CPqD 35 anos

Como não escolher a sua senha!
Será a senha gráfica o futuro das senhas?

Roteiro

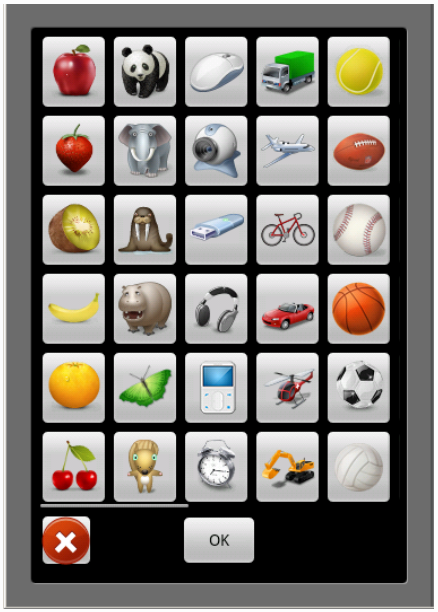
1. Revisando o ABC das senhas
2. Fatos desagradáveis sobre senhas
3. O que é a senha gráfica
4. Alguns resultados interessantes
5. Aspectos de implementação

O CPqD

P&D + Produto = Inovação

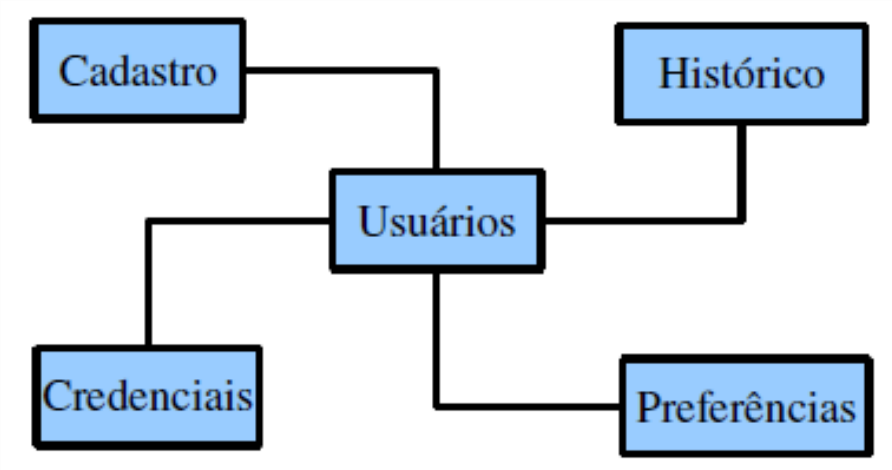
Motivação

- Senhas gráficas podem substituir as senhas alfanuméricas
  - Quando o meio de entrada de dados não é um teclado
  - Como é o caso dos dispositivos móveis com tela de alta resolução e de superfície sensível ao toque.



O ABC das senhas

- A modelagem dos dados do usuário não pode ser feita de forma monolítica.
- A obediência ao princípio da separação de responsabilidades leva à modelagem das informações de um usuário de forma a separar a senha de outros dados.
  - Cada grupo de informações pode ser usado por sistemas distintos, os quais não precisam ter acesso a informação desnecessária.
  - Um histórico evita o reuso de credenciais antigas.
  - Dados cadastrais são mantidos separados das preferências.



O ABC das senhas

- Políticas de senhas fortes
  - Não devem conter palavras de dicionário
  - Maior cadeia possível: letras, números, caracteres especiais
  - Podem ser criadas a partir de frases longas:
    - Ex: “Minha terra tem palmeiras onde canta o sabiá, os passaros daqui não cantam como os de lá” → Mttpocsopdnccodl → M77p0c$0pdncc0d1
    - Primeira letra de cada palavra, trocar caracteres por números ou símbolos parecidos

- Proteção do repositório das senhas
  - Proteção do arquivo de senhas
    - Admin read-only
  - Proteção das senhas armazenadas
    - Hash, salt, stretching

“ I needed a password with eight characters so I picked Snow White and the Seven Dwarves."

O ABC das senhas

- Salting
  - Inclusão de informação adicional no hash
    - Número aleatório
    - Sal para personalizar o sabor…
  - Adiciona um terceiro campo ao armazenamento da senha (salt)
  - Exemplo: alice com a senha automovel
  - Hash da senha concatenada com o salt:
    - h(automovel|1115) = ScF5GDhW...

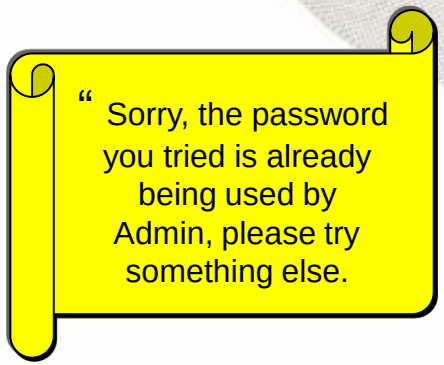
- H(senha) = h → função de hash
  - Uso comum sem salt
- Estratégia 1
  - H(senha + salt) = h'
- Estratégia 2
  - H(H(senha)+salt) = h''

1f u c4n r34d th1s u
r34lly n33d t0 g37 l41d

```
alice:dJoTsDhWeHr2q5m7mSDuGPVasV2NHz4kuu5n5eyuMbo=:1115
```

O ABC das senhas

- Senha “Honeypot”
  - Combinação simples usuário/senha como isca para atacantes
    - (guest/guest), (usuario/usuario), (abcd/1234)
  - Grande chance de atacantes testar combinações simples
  - Alarme quando a combinação isca for ativada
  - Pode ser uma indicação de ataque
    - Ação pós alarme depende do objetivo almejado
    - Ex.: Rastreamento do atacante ou estatística de varredura indevida.



- Filtragem de senhas ruins
  - Usuário escolhe sua senha
    - Dentro de certas restrições para garantir senhas fortes Ex: dicionário
  - Sintaxe da senha
    - Expressão regular – conjunto seguro de senhas
    - Mistura de letras maiúsculas, números e caracteres especiais
    - Tamanho mínimo

O ABC das senhas

- Envelhecimento das senhas
  - Encorajar ou exigir que os usuários troquem as senhas
    - com certa freqüência
    - Toda vez que usuário fornece a senha, há ameaça potencial de captura
    - A troca freqüente limita o tempo de disponibilidade da senha capturada
  - A senha pode ser aceite um número fixo de vezes
  - Observação: a senha trocada com muita freqüência leva à evasão

- Limite de tentativas de login
  - Permite de 3 a 4 tentativas, então desabilita ou bloqueia o usuário
  - Atacante só tem um número pequeno de “chutes”
  - Inconveniente para usuários “esquecidinhos”
  - Usuário legítimo deve acessar o administrador para liberara acesso ou recriar a senha
  - Há potencial para ataque de DoS sobre sistemas
    - Número grande de usuários bloqueados de forma aleatória

