou para todo o sistema!):
  - taxa de falsos positivos (FAR): probabilidade de aceitar indevidamente
  - taxa de falsos negativos (FRR): probabilidade de rejeitar indevidamente
- FAR baixo => parece desejável
  - mais difícil de atacar, não aceitamos “o atacante”, mas
  - sempre associado a FRR alto, o que prejudica a usabilidade
  - muitas vezes calibra-se para o Equal Error Rate Point: FAR = FRR

Biometria: segurança

- Os ataques maliciosos a biometria geralmente são de dois tipos:
  - intercepção: perda de confidencialidade
    - permite falsificação (ver spoofing), problemático porque característica biométrica não pode ser alterada nos indivíduos
    - perda de privacidade: podemos ser reconhecidos
  - usurpação (spoofing):
    - criação de característica falsa que engana o sensor, e.g., modelo de dedo, imagem de iris, etc.
    - pode ser simplesmente um replay attack (fácil de detectar com log)
    - maior precisão na verificação pode tornar difícil este tipo de ataques
    - podem usar-se fatores biométricos adicionais para “liveness”, como temperatura, pulso, etc.
    - pode usar-se combinação com fator não biométrico: pin, token, etc.

Autenticação e Sessões Web

O que é uma sessão?

- Sequência de pedidos/resposta a um (ou mais) sites/aplicações:
  - pode ser longa (e.g., Gmail) ou curta
  - sem o conceito de sessão: todos os pedidos exigiriam nova autenticação
- Sessão:
  - autenticar utilizador uma vez
  - manter essa informação para os pedidos seguintes

HTTP auth

- Na pré-história => HTTP auth
  - Servidores HTTP mantinham ficheiros de (hash de) passwords em pastas
  - Resposta do servidor incluía pedido de autenticação por password
  - Browser mostra formulário
  - Browser envia hash de password em todos os pedidos subsequentes para a mesma pasta!
    - `Authorization: Basic ZGFddfibzsdfgkjheczI1NXRleHQ=`

HTTP auth

- Na pré-história => HTTP auth => Não utilizar!
  - Log-out implica fechar o browser
    - como gerir múltiplas contas do mesmo utilizador?
  - O site não controla o interface para inserção de password
    - essa interface é confusa/facilmente se engana utilizadores

Tokens de sessão

- O servidor cria um “testemunho” que fica guardado do lado do cliente e é devolvido em todos os pedidos relacionados:
  - cookie
  - informação embutida nos links clicáveis
  - campos escondidos em formulários
- Hoje em dia utiliza-se uma combinação destes mecanismos, para garantir robustez (não vamos ver detalhes)
- Tokens devem ser imprevisíveis, e invalidados (em ambos os lados) no logout

Session Hijacking

- Ataques a tokens de sessão:
  - roubo de token:
    - Cross-Site Scripting (XSS)
    - eavesdropping sobre HTTP ou MitM em HTTPS
    - falha de logout (token não invalidado no servidor)
    - mitigação: ligar token à máquina, e.g., endereço IP (pode levar a logout acidental)

Session Hijacking

- Ataques a tokens de sessão:
  - token fixation:
    - Atacante inicia sessão (low privilege) e recebe token
    - Atacante “convence” utilizador a fazer login com o mesmo token (e.g., embutido numa URL)
      - o token do atacante passa a ter privilégios do utilizador
    - mitigação: nunca elevar privilégios/fazer login sem criar token novo