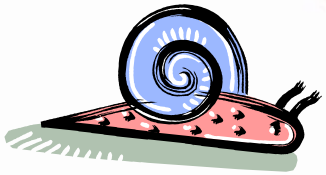
O ABC das senhas

- Atraso forçado ou artificial
  - Atraso artificial para cada tentativa de login no sistema
  - Uma estratégia de implementação possível
    - Espera 2^n segundos após a n-ésima falha do usuário
      - De um IP/MAC/número
  - Inconveniência é insignificante para usuário legítimo
    - Após uma falha o atraso é de 2 segundos
    - Após três falhas o atraso é de 8 segundos
  - O ataque de força bruta é mais custoso
    - O número de tentativas por intervalo de tempo diminui
  - CUIDADO Proxies HTTP!
    - O erro de digitação de um usuário vai atrasar todos os outros do mesmo IP

```
ContaTentativa := 0;
sucesso = falso;
faça {
      lê credencial;
      testa credencial;
      se (teste bem sucedido){
            sucesso = verdade;
            retorna sucesso;
      }
      contaTentativa := contaTentativa + 1;
      Insere atraso;
      atraso := atraso * 2;
} enquanto (contaTentativa < limite);
se (contaTentativa = limite) {
      bloqueia sistema;
}
retorna sucesso;
```

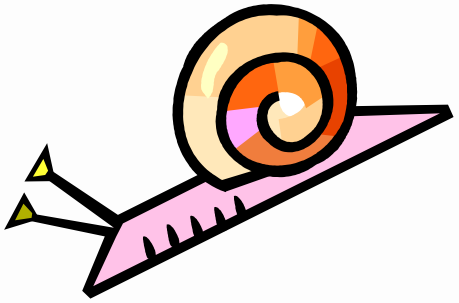
O ABC das senhas

- Computação mais demorada
  - Atraso pelo aumento do tempo de computação
    - durante a verificação da senha



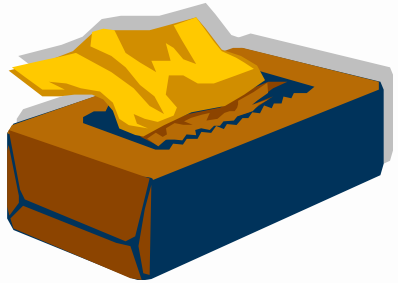
  - Uma estratégia de implementação possível
    - Cadeia de hashes
    - H(x) = h → computação rápida
    - H(H(H(... H(x)...))) = h → computação demorada

  - Assim como no caso anterior
    - Inconveniência é insignificante para usuário legítimo
    - O ataque de força bruta é mais custoso
  - Key stretching
    - http://en.wikipedia.org/wiki/Key_stretching



O ABC das senhas

- Login bem sucedido mais recente
  - Notificação do usuário quando da último login bem sucedido
    - Data, hora, localização
    - Usuário deve ser incentivado a prestar atenção
    - Usuário deve ser orientado a informar discrepâncias
  - Discrepâncias podem indicar ataques
    - Localização improvável e/ou Horário impróprio



- Senhas Descartáveis (One-Time-Password → OTP)
  - Múltiplos usos da senha dão ao atacante múltiplas oportunidades
  - Login com senha diferente toda vez
  - Dispositivos geram senhas para serem usadas a cada login
    - Dispositivo usa semente para gerar cadeia de senhas
    - Servidor sabe a semente, tempo atual e/ou contagem, e pode verificar senha
  - Dispositivos OTP integrados a PADs e Smartphones

O ABC das senhas

- Usabilidade para segurança
  - O software seguro é usável se seus usuários:
    - Estão conscientes das tarefas de segurança que precisam realizar
    - Sabem como realizar as tarefas de segurança com sucesso
    - Não cometem erros perigosos
    - Estão confortáveis com a interface.

Segurança X Usabilidade

- As mensagens de erro “senha inválida” e “usuário incorreto” informam qual das duas informações (identificação ou credencial) fornecidas está incorreta.
- Logo, o atacante obtém uma informação até então ignorada.
- A mensagem adequada seria “usuário ou senha inválidos” ou “Autenticação mal sucedida”.

Fatos desagradáveis sobre as senhas

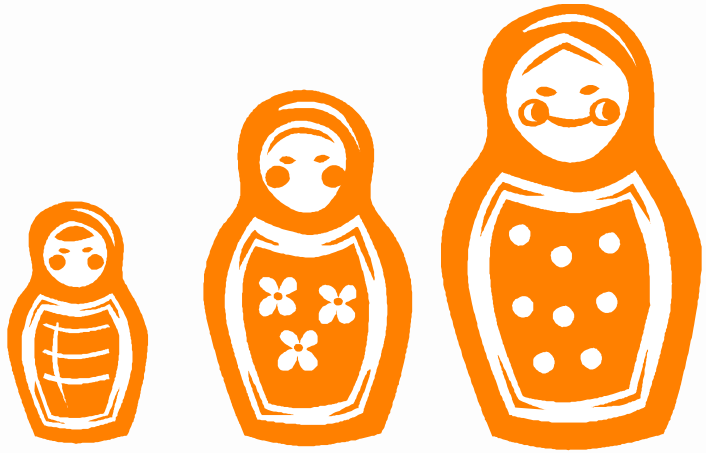
- Fato 1: Para os usuários, as senhas são reutilizáveis
  - Quando um usuário tem várias senhas (textuais), a dificuldade de memorização de algumas senhas fortes, ou até mesmo de várias senhas fracas, faz com que o usuário reutilize senhas entre sistemas.

Fatos desagradáveis sobre as senhas

- Fato 2: Usuários preferem as senhas pronunciáveis
  - Formato de palavra (Consoantes e vogais formando sílabas) é mais fácil de lembrar
  - Se as senhas são legíveis (se parecerem com palavras), elas são vulneráveis aos ataques de dicionário.
  - Senhas pronunciáveis seriam mais memorizáveis e por isto usuários tenderiam a escolher senhas que se parecem com palavras.
  - Este comportamento preserva a freqüência das letras no idioma nativo do usuário.

Fatos desagradáveis sobre as senhas

- Fato 3: Usuário escolhe a menor senha possível
  - Ao oferecer uma faixa (um intervalo) de tamanhos válidos para as senhas (ex. 6 a 9 caracteres), é bastante provável que uma boa parte dos usuários escolha senhas pequenas (próximas ao limite inferior).



Fatos desagradáveis sobre as senhas

- Fato 4: Interpretação literal da política é mais fácil
  - As orientações para construção de senhas alfanuméricas são geralmente excessivamente genéricas e não seriam capazes de produzir senhas resistentes aos ataques de força bruta.
  - A senha "12asLK!?" é fraca pois é derivada da leitura literal da seguinte orientação geral:

    “uma boa senha deve consistir de pelo menos dois números, duas letras minúsculas, duas letras maiúsculas e dois sinais de pontuação”.

Fatos desagradáveis sobre as senhas

- Fato 5: A senha nunca é tão boa quanto parece. Diante do aleatório, o usuário prefere inserir padrões e semântica.

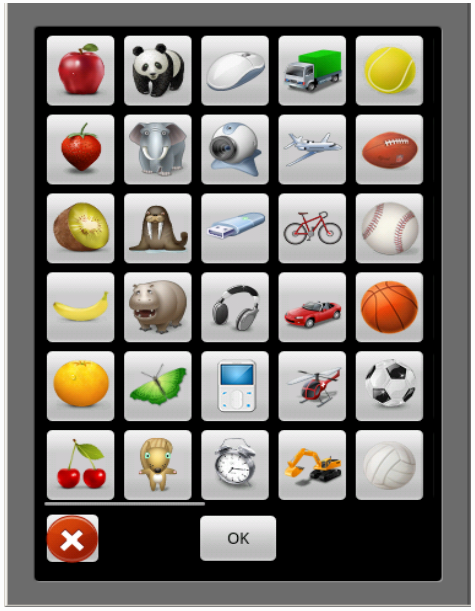
Política de senhas	#bits	up
PIN smartcard: 4 dígitos (0-9)	13,29	14
Data (DDMMAA) 365d X 100a	15,16	16
4 letras (AAAA) A-Z	18,8	19
6 dígitos (999999) 0-9	19,93	20
4 alfanum. A-Z e 0-9	20,68	21
8 dígitos (99999999) 0-9	26,58	27
6 letras (AAAAAA) A-Z	28,2	29
6 alfanuméricos A-Z e 0-9	31,02	32
6 letras c/ M != m	34,2	35
6 alfanuméricos c/ M != m	35,73	36
8 letras A-Z	37,6	38
8 alfanuméricos A-Z e 0-9	41,36	42
PCI:7 alfanuméricos (26 + 10 + 26)	41,68	42
8 letras c/ M != m	45,6	46
8 alfanumérico c/ M != m	47,63	48
Senha corporativa (8): 52 + 10 + 24	51,41	52

Senha PCI degenerada	#bits	up
Senha com caracteres aleatórios	41,7	42
contendo palavra de 4 letras	33,3	34
contendo palavra com 5 letras	27,4	28
contendo palavra com 6 letras	21,8	22
palavras com 4 letras na frequência	28,7	29
palavras com 5 letras na frequência	25,5	26
palavras com 6 letras na frequência	20,3	21
contendo data na forma DDMM	28,4	29
palavra com 4 letras + data DDMM	18,5	19

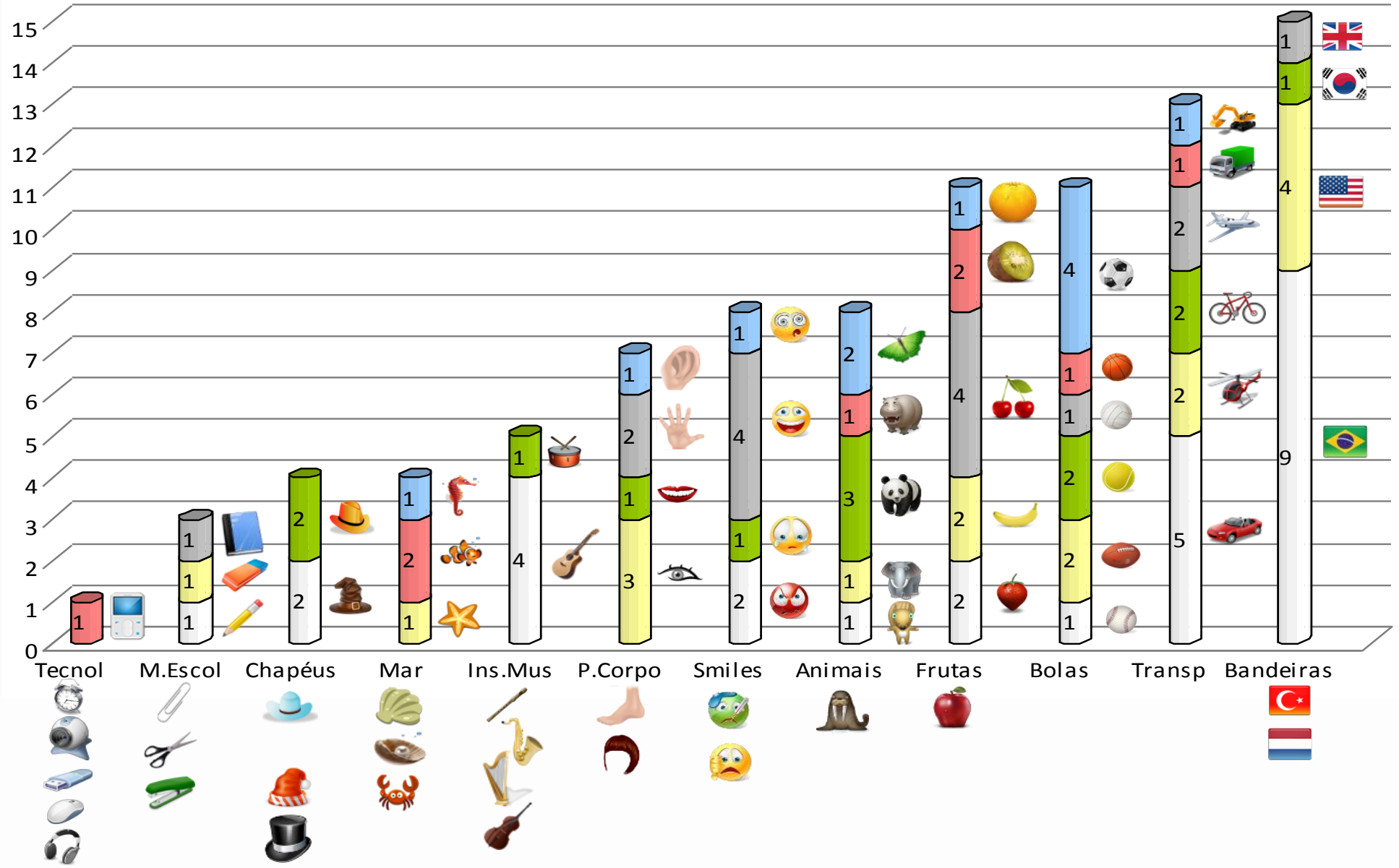
Odeio as senhas, mas sou obrigado a usa-las

Melhorando a experiência do usuário

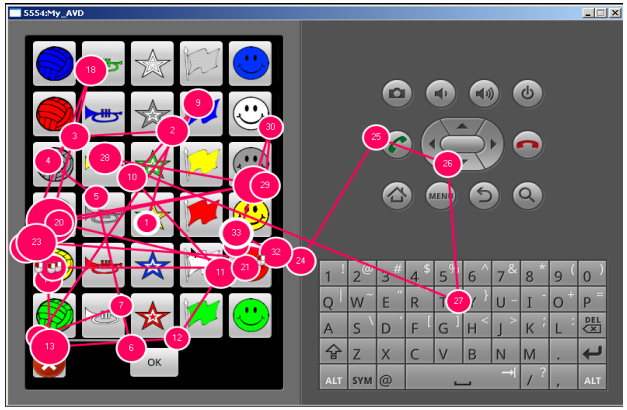
Senha icônica

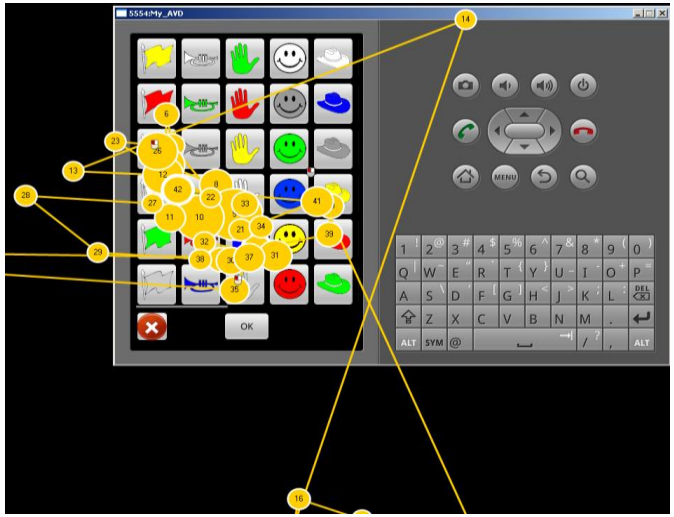


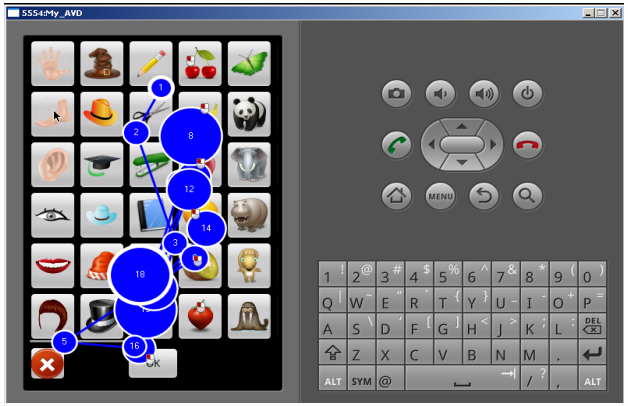
Categorias e imagens escolhidas



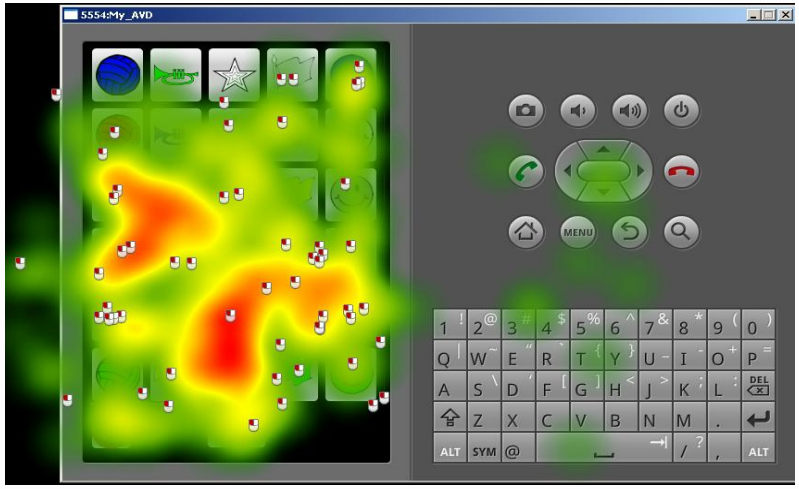
Exemplos do uso do Eyetracking e comportamentos observados

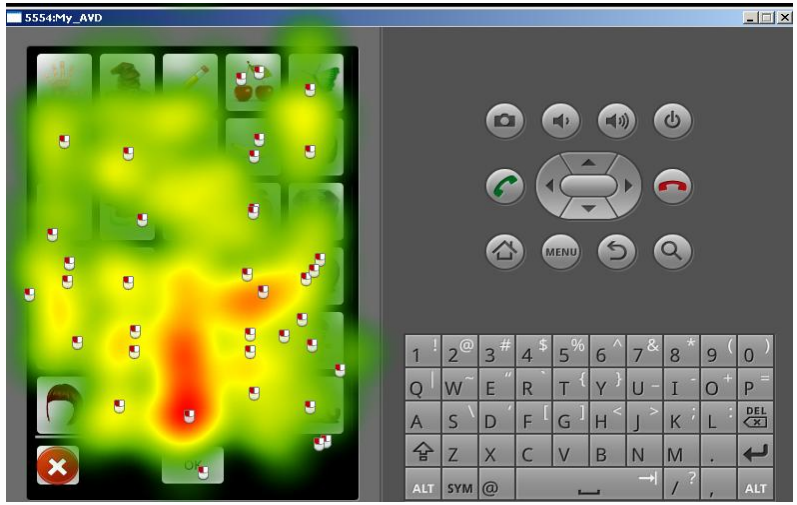






Exemplos do uso do Eyetracking e comportamentos observados

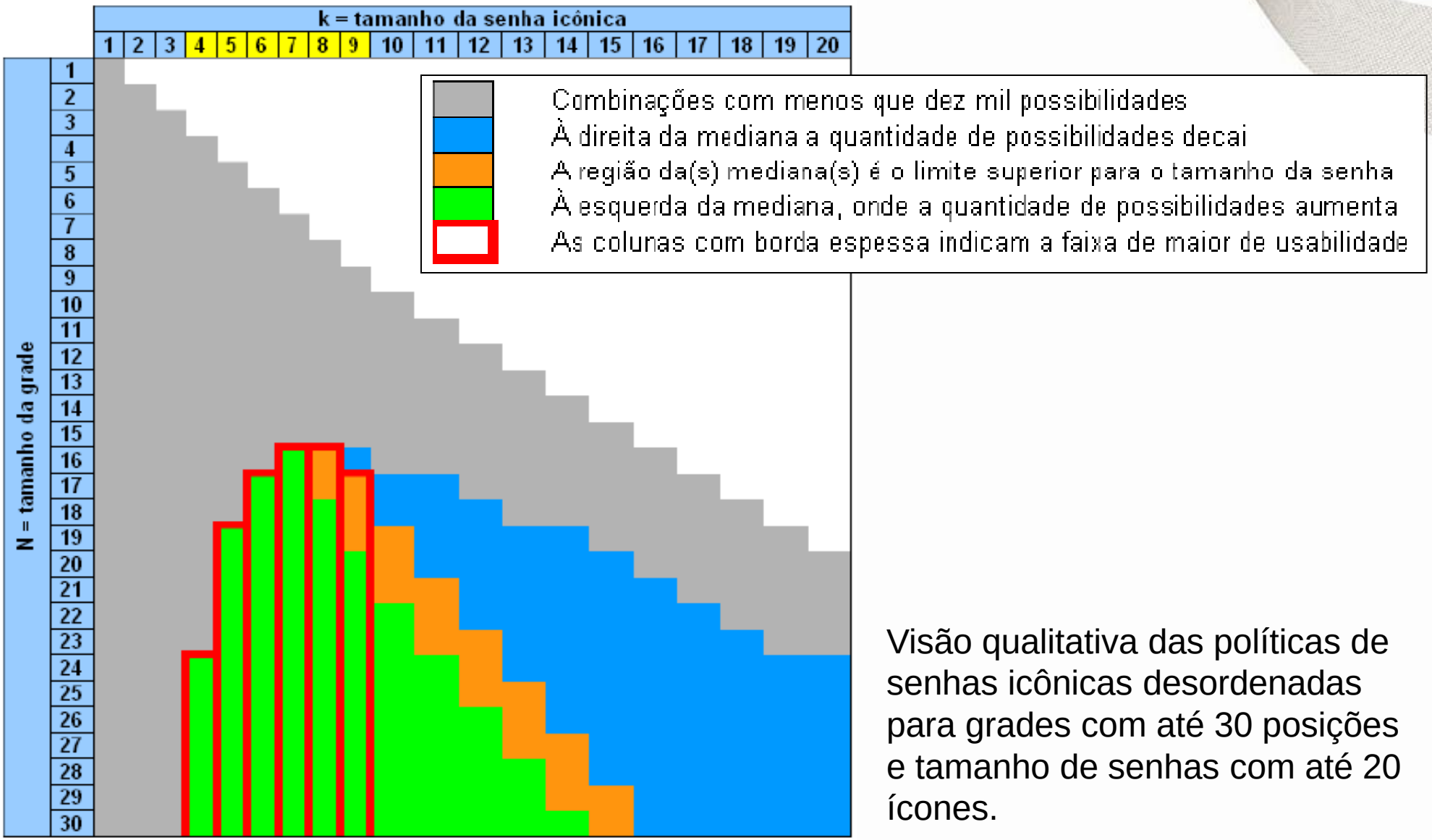




Fórmulas para determinação da quantidade de senhas icônicas

A quantidade de senhas icônicas desordenadas de tamanho fixo com k ícones escolhidos a partir de um repertório de n ícones.	$S(n,k)_{desordenada} = C_k^n = \frac{n!}{k!(n-k!)}$ (1)
A quantidade de senhas icônicas ordenadas de tamanho fixo, com k ícones de um grupo de n.	$S(n,k)_{ordenada} = A_k^n = \frac{n!}{(n-k)!} = (C_k^n) * k!$ (2)
A quantidade de senhas icônicas desordenadas e de tamanho variável, com k variando em um intervalo de li a ls de um grupo de n ícones.	$S_{desordenada}(n, li, ls) = \sum_{k=li}^{ls} C_k^n$ (3)
Quantidade de senhas icônicas desordenadas e de tamanho fixo k de n solicitadas em r rodadas.	$S_{rodada}(n,k,r) = (S_{desordenada}(n,k))^r$ (4)

Métricas para senhas icônicas desordenadas



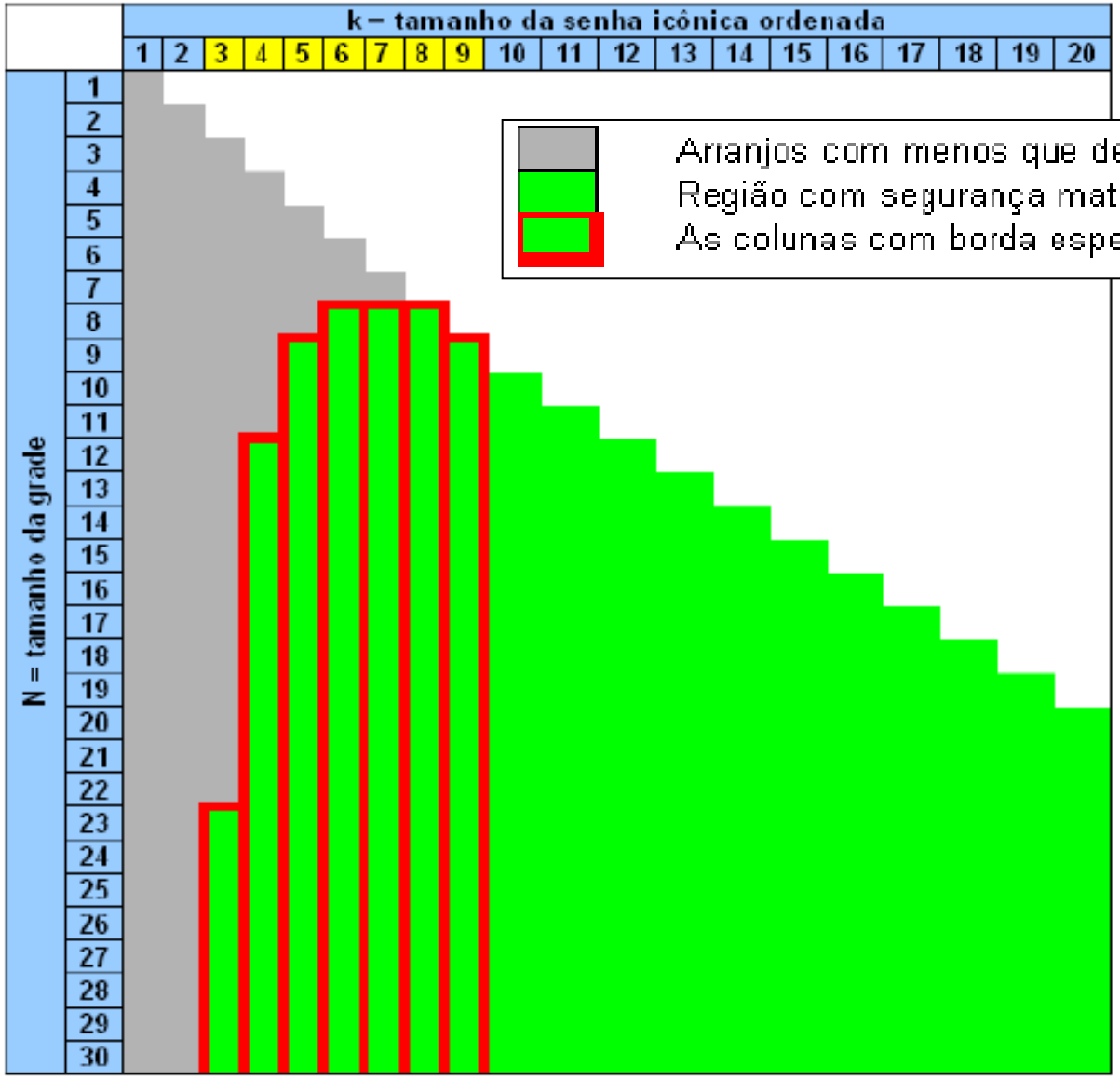
Visão qualitativa das políticas de senhas icônicas desordenadas para grades com até 30 posições e tamanho de senhas com até 20 ícones.

Métricas para senhas icônicas desordenadas

N = tamanho da grade	# bits para senha k						
	4	5	6	7	8	9	soma
16				13,48	13,65		14,57
17			13,60	14,25	14,57	14,57	16,30
18			14,18	14,96	15,42	15,57	17,12
19		13,51	14,73	15,62	16,21	16,50	17,97
20		13,92	15,24	16,24	16,94	17,36	18,70
21		14,31	15,73	16,83	17,63	18,17	19,39
22		14,68	16,19	17,38	18,29	18,92	20,05
23		15,04	16,62	17,90	18,90	19,64	20,69
24	13,38	15,38	17,04	18,40	19,49	20,32	21,30
25	13,63	15,70	17,43	18,87	20,04	20,96	21,88
26	13,87	16,01	17,81	19,33	20,58	21,58	22,43
27	14,10	16,30	18,18	19,76	21,08	22,16	22,97
28	14,32	16,58	18,52	20,18	21,57	22,72	23,48
29	14,54	16,86	18,86	20,57	22,03	23,26	23,97
30	14,74	17,12	19,18	20,96	22,48	23,77	24,45

 # de bits das combinações icônicas na faixa de usabilidade.

Métricas para senhas icônicas ordenadas



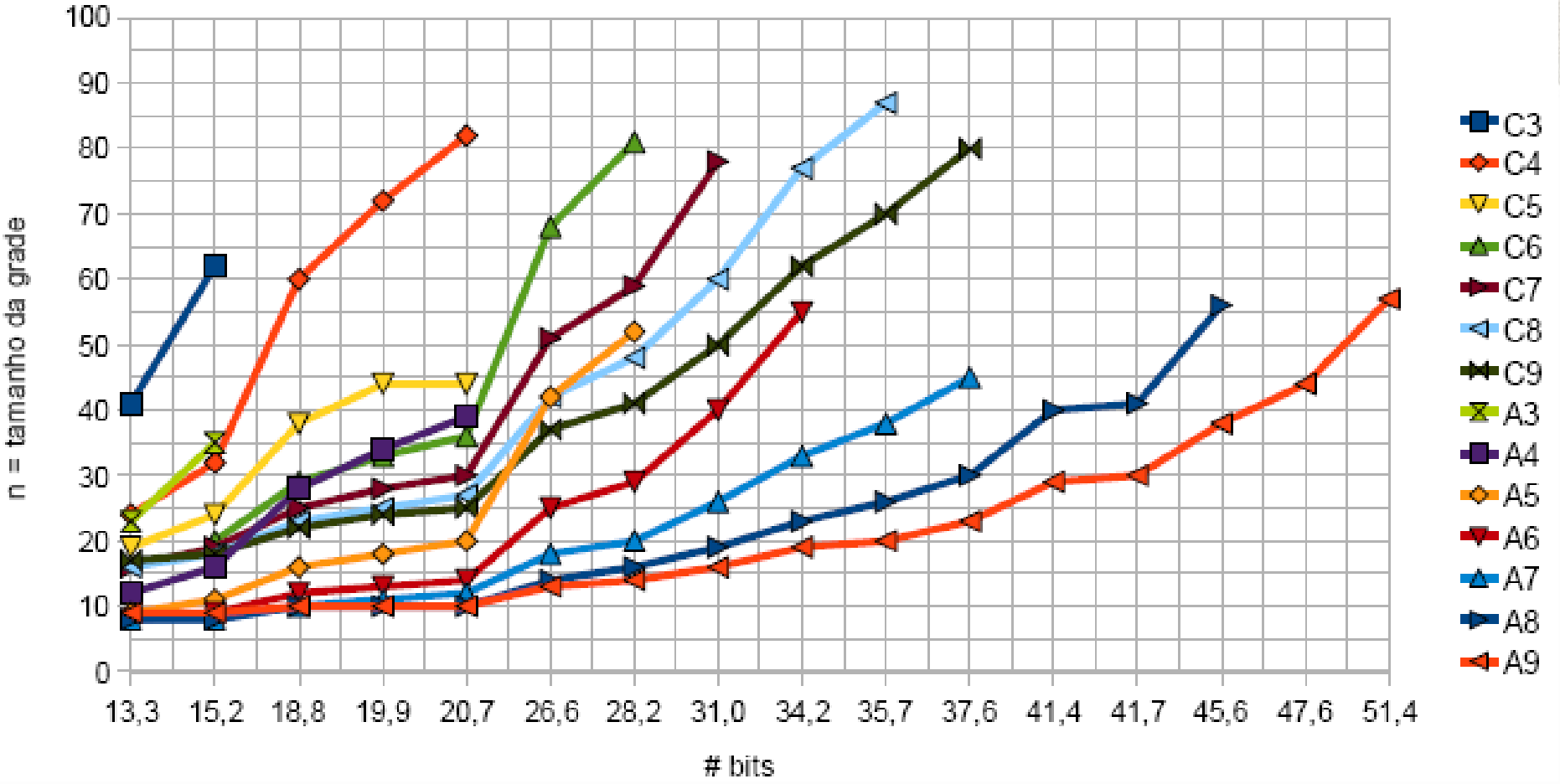
Visão qualitativa das políticas de senhas icônicas ordenada para grades com até 30 posições e tamanho de senhas com até 20 ícones.

Métricas para senhas icônicas ordenadas

N = tamanho da grade	# bits para senha k ordenada							
	3	4	5	6	7	8	9	soma
8				14,30	15,30	15,30		16,62
9			13,88	15,88	17,47	18,47	18,47	19,91
10			14,88	17,21	19,21	20,79	21,79	22,57
11			15,76	18,34	20,67	22,67	24,25	24,77
12		13,54	16,54	19,34	21,93	24,25	26,25	26,64
13		14,07	17,24	20,24	23,04	25,63	27,95	28,26
14		14,55	17,87	21,04	24,04	26,85	29,44	29,69
15		15,00	18,46	21,78	24,95	27,95	30,76	30,98
16		15,41	19,00	22,46	25,78	28,95	31,95	32,14
17		15,80	19,50	23,09	26,55	29,87	33,04	33,21
18		16,16	19,97	23,67	27,26	30,72	34,04	34,19
19		16,51	20,41	24,22	27,92	31,50	34,96	35,10
20		16,83	20,83	24,73	28,54	32,24	35,83	35,95
21		17,13	21,22	25,22	29,13	32,93	36,63	36,75
22		17,42	21,59	25,68	29,68	33,59	37,39	37,50
23	13,38	17,70	21,95	26,12	30,20	34,20	38,11	38,21
24	13,57	17,96	22,28	26,53	30,70	34,79	38,79	38,88
25	13,75	18,21	22,60	26,93	31,17	35,34	39,43	39,52
26	13,93	18,45	22,91	27,30	31,63	35,87	40,04	40,13
27	14,10	18,68	23,21	27,67	32,06	36,38	40,63	40,71
28	14,26	18,91	23,49	28,02	32,47	36,87	41,19	41,26
29	14,42	19,12	23,76	28,35	32,87	37,33	41,72	41,80
30	14,57	19,33	24,03	28,67	33,26	37,78	42,24	42,31

 # de bits dos arranjos icônicos na faixa de usabilidade.

Senhas icônicas em bits



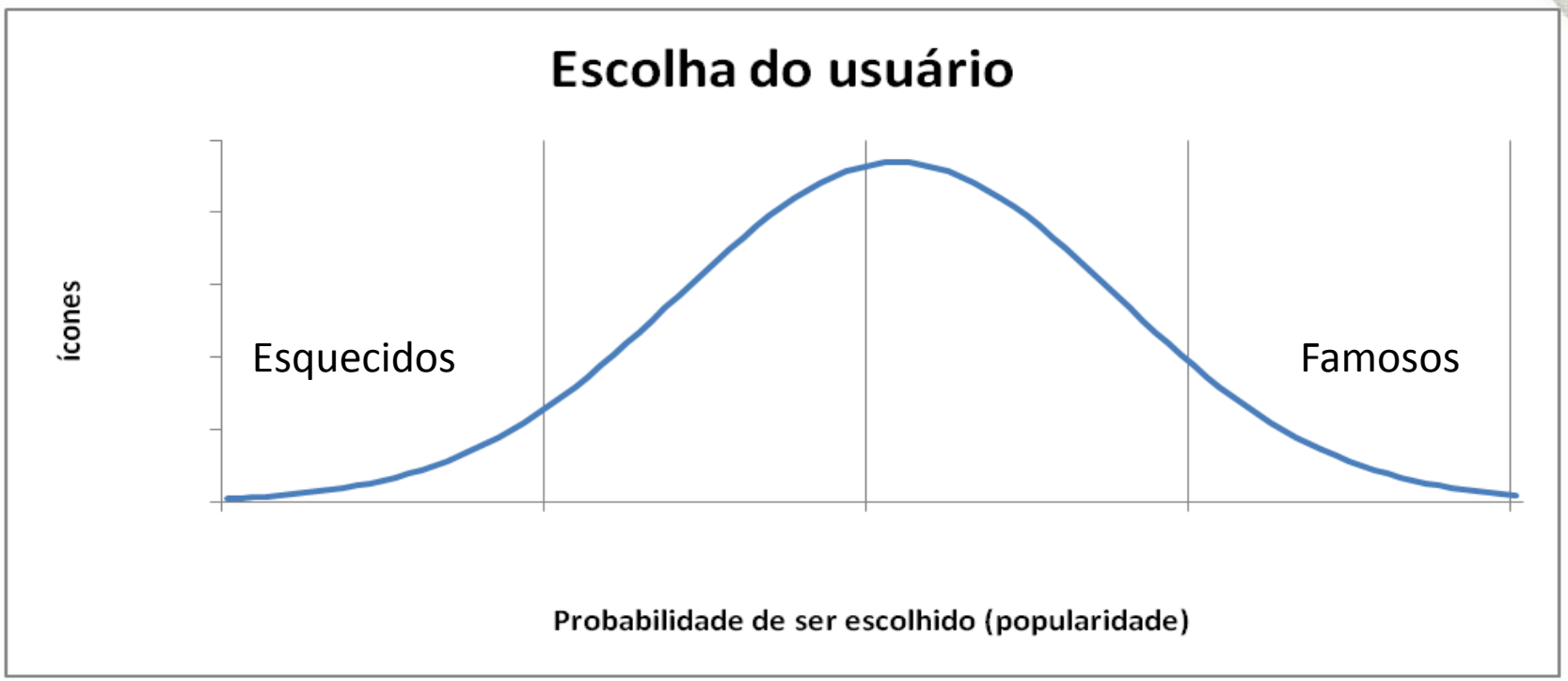
Seqüências de tamanho de grade para combinações icônicas (C#) e para arranjos icônicos (A#). O caractere # indica o tamanho da senha.

Métrica para integração de senhas icônicas e alfanuméricas

Políticas de senhas		Combinação icônica							Arranjo icônico							
k = Tamanho da senha icônica	# bits	3	4	5	6	7	8	9	3	4	5	6	7	8	9	
4 dígitos de 0-9	13,29	41	24	19	17	16	16	17	23	12	9	8	8	8	9	N = Tamanho de grade
Data (DDMMAA) 365d X 100a	15,16	62	32	24	20	19	18	18	35	16	11	9	8	8	9	
4 letras (AAAA) A-Z	18,8		60	38	29	25	23	22		28	16	12	10	10	10	
6 dígitos (999999) 0-9	19,93		72	44	33	28	25	24		34	18	13	11	10	10	
4 alfanuméricos A-Z e 0-9	20,68		82	44	36	30	27	25		39	20	14	12	10	10	
8 dígitos (99999999) 0-9	26,58				68	51	42	37			42	25	18	14	13	
6 letras (AAAAAA) A-Z	28,2				81	59	48	41			52	29	20	16	14	
6 alfanuméricos A-Z e 0-9	31,02					78	60	50				40	26	19	16	
6 letras c/ M != m	34,2						77	62				55	33	23	19	
6 alfanuméricos c/ M != m	35,73						87	70					38	26	20	
8 letras A-Z	37,6							80					45	30	23	
8 alfanuméricos A-Z e 0-9	41,36													40	29	
PCI: 7 alfanuméricos (26+10+26)	41,68													41	30	
8 letras c/ M != m	45,6													56	38	
8 alfanumérica c/ M != m	47,63														44	
Senha corporativa (8): 52+10+24	51,41														57	

Faixa de usabilidade de 3 a 9.

Curva de popularidade dos ícones



$$p(i) = f(popularidade_i)$$

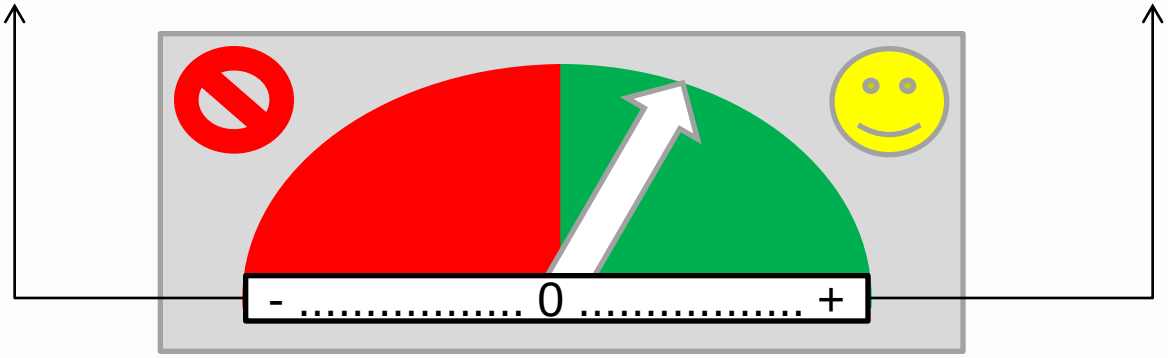
$$P(S(x)) = \prod_{i \in S(x)} p(i)$$

S(x) é o conjunto de x ícones selecionados pelo usuário

Medidor de qualidade/segurança

$$P(S(x))_{max} = \prod_{i=n}^{n-x} p(i)$$

$$P(S(x))_{min} = \prod_{i=1}^{x} p(i)$$



$A(x) \rightarrow$ *Conjuto de ícones aleatórios*

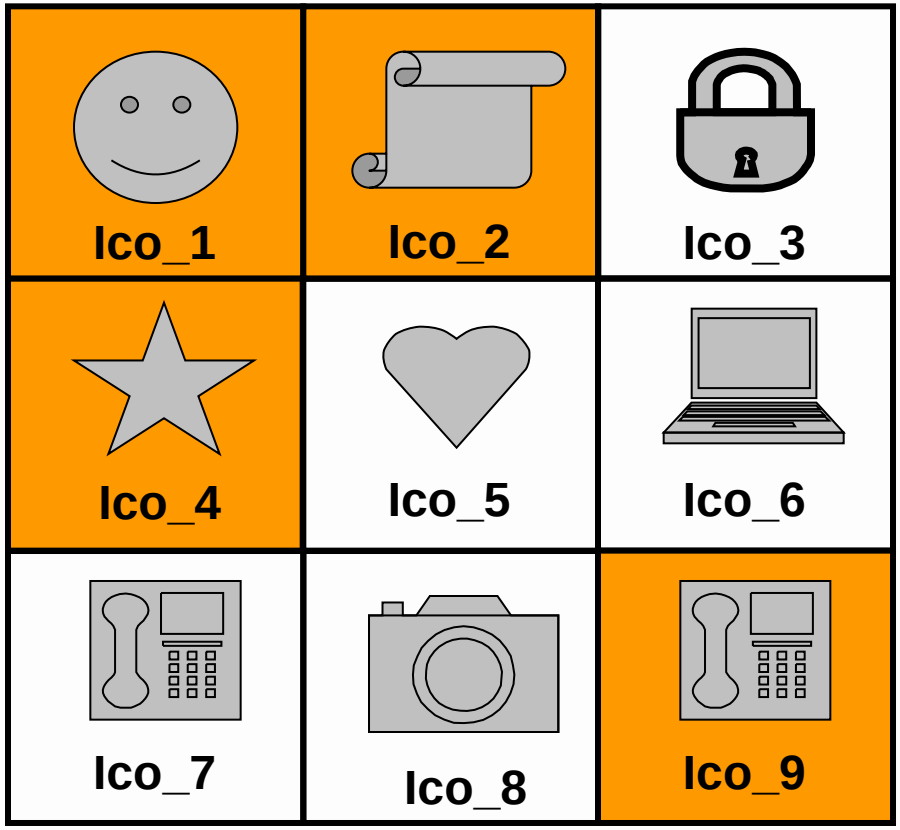
$S(x) \rightarrow$ *Conjuto de ícones selecionados*

$P(A(x)) - P(S(x)) = 0$ *Equivalente ao aleatório*

$P(A(x)) - P(S(x)) \geq 0$ *Senha boa*

$P(A(x)) - P(S(x)) < 0$ *Senha ruim*

Armazenamento da senha icônica



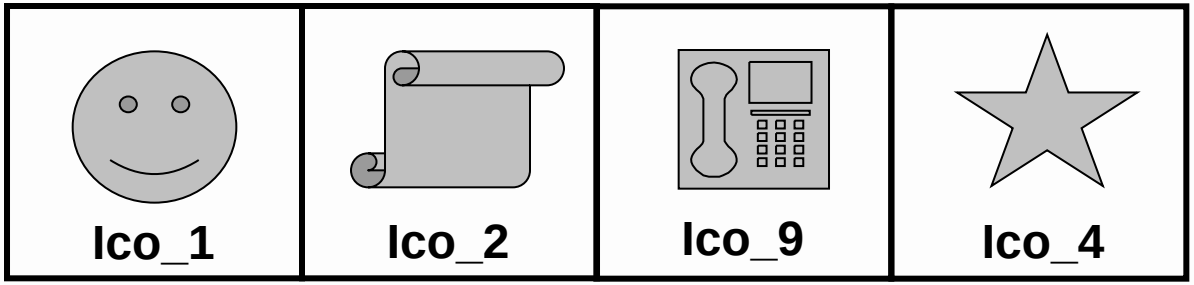
4! = 24 permutações de 1,2,9,4.

Qual será utilizada?
Esta é uma informação que vai no registro da senha.

Utilização vagamente análoga ao Salt.

Perm = permutação contada a partir da ordenada (ex. 1,2,9,4.)

Senha = 1,2,9,4



O registro da senha icônica

- userID:hash:salt:depth:perm
  - userID → depende da aplicação
  - Hash → >= 256 bits (SHA-2) preferível
    - 160 bits (SHA-1), se não houver outra opção
  - Salt → >= 48 bits, <= 128 bits
  - Depth → profundidade da cadeia de hash (key stretching)
    - H(H(H(…H(x)…))) = h
  - Perm → se senha desordenada, qual ordenação é usada no match?
    - Ico_1 < ico_2 < ico_3 < ico_4 ou Ico_1 > ico_2 > ico_3 > ico_4 ou …

O problema resolvido pelo perm

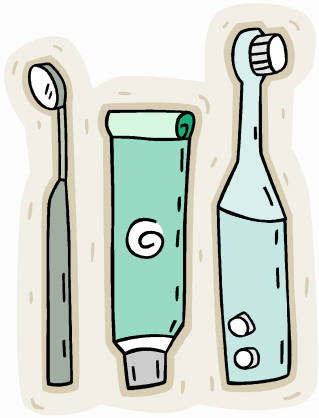
- n = tamanho do repertório
- r = tamanho da senha
- Senha desordenada
  - total = n! / (n-r)!
- Senha ordenada
  - C = n! / (n-r)! ; total = C*r!
- Senha desordenada é mais suscetível ao ataque de força bruta por um fator de r!

- Para senha de tamanho r, existem r! (fatorial) permutações
- Se senha desordenada, posição do ícone não é relevante
- Campo perm → se senha desordenada, qual permutação é usada no match?
  - Ico_1 < ico_2 < ico_3 < ico_4 (primeira opção)
  - … várias opções intermediárias …
  - Ico_1 > ico_2 > ico_3 > ico_4 (última opção)

O campo perm

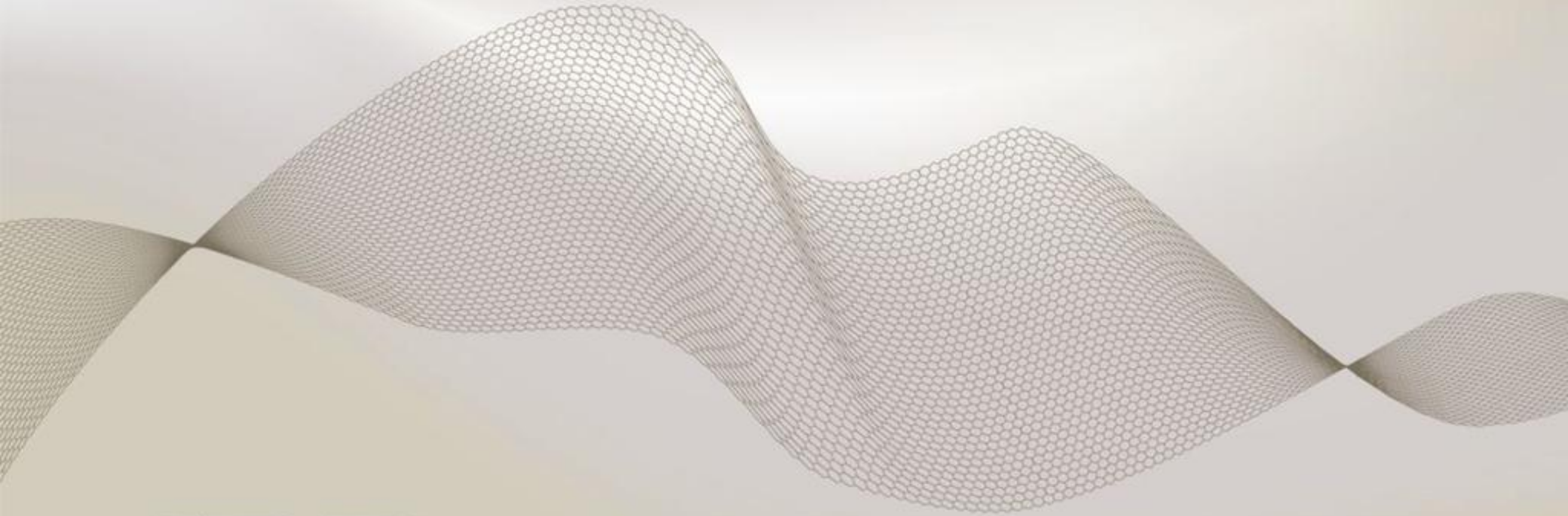
- A geração da permutação adequada ao match pode ser usada como estratégia de defesa contra força-bruta
  - aumento do tempo da computação.
  - Semelhante e complementar (mas não em substituição) ao stretching e ao salt.
- Recomendação de desempenho
  - senha de tamanho máximo 10
  - Algoritmo de geração de permutações é O(n!)

Concluindo com sabedoria



"Treat your password like
your toothbrush.
Don't let anybody else use it,
and get a new one every six months"

Alexandre Braga
ambraga@cpqd.com.br
(19) 3705-6255



www.cpqd.com.br